



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLAN

SISTEMAS DE INFORMACION: "DESARROLLO DE UN SISTEMA PARA UN CENTRO DE AYUDA A USUARIOS DE UNA TIENDA DE AUTOSERVICIO UTILIZANDO LA HERRAMIENTA ACTION REQUEST SYSTEM."

298028

TRABAJO DE SEMINARIO QUE PARA OBTENER EL TITULO DE: LICENCIADA EN INFORMATICA PRESENTA: ALEJANDRA OLIVARES VEGA

ASESOR: ING. MIGUEL ALVAREZ PASAYE.

CUAUTITLAN IZCALLI, EDO. DE MEXICO. 2001.



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.

FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLAN  
UNIDAD DE LA ADMINISTRACION ESCOLAR  
DEPARTAMENTO DE EXAMENES PROFESIONALES



UNIVERSIDAD NACIONAL  
AUTÓNOMA DE  
MÉXICO

DR. JUAN ANTONIO MONTARAZ CRESPO  
DIRECTOR DE LA FES CUAUTITLAN  
PRESENTE

DEPARTAMENTO DE  
EXAMENES PROFESIONALES

ATN: Q. Ma. del Carmen García Mijares  
Jefe del Departamento de Exámenes  
Profesionales de la FES Cuautitlán

Con base en el art. 51 del Reglamento de Exámenes Profesionales de la FES-Cuautitlán, nos permitimos comunicar a usted que revisamos el Trabajo de Seminario

sistemas de Información: "Desarrollo de un sistema para un

Centro de Ayuda a usuarios de una tienda de autoservicio

utilizando la herramienta Action Request System."

que presenta la pasante: Alejandra Olivares Vega

con número de cuenta: 9312502-4 para obtener el título de

Licenciada en Informática

Considerando que dicho trabajo reúne los requisitos necesarios para ser discutido en el EXÁMEN PROFESIONAL correspondiente, otorgamos nuestro VISTO BUENO

ATENTAMENTE  
"POR MI RAZA HABLARA EL ESPIRITU"

Cuautitlán Izcalli, Méx. a 06 de Septiembre de 2001

MODULO	PROFESOR	FIRMA
<u>I</u>	<u>M.C.D. Araceli Nivón Laghi</u>	<u>[Firma]</u>
<u>II</u>	<u>M.C.D. Valentín Roldán Vázquez</u>	<u>[Firma]</u>
<u>III</u>	<u>Inq. Miguel Álvarez Pasaye</u>	<u>[Firma]</u>

*Agradecimientos.*

*Quiero dar gracias a Dios por todas las bendiciones que he recibido, entre ellas, el darme la oportunidad de terminar mis estudios universitarios.*

*Mi más profundo agradecimiento a mi mamá, por estar conmigo en todo momento, por sus consejos y por darme los ánimos necesarios para seguir adelante.*

*A mi papá, por que sin su apoyo no lo hubiera logrado, agradezco su ejemplo de responsabilidad y de constante esfuerzo.*

*Gracias a mis hermanos (Mariano, Ricardo y Norma), por su apoyo y por formar parte de mi vida.*

*César David, gracias por tu apoyo incondicional, por tus regaños y consejos, por escucharme. Siempre que te necesitaba, ahí estabas. Gracias!*

*Gracias a la Universidad Nacional Autónoma de México y a los maestros que me compartieron sus conocimientos*

---

*Al Ing. Miguel Pasaye, a la M.C.C. Araceli Nivon, al Ing. Valentín Roldán y al Ing. Francisco Hiram por sus comentarios, observaciones y correcciones en el presente trabajo.*

*Mi sincero agradecimiento a Vianey Cortes, por darme las facilidades necesarias para la realización de este trabajo.*

*Y a todas las personas que no he mencionado, pero directa o indirectamente han contribuido para que termine esta etapa de mi vida.*

*Alejandra Olivares Vega.*

---

## Sistemas de Información:

*“ Desarrollo de un sistema para un Centro de Ayuda a Usuarios de una tienda de autoservicio utilizando la herramienta Action Request System”*

### **Objetivo General.**

Desarrollar un sistema para un Centro de Ayuda a Usuarios de una tienda de autoservicio utilizando una herramienta cliente/servidor

### **Objetivos Particulares.**

1. Analizar los procedimientos de un Centro de Ayuda
2. Describir la arquitectura cliente/servidor.
3. Describir las ventajas y desventajas de AR System.
4. Desarrollar un sistema para un Centro de Ayuda en base a los requerimientos del cliente.

---

---

**ÍNDICE**

	Pag.
Objetivo General	I
Objetivos Particulares	I
Introducción	V
1. SISTEMAS DE INFORMACIÓN	1-1
1.1. Componentes	1-1
1.2. Actividades de un sistema de información	1-2
1.3. Tipos de Sistemas de Información	1-3
1.4. Ciclo de vida del desarrollo de sistemas	1-6
1.5. Métodos para la adquisición de sistemas	1-10
1.5.1. Método tradicional	1-10
1.5.2. Compra de sistemas	1-14
1.5.3. Método tradicional vs compra de sistemas.	1-15
	2-1
2. ARQUITECTURA CLIENTE/SERVIDOR	
2.1. Evolución de la arquitectura cliente/servidor.	2-1
2.2. Estilos de la arquitectura cliente/servidor.	2-3
2.3. Características de los sistemas cliente/servidor.	2-6
2.4. Tipos de servidores	2-8
2.5. Llamadas a procedimientos remotos (RPC)	2-11
2.6. Manejo de mensajes y colas (MOM)	2-11
2.7. Comparación entre MOM y RPC	2-12
2.8. Protocolo TCP/IP	2-13
2.8.1. Direcciones IP	2-14

---

3. CENTROS DE AYUDA AL USUARIO	3-1
3.1. Beneficios	3-2
3.2. Utilidad en la base de datos	3-3
3.3. Requerimientos para la implementación del software de un Help Desk	3-5
3.3.1. Ejemplo	3-8
4. ACTION REQUEST SYSTEM	4-1
4.1. Remedy	4-1
4.2. Action Request System (AR System)	4-2
4.2.1. Beneficios del AR System	4-3
4.3. Arquitectura	4-5
4.3.1. Arquitectura cliente / servidor	4-5
4.3.2. Requerimientos del sistema	4-6
4.3.3. Capas del AR System Server	4-9
4.3.4. Programas y servicios del AR System	4-11
4.3.5. Servidores del AR System	4-12
4.4. Estructura de la base de datos	4-15
4.5. Objetos	4-16
4.6. Control de Acceso	4-18
4.6.1. Grupos explícitos e implícitos	4-19
4.6.2. Grupos reservados	4-19
4.6.3. Licenciamiento	4-21
4.6.3.1. Tipos de licencias	4-21
4.7. Usuarios del AR System	4-22
4.8. Componentes del AR System	4-23
4.9. Análisis de Work Flow	4-25
4.9.1. Diseño de una aplicación en AR System	4-26
4.9.2. Pasos básicos para el desarrollo de una aplicación en AR System.	4-28

---

---

5. CASO PRÁCTICO	5-1
5.1. Planteamiento	5-1
5.2. Requerimientos	5-2
5.3. Análisis y Diseño del Sistema	5-3
5.3.1. Definir el proceso.	5-3
5.3.2. Identificar los puntos de control o estados del proceso.	5-5
5.3.3. Identificar los roles involucrados.	5-6
5.3.4. Identificar las mediciones o reportes.	5-6
5.3.5. Definir los elementos de los datos (esquemas y campos) e interfaces externas.	5-6
5.4. Desarrollo	5-21
Conclusiones	VIII
Bibliografía	IX

## INTRODUCCIÓN.

Los Centros de Ayuda constituyen un fenómeno que está dando lugar a un cambio radical en la forma de operar de las empresas y de realizar negocios. Entre los servicios que se ofrecen en estos Centros se encuentran los siguientes:

- Atención al cliente / servicios.
- Encuestas telefónicas (estudios de mercado, sondeos de opinión, calidad y satisfacción de clientes).
- Acciones de Telemarketing<sup>1</sup>.
- Creación y actualización de bases de datos.
- Recepción de pedidos.
- Soporte técnico, etc.

Su actividad se ha desarrollado en los últimos 20 años de forma exponencial, hasta llegar al punto en que se hace prácticamente imposible para cualquier empresa no plantearse seriamente el ofrecer a sus clientes los servicios de uno de estos Centros de Ayuda.

Hoy en día, un Centro de Ayuda o Help Desk une la tecnología informática con las comunicaciones para servir de una forma integral al cliente, y mejorar el servicio que ofrece la empresa. Es, por lo tanto, dado el vertiginoso desarrollo de los dos sectores (informático y de comunicaciones), algo cada vez más importante en la vida de las empresas y en su relación con los clientes, así como su supervivencia en el mercado.

En el presente trabajo se desarrolla una aplicación cuya función principal es cubrir las necesidades que tiene un Centro de Ayuda a Usuarios de una tienda de Autoservicio para sistematizar su información, tomando en cuenta todas las actividades que tiene que cubrir, de acuerdo a los requerimientos del cliente.

El Centro de Ayuda a Usuarios o Help Desk para el cual se desarrollará el sistema, funcionará como único punto de contacto para los Usuarios finales internos de equipo de cómputo en la tienda de autoservicio, brindándoles soporte telefónico, para aquellos problemas de operación relacionados con el hardware y software de sus estaciones de trabajo.

El Objetivo del Centro de Ayuda a Usuarios es proporcionar servicio de soporte telefónico que contribuya a dar una rápida solución a los problemas que se presentan en los equipos de los usuarios finales.

Al contar con un sistema para el Centro de Ayuda a Usuarios, se registrarán los requerimientos y todos sus detalles, como el nombre del contacto, fecha y hora del registro, la persona a la que se le asignará, el tiempo de solución restante, así como otros datos. También se pueden analizar los reportes generados, para determinar que tipo de llamadas se registran con mayor frecuencia, los clientes que más reportan, los problemas más comunes, etc. Además de generar una base de conocimientos, al guardar las soluciones que se le dan a los problemas. Al contar con una base de conocimientos, el equipo de soporte ahorrará un tiempo precioso, porque ellos pueden hacer uso de esa información, solucionando mas rápidamente el problema y utilizando ese tiempo en otros requerimientos que quizá necesiten más de sus conocimientos o habilidades. Además, teóricamente, un sistema para un Centro de Ayuda debe ser capaz de hacer repetidas tareas fácilmente, consumiendo el menor tiempo posible.

Por lo tanto, el sistema mejorará considerablemente la calidad del servicio ofrecido por el Centro de Ayuda, porque permitirá resolver los problemas más eficazmente, dar un seguimiento efectivo, en caso de que no se cierren inmediatamente, hasta llegar a la solución, ganando con ello, la satisfacción del cliente.

---

<sup>1</sup> Mercadeo por teléfono.

Para cumplir el objetivo de este trabajo, se desarrollaran cinco capítulos, los cuales describimos a continuación.

En el primer capítulo, se hace referencia a los sistemas de información, sus componentes y su clasificación, así como el ciclo de vida para el desarrollo, esto es importante, porque la finalidad de este trabajo, es el desarrollo de un sistema.

En el capítulo dos, se describen las características que tiene un Centro de Ayuda o Help Desk<sup>2</sup>, así cómo los beneficios que trae a la organización la implementación de un sistema para su Help Desk y lo que se debe de considerar para la elección de la herramienta para su desarrollo o para la compra del mismo.

Para el desarrollo del sistema, se utilizará la herramienta Action Request System, la cual se basa en el modelo cliente/servidor, el cual se describirá en el tercer capítulo. Las características y las ventajas que ofrece la herramienta Action Request System se tratará en el cuarto capítulo, así como el método que propone la herramienta para el análisis, diseño y desarrollo de una aplicación para un Help Desk.

Por último, en el quinto capítulo se presentará el caso práctico, que es el desarrollo del sistema para el Centro de Ayuda a usuarios de una tienda de autoservicio, en base al método propuesto por la herramienta Action Request System. Sin olvidar, que la implementación del sistema no está dentro del alcance de este trabajo.

---

<sup>2</sup> Los dos términos se usan indistintamente a lo largo del trabajo.

## CAPÍTULO 1

### SISTEMAS DE INFORMACIÓN.

Los sistemas de información se pueden definir como un conjunto de componentes que interaccionan entre sí con el fin de apoyar las actividades de una empresa o negocio.

#### 1.1. COMPONENTES.

Podemos decir, que los componentes elementales que conforman los sistemas de información, son los siguientes:

- **Recursos Humanos.** Usuarios, directivos, miembros de los grupos de desarrollo de los sistemas de información, o cualquier persona que interactúa con el sistema de información, alimentándolo con datos o utilizando los resultados que genere.
- **Datos.** Materia prima empleada para crear información útil.
- **Procedimientos.** Actividades de la empresa (incluidas las de tipo directivo) y actividades de proceso de datos y generación de información que apoyan las actividades de la empresa. Incluyen las políticas y reglas de operación.
- **Telecomunicaciones.** Son básicamente hardware y software que facilitan la transmisión de texto, datos, imágenes y voz en forma electrónica.
- **Programas.** Son ejecutados por la computadora, y producen diferentes tipos de resultados. Los programas son parte del software del sistema de información que hará que los datos de entrada introducidos sean procesados correctamente y generen los resultados que se esperan.

- **Tecnología.** Hardware y software que sostienen los componentes anteriores.

También es importante el elemento control dentro de los sistemas, que consiste en:

1. Un estándar para lograr un desempeño aceptable
2. Un método para medir el desempeño actual
3. Un medio para comparar el desempeño actual contra el estándar
4. Un método de retroalimentación.

Los sistemas que pueden ajustar sus actividades para mantener niveles aceptables continúan funcionando. Aquellos que no lo hacen, tarde o temprano dejan de trabajar.

## 1.2. ACTIVIDADES DE UN SISTEMA DE INFORMACIÓN.

Un sistema de información realiza cuatro actividades básicas: entrada, almacenamiento, procesamiento y salida de información. A continuación se definen cada una de estas actividades:

- **Entrada de información.** Es el proceso mediante el cual el sistema de información toma los datos que requiere para procesar la información.
- **Almacenamiento de información.** A través de esta propiedad, el sistema puede recordar la información guardada en la sesión o proceso anterior. Esta información suele ser almacenada en estructuras de información denominadas archivos.
- **Procesamiento de información.** Es la capacidad del sistema de información para efectuar cálculos de acuerdo con una secuencia de

operaciones preestablecida. Estos cálculos pueden efectuarse con datos introducidos recientemente en el sistema o bien con datos que están almacenados. Esta característica de los sistemas permite la transformación de datos fuente en información que puede ser utilizada para la toma de decisiones.

- **Salida de información.** Es la capacidad de un sistema de información para sacar la información procesada o bien datos de entrada al exterior. Las unidades típicas de salida son las impresoras, estaciones de trabajo, diskettes, cintas magnéticas, la voz, los graficadores y los protters, entre otros. Es importante mencionar, que la salida de un sistema de información puede ser la entrada a otro sistema de información o módulo.

### 1.3. TIPOS DE SISTEMAS DE INFORMACIÓN.

Los sistemas de información son desarrollados con propósitos diferentes dependiendo de las necesidades del negocio. Deben cumplir con tres objetivos básicos dentro de las organizaciones:

1. Automatizar los procesos operativos.
2. Proporcionar información que sirva de apoyo al proceso de toma de decisiones.
3. Lograr ventajas competitivas a través de su implementación y uso.

De acuerdo a estos objetivos se pueden clasificar de la siguiente forma:

- **Sistemas de procesamiento de transacciones.** Son sistemas de información desarrollados para procesar gran cantidad de datos para transacciones rutinarias, tales como nómina o inventario, eliminando el

tedio de las transacciones operacionales necesarias y reducen el tiempo que alguna vez se requirió para ejecutarlas manualmente.

- **Sistemas de automatización de oficina y sistemas de manejo de conocimiento.** Al nivel de conocimiento de la organización hay dos clases de sistemas. Los sistemas de automatización de oficina que dan soporte a los empleados que manejan datos, pero que no crean un nuevo conocimiento, sino que utilizan la información para analizarla y transformar datos, o para compartirla o diseminarla formalmente para toda la organización. Los sistemas de manejo de conocimiento, dan soporte a los profesionales, tales como científicos, ingenieros y doctores, les ayudan a crear un nuevo conocimiento que contribuya a la organización o a la sociedad en general.
  
- **Sistemas de información gerencial.** Estos sistemas no reemplazan a los sistemas de procesamiento de transacciones, sino que lo incluyen. Los sistemas de información gerencial requieren que las personas, el software y el hardware trabajen al unísono, porque además de soportar las tareas organizacionales, incluyen el análisis de decisiones y la toma de decisiones. Para poder ligar la información, los usuarios de un sistema de información gerencial comparten una base de datos común, la cual guarda modelos que ayudan a los usuarios a interpretar y aplicar esos mismos datos, produciendo información que es usada en la toma de decisiones.
  
- **Sistemas de apoyo a decisiones.** Estos sistemas enfatizan el apoyo a la toma de decisiones en todas sus fases, aunque la decisión actual todavía es del dominio del tomador de decisiones. Los sistemas de apoyo a decisiones están más hechos a la medida de la persona o grupo que los usa que los sistemas de información gerencial, pero son similares porque ambos dependen de una base de datos como fuente.

- **Sistemas expertos e inteligencia artificial.** La inteligencia artificial puede ser considerada la meta de los sistemas expertos. Dos caminos de la investigación de la inteligencia artificial son la comprensión del lenguaje natural y el análisis de la habilidad para razonar un problema y llegar a conclusiones lógicas. Un sistema experto captura en forma efectiva y usa el conocimiento de un experto para resolver un problema particular experimentado en una organización. A diferencia de los sistemas de apoyo a decisiones, un sistema experto selecciona la mejor solución a un problema o a una clase específica de problemas.
  
- **Sistemas de apoyo a decisiones de grupo.** Están orientados para reunir a un grupo, a fin de que resuelva un problema con ayuda de varios apoyos como votaciones, cuestionarios, aportación de ideas y creación de escenarios.
  
- **Sistemas de apoyo a ejecutivos.** Estos sistemas ayudan a los ejecutivos para organizar sus interacciones con el ambiente externo, proporcionando apoyo de gráficos y comunicaciones en lugares accesibles tales como salas de juntas u oficinas personales corporativas.
  
- **Sistemas estratégicos.** Su principal función es lograr ventajas que los competidores no poseen, tales como ventajas en costos y servicios diferenciados con clientes y proveedores. Son creadores de barreras de entrada al negocio. Apoyan al proceso de innovación de productos y procesos dentro de la empresa, debido a que buscan ventajas respecto a los competidores y una forma de hacerlo es innovando y creando productos y procesos.

#### 1.4. CICLO DE VIDA DEL DESARROLLO DE SISTEMAS.

El desarrollo de sistemas es un proceso formado por las etapas de análisis y diseño, que sostiene que los sistemas de información son desarrollados de mejor manera mediante el uso de un ciclo específico de actividades de el analista y el usuario. Varias de estas actividades se pueden suceder simultáneamente, y las actividades pueden ser repetidas.

El ciclo de vida del desarrollo de sistemas, consta de las siguientes actividades:

- **Identificación de problemas, oportunidades y objetivos.** Esta etapa es crítica para el éxito del resto del proyecto, ya que si se comete un error, se perderá tiempo y dinero corrigiéndolo posteriormente. Las actividades de esta fase consisten en entrevistas con los administradores de los usuarios, resumir el conocimiento obtenido, estimar el alcance del proyecto y documentar los resultados. La salida de esta fase es un estudio de factibilidad tanto técnica, económica y operacional, que contiene una definición del problema y la sumarización de objetivos.
  
- **Determinación de los requerimientos de información.** En esta fase se determina que información necesitan los usuarios para realizar su trabajo. Se deben estudiar los procesos de la organización para dar respuesta a las siguientes preguntas clave:
  1. ¿Qué es lo que se hace?
  2. ¿Cómo se hace?
  3. ¿Con qué frecuencia se presenta?
  4. ¿Qué tan grande es el volumen de transacciones o de decisiones?
  5. ¿Cuál es el grado de eficiencia con el que se efectúan las tareas?
  6. ¿Existe algún problema?
  7. Si existe un problema, ¿qué tan serio es?
  8. Si existe un problema, ¿cuál es la causa que lo origina?

Para contestar estas preguntas, también se recurre a las entrevistas o a los cuestionarios con los usuarios, e incluso la observación en condiciones reales de las actividades del trabajo y, en algunas ocasiones, muestras de formas y documentos que se utilizan dentro del proceso. Al final de esta etapa, se debe de comprender el por qué de las funciones del negocio y tener información completa sobre las personas, objetivos, datos y procedimientos involucrados.

- **Análisis de las necesidades del sistema.** Una herramienta utilizada en esta fase es el uso de diagramas de flujos de datos, para determinar la entrada, proceso y salida de las funciones del negocio en forma gráfica estructurada. A partir de estos diagramas de flujo de datos se desarrolla un diccionario de datos, que lista todos los conceptos de datos usados en el sistema, así como sus especificaciones. También se analizan las decisiones estructuradas, que son aquellas para las que pueden ser determinadas las condiciones como alternativas de condición, acciones y reglas de acción. En esta etapa, se prepara una propuesta de sistema que resume lo que ha sido encontrado, proporciona análisis de costo/beneficio de las alternativas y se hacen recomendaciones sobre lo que debe ser hecho (en caso de haberlo).
  
- **Diseño del sistema recomendado.** En esta fase se producen los detalles que establecen la forma en la que el sistema cumplirá con los requerimientos identificados durante la fase de análisis. Es el diseño lógico del sistema de información. Se identifican los reportes y demás salidas que debe producir el sistema. Además, se diseña la interfaz con el usuario, la cual conecta al usuario con el sistema y es, por lo tanto, extremadamente importante. También incluye el diseño de archivos o bases de datos que guardarán la mayor parte de los datos para la toma de decisiones en la organización. Por último, se deben de diseñar procedimientos de control y respaldo para proteger al sistema y a los datos y producir paquetes de

especificaciones de programa para los programadores. Cada paquete debe contener diseños de entrada y salida, especificaciones de archivos y detalles de procesamiento, diagramas de flujo de datos, arboles de decisiones, un diagrama de flujo de sistemas y los nombres y funciones de cualesquiera de las rutinas de código que hayan sido escritas. Esto con la finalidad de facilitar su mantenimiento y actualización.

- **Desarrollo y documentación del software.** En esta fase se pueden instalar (o modificar y después instalar) software comprado a terceros o escribir programas diseñados a la medida. La elección depende del costo de cada alternativa, el tiempo disponible para escribir el software y de la disponibilidad de los programadores. Los programadores también son responsables de la documentación de los programas y de proporcionar una explicación de cómo y por qué ciertos procedimientos se codifican en determinada forma. La documentación es esencial para probar el programa y llevar a cabo el mantenimiento una vez que la aplicación se encuentra instalada.
  
- **Pruebas y mantenimiento del sistema.** Es mucho menos costoso encontrar problemas antes de que el sistema sea entregado a los usuarios. Por lo que es indispensable realizar pruebas al sistema, en algunas ocasiones se realiza por los programadores, por los analistas de sistemas o por los mismos usuarios, con datos ficticios o datos reales, lo importante, es asegurarse que no tenga fallas. El mantenimiento del sistema y de sus documentación comienzan en esta fase y es efectuado rutinariamente a lo largo de la vida del sistema de información. Las organizaciones gastan gran cantidad de dinero en el mantenimiento de sistemas, es por lo que se tienen que tener mucho cuidado a lo largo del ciclo de vida de desarrollo del sistema para asegurar que el mantenimiento se mantenga al mínimo.

- **Implementación y evaluación del sistema.** Es el proceso de verificar e instalar nuevo equipo, entrenar a los usuarios, instalar la aplicación y construir todos los archivos de datos necesarios para utilizarla. Dependiendo del tamaño de la organización se empleará la aplicación y el riesgo asociado con su uso, puede elegirse comenzar la operación del sistema sólo en un área de la empresa (prueba piloto). Algunas veces se deja que los dos sistemas, el viejo y el nuevo, trabajen en forma paralela con la finalidad de comparar resultados. En otras circunstancias deja de utilizarse el viejo sistema determinado día, para comenzar a emplear el nuevo al día siguiente. La estrategia de implantación se elige de acuerdo a la situación de la organización. Una vez instaladas, las aplicaciones se utilizan durante muchos años, sin embargo, las organizaciones y los usuarios cambian con el paso del tiempo, incluso el ambiente es diferente, por consiguiente, es indudable que debe darse mantenimiento a las aplicaciones; realizar cambios y modificaciones en el software, archivos o procedimientos para satisfacer las nuevas necesidades de los usuarios, por lo que los sistemas de información deben mantenerse siempre al día. La evaluación se muestra como parte de esta fase final del ciclo de vida del desarrollo del sistema, se lleva a cabo para identificar los puntos débiles y fuertes del sistema. Un criterio principal que debe ser satisfecho es si los usuarios pretendidos ya están usando el sistema.

Debe de tomarse en cuenta que los sistemas trabajan en forma cíclica. Cuando se termina una fase del desarrollo del sistema y pasa a la siguiente, el descubrimiento de un problema puede obligar a que se regrese a la fase anterior y se modifique el trabajo allí hecho. Lo importante es que al finalizar, el sistema cumpla eficientemente con las necesidades de la organización para el que fue elaborado.

## 1.5. MÉTODOS PARA LA ADQUISICIÓN DE SISTEMAS.

Después de conocer las variables que afectan en la calidad del desarrollo de sistemas y el ciclo de vida, es importante que una empresa considere las fuentes de proveerse de sistemas, las cuales son:

- **El método tradicional**, consiste en que la empresa desarrolle el sistema internamente o contrate servicios externos para ello.
- **La compra de sistemas** desarrollados y terminados o desarrollados de manera parcial por otras compañías que se encuentran en el mercado de desarrollo de software.

A continuación se explica más detalladamente estos métodos:

### 1.5.1. Modelo tradicional.

El Modelo Tradicional de desarrollo cumple con todas las fases del ciclo de vida del desarrollo de sistemas, explicado anteriormente, que se inician con un estudio de factibilidad de la realización y terminan con la operación del sistema. A éste método se le conoce como **cascada o caída de agua**, debido a que las fases son consecutivas. A pesar de que se sigue un orden en la realización de cada una de las fases, es posible regresar a la fase anterior para hacer correcciones en caso de ser necesario.

El método tradicional de desarrollo de sistemas, cuenta con una técnica de diseño y documentación, que se explica enseguida:

### **Técnica de diseño y documentación.**

Es necesario contar con técnicas adecuadas para realizar las fases de análisis y diseño para tener documentado todo el proceso. El diseño de un sistema puede ser ascendente (*bottom-up*) o descendente (*top-down*). Cuando se realiza un diseño ascendente se inicia por los niveles operativos de la organización y, posteriormente, se definen los requisitos de los niveles más altos, según las necesidades de sistemas que se tengan. En el caso del diseño descendente, el diseñador parte de la estructura global de la empresa y de sus objetivos y busca la mejor manera de satisfacerlos al desarrollar el sistema.

Por otro lado, la documentación debe proporcionar un panorama del sistema, especificar los procedimientos que se llevan a cabo y la forma de operarlo. Además de esta documentación dirigida al usuario, debe documentarse y detallarse la estructura de archivos y programas con el objeto de que pueda realizarse un mantenimiento adecuado.

## Diagramas de flujo de datos.

Un diagrama de flujo de datos describe el flujo y la dirección de los datos de un procedimiento, por ejemplo, en las áreas operativas de un negocio. La sencillez de esta técnica se debe a que únicamente hace uso de cuatro diferentes símbolos para describir los flujos de datos. Un rectángulo para señalar entidades externas al sistema, como por ejemplo proveedores, clientes u otros departamentos. Un círculo o cuadrado para describir los procesos (lugar en donde se realiza una actividad con los datos para transformarlos en información). Un semirectángulo define lugar de almacenamiento de datos y finalmente, una flecha sirve para dar la dirección de los datos, los cuales viajan de una entidad externa hacia un proceso y viceversa. En la siguiente figura se muestra la simbología utilizada por los diagramas de flujo de datos:

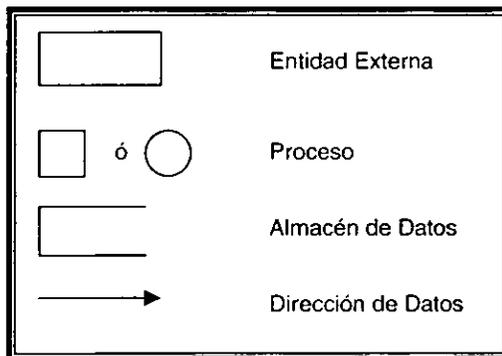


Figura 1.1 Simbología utilizada por los diagramas de flujos de datos.

## Pruebas del sistema

Este proceso se realiza con el fin de asegurar que el sistema este libre de errores y debe realizarse durante todo el proceso y no solo en la fase final. Para realizar las pruebas puede utilizarse *el modelo de Kendall & Kendall*, el cual consiste en cuatro tipos de prueba. El primer tipo de pruebas se realiza a nivel de los programadores para comprobar los programas utilizando datos de prueba o ficticios. El segundo deben realizarlo los analistas para probar el funcionamiento entre los programas, utilizando para ello datos de prueba, para verificar que el sistema trabaja como unidad. En el tercero participan los operadores, que prueban todo el sistema con datos de prueba y por último, en el cuarto nivel participan los usuarios, probando todo el sistema con datos reales.

## Mantenimiento.

Después de que el sistema está instalado se le debe de dar mantenimiento, esto significa que los programas de computadora deben ser modificados y mantenidos actualizados.

El mantenimiento se realiza por dos razones. La primera de estas es para corregir errores de software. Sin importar que tan completamente se pruebe el sistema, se deslizan errores. La otra razón es para mejorar las capacidades del software en respuesta a las necesidades organizacionales cambiantes, por lo general involucran tres situaciones:

1. *Los usuarios frecuentemente solicitan características adicionales después de que se familiarizan con el sistema de cómputo y sus capacidades.* Estas características solicitadas pueden ser tan simples como el desplegar otro dato en algún reporte o tan complicadas como el desarrollo de un nuevo software.
2. *El negocio cambia a través del tiempo.* Se debe modificar el software para abarcar cambios tales como nuevos requerimientos de reportes corporativos, la necesidad de producir nueva información para clientes, etc.

3. *El hardware y software están cambiando a un ritmo acelerado.* Un sistema que usa tecnología antigua puede ser modificado para usar las capacidades de una tecnología más nueva.

El mantenimiento es un proceso continúa a lo largo del ciclo de vida de un sistema de información, conforme pasa el tiempo y cambia el negocio y la tecnología, los esfuerzos de mantenimiento se incrementan dramáticamente, e incluso en cierto punto es más conveniente realizar un nuevo estudio del sistema, debido a que el costo de mantenimiento continuado del sistema es más alto que la creación de un sistema de información completamente nuevo.

#### **1.5.2. COMPRA DE SISTEMAS.**

En ocasiones una empresa necesita un sistema que ya se encuentra disponible en el mercado, y resulta más costoso comprarlo que desarrollarlo utilizando el método tradicional. Se debe observar y analizar los diferentes sistemas que ofrece el mercado y determinar cuál o cuáles son útiles para la empresa.

Un error en la compra de paquetes puede afectar profundamente las operaciones de la organización, incrementar costos y por consecuencia, una disminución en las utilidades y del nivel de servicio a clientes y usuarios. Para asegurarse de la calidad del sistema, debe tomarse en cuenta lo siguiente:

- Que el sistema satisfaga todos los requerimientos del usuario.
- Que opere con alta confiabilidad.
- Que sea entregado a tiempo para poder iniciar su operación
- Que cumpla con los requerimientos del presupuesto.
- Haber evaluado la magnitud de las adecuaciones del sistema para la empresa que lo adquiere.

### 1.5.3. MÉTODO TRADICIONAL VS COMPRA DE SISTEMAS

A continuación se muestra una tabla en donde se comparan las principales características de los métodos de adquisición de sistemas que se mencionaron anteriormente.

Concepto	Método Tradicional	Compra de Sistemas
<b>Costo</b>	Costo del desarrollo	Costo del sistema, más el costo de las modificaciones necesarias.
<b>Tiempo</b>	Mayor.	Menor.
<b>Mantenimiento</b>	Se realiza internamente	Se realiza en forma externa
<b>Tipo de aplicación</b>	Hecho a la medida	Aplicación general
<b>Cuidado con:</b>	Fechas optimistas.	No ser "conejiillo de indias"
	Rotación durante el proceso.	Asumir que las modificaciones son menores.
		Tener el visto bueno del usuario antes de comprar.
<b>Implantación</b>	Costos similares	Costos Similares.

Tabla 1.1. Comparación del método de tradicional de desarrollo de sistemas, contra la compra de los mismos.<sup>3</sup>

<sup>3</sup> Obtenido del libro "Cliente/servidor. Guía de supervivencia.", 2ª ed. Orfali, Robert.

## CAPÍTULO 2

### ARQUITECTURA CLIENTE/SERVIDOR.

La definición de la arquitectura cliente/servidor, se deduce del término mismo. Clientes y servidores son entidades lógicas independientes que operan en conjunto a través de una red para realizar una tarea. La arquitectura nos permite determinar la estructura y forma de los sistemas de cliente servidor que podemos construir para la satisfacción de diversas necesidades.

A continuación, se presenta la evolución de la arquitectura cliente/servidor.

#### 2.1. EVOLUCIÓN DE LA ARQUITECTURA CLIENTE /SERVIDOR

La arquitectura cliente/servidor se ha desarrollado a través de varias etapas, las cuales se presentan en seguida:

- **La era de la computadora central.** Desde sus inicios el modelo de administración de datos a través de computadoras se basó en el uso de terminales remotas, que se conectaban de manera directa a una computadora central. Dicha computadora central se encargaba de prestar servicios caracterizados por que cada servicio se prestaba solo a un grupo exclusivo de usuarios. El personal de la llamada área de sistemas se encargaba de consolidar o integrar la información cuando las necesidades de los usuarios lo exigían.
  
- **La era de las computadoras dedicadas.** Esta es la era en la que cada servicio empleaba su propia computadora que permitía que los usuarios de ese servicio se conectaran directamente. Esto es consecuencia de la aparición de computadoras pequeñas, de fácil uso, más baratas y más poderosas que las convencionales. Este modelo de funcionamiento posee en la actualidad importantes inconvenientes. El primero que, conforme crece el número de

usuarios que requieren acceso a los datos administrados por cada sistema, se presenta la necesidad de hacer uso de computadoras cada vez más poderosas en sus sistemas de entrada/salida; los nuevos costos producidos son consecuencia del aumento del número de usuarios a quienes se les tiene que otorgar licencias para poder hacer uso legal de los programas contenidos en los sistemas de información. El segundo inconveniente es que estas computadoras son incapaces de comunicarse entre sí y por tanto la información compartida es nula. Estos inconvenientes que se observan hoy son consecuencia directa de la tecnología que se empleo para dar origen a este esquema y que hoy es obsoleto.

- **La era de la conexión libre.** Hace mas de 10 años que la computadoras de escritorio aparecieron de manera masiva. Esto permitió que parte apreciable de la carga de trabajo de cómputo tanto en el ámbito de cálculo como en el ámbito de la presentación se lleven a cabo desde el escritorio del usuario. En muchos de los casos el usuario obtiene la información que necesita de alguna computadora de servicio. Estas computadoras de escritorio se conectan a las computadoras de servicio empleando software que permite la emulación de algún tipo de terminal. En otros de los casos se les transfiere la información haciendo uso de recursos magnéticos o por transcripción.
  
- **La era del cómputo a través de redes.** Esta es la era que esta basada en el concepto de redes de computadoras, en la que la información reside en una o varias computadoras. Los usuarios de esta información hacen uso de computadoras para laborar y todas ellas se encuentran conectadas entre sí. Esto brinda la posibilidad de que todos los usuarios puedan acceder a la información de todas las computadoras y a la vez que los diversos sistemas intercambien información.

➤ **La era de la arquitectura cliente/servidor.** En esta arquitectura la computadora de cada uno de los usuarios, llamada cliente, produce una demanda de información a cualquiera de las computadoras que proporcionan información, conocidas como servidores; estos últimos responden a la demanda del cliente que la produjo. Los clientes y los servidores pueden estar conectados a una red local o una red amplia, como la que se puede implementar en una empresa o a una red mundial como lo es la Internet. Bajo este modelo cada usuario tiene la libertad de obtener la información que requiera en un momento dado proveniente de una o varias fuentes locales o distantes y de procesarla como según le convenga. Los distintos servidores también pueden intercambiar información dentro de esta arquitectura.

Existen diversos estilos del modelo cliente/servidor, enseguida se presentan sus características, sus ventajas y sus desventajas de cada uno:

## 2.2. ESTILOS DEL MODELO CLIENTE/SERVIDOR

Existen varios estilos del modelo cliente/servidor, a continuación se presentan sus principales características, así como sus ventajas y sus desventajas:

### ➤ **Presentación distribuida.**

1. Se distribuye la interfaz entre el cliente y la plataforma servidora.
2. La aplicación y los datos están ambos en el servidor.
3. Similar a la arquitectura tradicional de un Host y Terminales.
4. La PC se aprovecha sólo para mejorar la interfaz gráfica del usuario.

#### **Ventajas:**

- Revitaliza los sistemas antiguos.
- Bajo costo de desarrollo.
- No hay cambios en los sistemas existentes.

**Desventajas:**

- El sistema sigue en el Host.
- No se aprovecha la GUI y/o LAN.
- La interfaz del usuario se mantiene en muchas plataformas.

➤ **Presentación remota.**

1. La interfaz para el usuario está completamente en el cliente.
2. La aplicación y los datos están en el servidor.

**Ventajas:**

- La interfaz del usuario aprovecha bien la GUI y la LAN.
- La aplicación aprovecha el Host.
- Adecuado para algunos tipos de aplicaciones de apoyo a la toma de decisiones.

**Desventajas:**

- Las aplicaciones pueden ser complejas de desarrollar.
- Los programas de la aplicación siguen en el Host.
- El alto volumen de tráfico en la red puede hacer difícil la operación de aplicaciones muy pesadas.

➤ **Lógica distribuida.**

1. La interfaz está en el cliente.
2. La base de datos está en el servidor.
3. La lógica de la aplicación está distribuida entre el cliente y el servidor.

**Ventajas:**

- Arquitectura más común que puede manejar todo tipo de aplicaciones.

- Los programas del sistema pueden distribuirse al nodo mas apropiado.
- Pueden utilizarse con sistemas existentes.

**Desventajas:**

- Es difícil de diseñar.
- Difícil prueba y mantenimiento si los programas del cliente y el servidor están hechos en distintos lenguajes de programación.
- No son manejados por la GUI 4GL.

➤ **Administración de datos remota.**

1. En el cliente residen tanto la interfaz como los procesos de la aplicación.
2. Las bases de datos están en el servidor.
3. Es lo que comúnmente imaginamos como aplicación cliente servidor

**Ventajas:**

- Configuración típica de la herramienta GUI 4GL.
- Muy adecuada para las aplicaciones de apoyo a las decisiones del usuario final.
- Fácil de desarrollar ya que los programas de aplicación no están distribuidos.
- Se descargan los programas del Host.

**Desventajas**

- No maneja aplicaciones pesadas eficientemente.
- La totalidad de los datos viaja por la red, ya que no hay procesamiento que realice el Host.

➤ **Base de datos distribuida.**

1. La interfaz, los procesos de la aplicación, y , parte de los datos de la base de datos están en cliente.
2. El resto de los datos están en el servidor.

**Ventajas:**

- Configuración soportada por herramientas GUI 4GL.
- Adecuada para las aplicaciones de apoyo al usuario final.
- Apoya acceso a datos almacenados en ambientes heterogéneos.
- Ubicación de los datos es transparente para la aplicación.

**Desventajas:**

- No maneja aplicaciones grandes eficientemente.
- El acceso a la base de datos distribuida es dependiente del proveedor del software administrador de bases de datos.

### 2.3. CARACTERÍSTICAS DE LOS SISTEMAS CLIENTE/SERVIDOR.

Los sistemas cliente/servidor poseen las siguientes características:

- **Servicio.** Cliente/servidor es fundamentalmente una relación entre procesos ejecutados en aparatos distintos. El proceso del servidor hace de este un proveedor de servicios, el cliente es un consumidor de servicios.
- **Recursos compartidos.** Un servidor puede atender a muchos clientes, al mismo tiempo y regular su acceso a recursos compartidos.

- **Protocolos asimétricos.** Entre clientes y servidor se establece una relación de muchos a uno, son siempre los clientes los que inician el diálogo al solicitar un servicio. Los servidores aguardan pasivamente las solicitudes de los clientes.
  
- **Transparencia de ubicación.** El servidor es un proceso que puede residir en el mismo aparato que el cliente, o en un aparato distinto a lo largo de la red. El software de cliente/servidor suele ocultarles a los clientes la ubicación del servidor mediante el redireccionamiento de las llamadas de servicio en caso necesario. Un programa puede ser cliente, un servidor o ambos.
  
- **Mezcla e igualdad.** El software ideal de cliente servidor es independiente del hardware o de las plataformas del software del sistema operativo.
  
- **Intercambios basados en mensajes.** Clientes y servidores son sistemas holgadamente acoplados que interactúan a través de un mecanismo de transmisión de mensajes. El mensaje es el mecanismo de entrega para las solicitudes y respuestas del servicio.
  
- **Encapsulamiento de servicios.** El servidor es un especialista. Un mensaje le indica a un servidor que servicio se solicita; éste se le envía luego al servidor para determinar el cumplimiento de la tarea. Los servidores pueden ser sustituidos sin afectar a los clientes, siempre y cuando la interfaz para la publicación del mensaje no cambie.
  
- **Facilidad de escalabilidad.** Los sistemas de cliente/servidor pueden escalarse horizontal o verticalmente. La escalabilidad horizontal significa la adición o eliminación de estaciones de trabajo del cliente con apenas un ligero impacto en el desempeño. La escalabilidad vertical significa migrar a un aparato servidor más grande y más veloz o a servidores múltiples.

- **Integridad.** El código del servidor y los datos del servidor se conservan centralmente, lo que resulta en un mantenimiento de menor costo y en la protección de la integridad de los datos compartidos. Al mismo tiempo, los clientes mantienen su individualidad e independencia.

#### 2.4. TIPOS DE SERVIDORES.

Podemos clasificar a los servidores, por la naturaleza del servicio que ofrece a sus clientes:

- **Servidores de Archivos.** Con éste servidor el cliente envía solicitudes de registros de archivos al servidor de archivos a través de una red. Se trata de una modalidad de servicio de datos muy primitivo, pues se requieren numerosos intercambios de mensajes a través de la red para obtener los datos solicitados.
- **Servidores de Bases de Datos.** Con un servidor de bases de datos, el cliente envía solicitudes de SQL en calidad de mensajes al servidor de la base de datos. Los resultados de cada orden de SQL son devueltos por medio de la red. El código que procesa la solicitud de SQL y los datos, residen en la misma máquina, el servicio hace uso de su propia capacidad de procesamiento para encontrar los datos solicitados, en lugar de hacerle llegar todos los registros al cliente para que éste encuentre sus propios datos.
- **Servidor de Transacciones.** Con este servidor, el cliente invoca procedimientos remotos que residen en el servidor con un mecanismo de base de datos de SQL. Estos procedimientos, remotos en el servidor ejecutan un grupo de instrucciones de SQL. El intercambio por la red consiste en un solo mensaje de solicitud / respuesta (a diferencia de lo que ocurre con el servidor de base de datos en cuyo caso es necesario un mensaje de solicitud /

respuesta para cada instrucción de SQL en una transacción). Los enunciados de SQL aciertan o fallan todos como una sola unidad. A estas instrucciones de SQL agrupadas, se les conoce como transacciones. Con un servidor de transacciones se puede crear la aplicación cliente/servidor generando el código tanto de clientes, como de servidor. El componente de cliente suele incluir una interfaz gráfica de usuario GUI (Graphical User Interface). El componente del servidor consiste por lo general en transacciones de SQL contra una base de datos. A estas aplicaciones se les llama procesamiento de transacciones en línea OLTP (Online Transaction Processing).

- **Servidores de Groupware.** El groupware dirige la administración de información semiestructurada como texto, imagen, correo, tableros de aviso y flujos de trabajo. Estos sistemas cliente/servidor establecen un contacto directo entre personas. Lotus Notes es el ejemplo más conocido de sistemas de este tipo, aunque varias aplicaciones más también satisfacen parcialmente las mismas necesidades. Software de groupware especializado puede integrarse a paquetes de API de cliente/servidor de un proveedor. En la mayoría de los casos, se crean aplicaciones usando un lenguaje de creación de scripts e interfaces basadas en formas ofrecidos por el proveedor. El middleware de comunicación entre el cliente y el servidor es propio de cada distribuidor.
  
- **Servidor de Objetos.** Con este servidor, la aplicación cliente/servidor se genera como un conjunto de objetos de comunicación. Los objetos del cliente se comunican con los objetos del servidor mediante un corredor de solicitudes de objetos, ORB (Object Request Broker). El cliente invoca un método de un objeto remoto. El ORB localiza una instancia de esa clase de servidor de objetos, invoca el método solicitado y envía los resultados al objeto del cliente. Los objetos del servidor deben ofrecer soporte de concurrencia y participación. El ORB se encarga de reunir todos los elementos.

- **Servidor Web.** El World Wide Web es la primera aplicación cliente/servidor integrado por clientes "universales" compactos, portátiles, en comunicación con servidores superamplios. En su modalidad más simple, un servidor Web envía documentos cuando los clientes los solicitan por su nombre. Clientes y servidores se comunican mediante un protocolo semejante a RPC denominado *http (hypertext transfer protocol)*. Este protocolo define un conjunto simple de órdenes; los parámetros se transmiten en cadenas, sin estipulaciones de datos teclados.

Las aplicaciones cliente/servidor también se pueden diferenciar por la forma en que la aplicación distribuida se divide entre el cliente y el servidor. En el **modelo de servidor amplio**, se le atribuyen más funciones al servidor y en el **modelo de cliente grande** se procede a la inversa. Los servidores de groupware, transacciones y servidores Web son ejemplos de servidores amplios; los servidores de bases de datos y archivos, son ejemplos de clientes grandes. Los objetos distribuidos pueden ser de ambas clases.

## 2.5. LLAMADAS A PROCEDIMIENTOS REMOTOS (RPC)

Es una petición efectuada por una máquina a otra sobre la red. Las RPC, constituyen middleware, operan sobre diversas plataformas de red y protocolos de comunicación. En esencia una RPC es una llamada o petición que realiza una computadora dirigida a otra computadora. Es un proceso de petición/respuesta en el cual el emisor espera una respuesta, lo que implica que generalmente las RPCs son llamadas en tiempo real que tiene lugar sobre interfaces orientadas a la conexión.

RPC permite a los clientes interactuar con múltiples servidores y a los usuarios manejar múltiples clientes simultáneamente. Los RPC ayudan a dividir una tarea en componentes más pequeños que pueden ejecutarse en sistemas diferentes, de modo que aprovechan las ventajas del entorno de la informática distribuida. Un sistema operativo y una aplicación que proporciona soporte a hilos<sup>4</sup> pueden realizar múltiples procesos al mismo tiempo en lugar de un proceso tras otro.

## 2.6. MANEJO DE MENSAJES Y COLAS (MOM)

MOM permite que mensajes de propósito general sean intercambiados en unos sistemas cliente servidor mediante colas de mensajes. Las aplicaciones se comunican a través de redes insertando simplemente mensajes en colas y obteniendo mensajes en colas.

---

<sup>4</sup> Los hilos son unidades de concurrencia provistas dentro del programa mismo. Son útiles para la creación de programas servidores muy concurrentes y sujetos a eventos. Cada evento en espera puede ser asignado a un hilo, que bloqueará las acciones hasta que el evento efectivamente ocurra. Mientras tanto, otros hilos pueden emplear productivamente los ciclos del CPU para desempeñar tareas útiles.

**2.7 COMPARACIÓN ENTRE MOM Y RPC.**

A continuación se comparan las principales características entre MOM y RPC.

Características	MOM	RPC
Similitud.	Semejante al correo	Semejante al teléfono
Relación temporal de cliente servidor.	Asíncrona: Clientes y servidores pueden operar a diferentes tiempos y velocidades.	Síncrona: Clientes y servidores deben operar concurrentemente. Los servidores deben marchar al paso de los clientes.
Secuencia cliente servidor.	Sin secuencia fija	Los servidores deben ponerse en marcha primero para que los clientes puedan comunicarse con ellos
Estilo.	Colas	Llamada – respuesta
Equilibrio de cargas.	Puede emplearse una sola cola para implementar atención de acuerdo con el orden de recepción o con una política basada en prioridades.	Se requiere un monitor de TP independiente, es decir de un software especializado para administrar y priorizar.
Desempeño.	Lento: Se requiere de una etapa intermedia.	Rápido.
Procesamiento asíncrono.	Si: Se requiere de colas y actividades	Limitado: Se requiere de hilos y códigos complejos para la administración de hilos.

Tabla 2.1. Tabla comparativa entre MOM y RPC.

## 2.8. PROTOCOLO TCP/IP

Un protocolo establece una descripción formal de los formatos que deberán presentar los mensajes para poder ser intercambiados por equipos de cómputo; además definen las reglas que ellos deben seguir para lograrlo.

El protocolo de transporte de Internet: "*Transmission Control Protocol/Internet Protocol*" (TCP/IP) fue desarrollado en la década de los 60 por colaboradores del Pentágono con la finalidad de crear una red de computadoras a prueba de fallos para las situaciones de crisis, a la que se le dio el nombre de ARPANET.

El protocolo TCP/IP adquirió las propiedades siguientes:

- Es un protocolo de transporte orientado a paquetes que fracciona los datos en bloques, que se transmiten por separado. Los distintos bloques pueden seguir vías de transmisión diferentes desde el origen hasta el destino. En el destino se recompone el paquete original de datos. Las informaciones necesarias para todo ello se encuentran en la línea de cabecera de cada uno de los paquetes de datos.
- El TCP/IP es un protocolo con capacidad de encaminamiento, es decir, es capaz de determinar correctamente el desplazamiento de los datos a través de la red: el requisito más importante para las redes WAN.
- El TCP/IP es seguro contra los fallos producidos en las distintas líneas de la red: los paquetes de datos se envían por líneas distintas hasta el destino, en el que se recomponen.
- Con el TCP/IP se puede conectar entre sí sistemas operativos completamente diferentes.
- El TCP/IP se puede utilizar como plataforma para numerosas aplicaciones cliente/servidor que envían los datos con este protocolo; pero, además el TCP/IP también soporta otros niveles de protocolos de la aplicación.

### 2.8.1. DIRECCIONES IP

Para que en una red dos computadoras puedan comunicarse entre sí, deben estar identificadas con precisión. Por lo que el protocolo TCP/IP utiliza un identificador denominado dirección IP, cuya longitud es de 32 bits. Esta dirección identifica tanto a la red a la que pertenece una computadora, como a ella misma dentro de dicha red.

Las computadoras se identifican por un número binario de 32 posiciones. Este número se agrupa en cuatro bloques de números binarios de 8 posiciones. Cada uno de estos bloques representa un número decimal entre 0 y 255. En la notación decimal, los distintos bloques se separan unos de otros con un punto. El resultado es el conjunto de direcciones IP únicas, por ejemplo: 196.171.43.12

Las direcciones IP de una red TCP/IP tienen que desempeñar las funciones siguientes:

- Para poder direccionar inequívocamente las computadoras de la red, sus direcciones sólo se podrán conceder una sola vez.
- A las subredes se tendrá que acceder por medio de tales direcciones o subconjuntos de una dirección.
- El formato de la dirección tiene que permitir una numeración lo suficientemente amplia como para poder abarcar una gran cantidad de computadoras (ya que el protocolo TCP/IP se usa en la red universal Internet).

## CAPÍTULO 3

### CENTROS DE AYUDA AL USUARIO

El Centro de Ayuda al Usuario, también es conocido como *Help Desk*, Centro de Soporte Técnico, Centro Administrador de Problemas, o Centro de Soporte al Cliente o el nombre que la compañía le quiera dar, es el punto de contacto al que recurre un cliente cuando tiene un requerimiento o problema que necesite solucionar.

El *Help Desk* o Centro de ayuda necesita un *software* para guardar los registros de las llamadas y habilitar a una persona con un acceso electrónico para que pueda resolver el problema. Pero un *software* puede hacer mucho más que esto, también puede:

- Registrar el requerimiento y todos sus detalles, como el nombre del contacto, la fecha y el tiempo del problema, la persona que es asignada para su solución, etc.
- Crear gráficas de acuerdo al tipo de llamada o cliente, facilitando el análisis de las tendencias del sistemas.
- Enviar un requerimiento, vía correo electrónico, a la persona asignada para que sea resuelto.
- Auditar la PC del usuario en forma constante, con un *software* especializado.
- Guardar en una base de datos los problemas y las soluciones, para formar una base de conocimientos.

Un sistema para un Centro de Ayuda se puede clasificar dentro de los **sistemas de automatización de oficina y sistemas de manejo de conocimiento**, porque estará dando soporte a los analistas de un Centro de Ayuda, en el manejo de la información proporcionada por el usuario, acerca de su requerimiento y a su vez, también propone soluciones con su base de conocimiento que se alimenta día con día con las soluciones dadas a los problemas que se presentan.

### 3.1. BENEFICIOS.

Al instalar un *software* en el Centro de Ayuda, se obtienen muchos beneficios. Para la organización, el costo por soporte técnico es elevado y en la actualidad, la satisfacción del cliente es lo más importante. Las empresas que cumplan ó superen las necesidades de los clientes, no solo sobrevivirán, sino que prosperarán. Por lo que con un Centro de Ayuda automatizado cumplirá con las expectativas del cliente, manteniéndolo contento, lo cual asegura su lealtad para la organización.

Los empleados se ven beneficiados, porque, teóricamente, el *software* para el Centro de Ayuda a Usuarios, es capaz de hacer tareas repetitivas fácilmente y con un mínimo consumo de tiempo. Se puede utilizar una base de conocimientos interna o externa, de manera que se puede encontrar la respuesta al problema que se esta presentando. El equipo de soporte técnico ahorra un tiempo muy valioso porque puede utilizar un sistema que este haciendo algo del trabajo de ellos. Y si pueden ahorrar tiempo, esto significa que ellos pueden utilizarlo en otros requerimientos, quizá en requerimientos que tengan mayor prioridad. Tal vez no se vea muy significativo, pero hay que considerar el volumen de transacciones del Centro de Ayuda en 2 o 3 años. Los segundos que se pueden ahorrar al utilizar un sistema, se vuelven muy valiosos.

El usuario también se ve muy beneficiado, porque un centro de ayuda automatizado, permitirá dar un soporte más rápido y exacto a sus usuarios. El resultado final, serán muchos clientes satisfechos.

Además de conocer en que puede beneficiar un *software* al Centro de Ayuda, también es importante saber que es lo que hace. Es importante recordar que manejar una eficiente relación con el cliente es indispensable para toda organización, y si a esto agregamos clientes satisfechos, la organización crecerá día con día.

### 3.2. Utilidad en la Base de Datos.

Un sistema en el Help Desk puede registrar en una base de datos cada llamada telefónica y permite que el personal de soporte técnico guarde todos los datos relevantes de la llamada y las soluciones a las mismas. Ayudando a la organización en lo siguiente:

- Facilita el almacenamiento de los datos, como la hora en que se hizo la llamada, una descripción del requerimiento y a quién se asignó el mismo.
- El sistema para *help desk* hará evidente cuantos *tickets*<sup>5</sup> de clientes se tienen y cuales se han solucionado, de esta forma se evita que accidentalmente el *ticket* se pierda o se olvide y no se atienda.
- En la base de datos no solo se almacenan datos relacionados con los *tickets*, como direcciones, números telefónicos y contactos, también es posible realizar búsquedas, reportes y gráficas.
- Con algunos sistemas para *help desk* es posible registrar los *tickets* de acuerdo al inventario del equipo, esto asegura que solo se reporte una vez la falla del equipo, solucionando el requerimiento del cliente más rápidamente y en forma precisa.

---

<sup>5</sup> Un *ticket* es una colección de información que describe un evento o incidente como un problema o un requerimiento de servicio.

Algunos sistemas de *Help Desk* incluyen una **base de conocimiento**, en la cual se almacenan el requerimiento y la solución que se dio, permitiendo crear una relación de las conclusiones. También algunos sistemas permiten almacenar las conclusiones en forma de preguntas y respuestas. De esta manera, el tiempo de solución para un requerimiento similar, disminuye en el futuro. Por lo que el *Help Desk* ahorrará tiempo y dinero, dos cosas que son muy valiosas para cualquier organización. Si el personal de soporte utiliza menos tiempo en la solución de los requerimientos, ese mismo personal puede recibir más llamadas, evitando contratar más personal.

#### **Uso de la base de datos.**

Todas las llamadas y respuestas guardadas en la base de datos son muy valiosas. Con esa información se pueden analizar las tendencias. Es decir, se pueden generar reportes en donde se puedan analizar los requerimientos agrupados por equipo, por falla, en resumen o detallados, o en forma de gráficas. Se pueden filtrar los datos con una búsqueda específica. Muchos sistemas para *Help Desk* facilitan estas búsquedas con reportes rutinarios o permiten el uso de una herramienta externa.

➤ **Reportes Rutinarios.** El *software* para *Help desk* deben ser capaces de producir reportes que presenten datos de lo que ocurre todos los días. Muchos sistemas tienen varios reportes predefinidos para solucionar las preguntas más comunes, como:

- ¿Qué departamentos, clientes, áreas, etc, llaman?
- Seleccionar los departamentos o clientes que llaman más frecuentemente.
- ¿Cuál es el promedio de llamadas recibidas al día?
- ¿Cuál es el promedio de tiempo que se toma en resolver un *ticket*?
- ¿Qué tantos *tickets* tiene asignado cada miembro del grupo de soporte?
- ¿Algunas piezas de cierto equipo se rompen más que otras?

- **Reportes Externos.** Aunque el software para el *Help Desk* tiene varios reportes con las búsquedas más comunes, algunos sistemas no pueden hacer todos. Tal vez sea necesario adquirir alguna herramienta externa que permita al cliente crear sus propios reportes. Algún *software* de *Help Desk* permite esto vía código, con lenguajes de programación, como C, C++, SQL, etc. Otros permiten elaborar reportes en una aplicación, como es *Crystal Reports* y entonces enviar estos datos al software de *Help Desk* utilizado. Naturalmente, para elaborar estos reportes se requiere de conocimientos de programador para el código de esos programas especiales. A pesar de que es más costoso, se pueden crear toda clase de reportes que cubran la demanda de la organización en el análisis de las tendencias.

Obviamente a lo que llevará el analizar la información almacenada en la base de datos, es al ahorro de dinero. Por ejemplo, si se analiza la tendencia de cual impresora se descompone más frecuentemente, podemos aconsejar al administrador que compre otra clase de impresoras o tal vez sea el uso que le de el cliente, etc. Disminuyendo las llamadas referentes a esos equipos y asignando al personal de soporte a otros problemas. Sin embargo, esto no siempre se refleja en el ahorro de dinero, algunas veces en la satisfacción del cliente. Si vemos que las tendencias muestran un incremento de las llamadas los lunes entre las 9 – 11 a.m., los clientes pueden ser atendidos mejor, si más personal está atendiendo sus requerimientos en esa hora.

### 3.3. Requerimientos para la implementación del software de un Help Desk.

Existen muchos software para *Help Desk*, no hay ninguno que sea la panacea. Para elegir el adecuado, se debe verificar que seguimiento se dará a las llamadas que se reciban. A continuación se listan algunas cosas que se deben considerar para buscar el mejor software para el *Help Desk* de la organización:

- **¿Es un software de uso interno?** En otras palabras, ¿el software será usado por los clientes en la propia compañía u organización?, si es así, algunas características que deberá tener, son más importantes que otras, por ejemplo, el *Help Desk* puede clasificar el tipo de preguntas que recibirá y el software del *help desk* debe permitir guardar los registros en la base de datos. La auditoría a PC's puede ser una característica que se debe considerar en la compra.
  
- **Similitud, si se utilizará el software con clientes externos,** Si la compañía ofrece sus servicios a otra organización, en donde la separación de cliente (una compañía puede ser un cliente) y contacto (una persona de la propia compañía que contacta a los clientes) es una importante consideración. Tener la base de conocimientos en línea (vía web), puede ser otra característica deseable en el *software*, que habilite el costo-eficiencia "solución ayuda-uno mismo" para los clientes.
  
- **¿Puede el software del *Help Desk* crecer junto con la compañía?** Esto es crucial. Las restricciones de ciertos *software* pueden impedir el crecimiento y forzar a comprar otro sistema. Se puede considerar un software que trabaje con múltiples bases de datos, en donde se pueda considerar cambiar fácilmente de Access a SQL Server a DB2 o a Oracle.
  
- **¿Qué sistema operativo y hardware utilizará el software para help desk?** Muchos software de *help desk* son diseñados para trabajar en un ambiente cliente/servidor.
  
- **¿Es compatible el software del *help desk* con el sistema de correo electrónico de la compañía?** Es importante tomar en cuenta que el software del *Help Desk* sea compatible con el sistema de correo electrónico de la empresa, si no se quiere invertir más dinero y tiempo adquiriendo otro o desarrollando la interface.

- **¿Es el software suficientemente flexible para adaptarse a las necesidades de la organización?** Por ejemplo, puede se soportado un departamento especial que demanda algunos campos propios en la base de datos de contactos.
  
- **Y por último, ¿Cuál es su costo?** Obviamente se quiere una software que permita realizar los procesos con la mejor calidad. Se necesita encontrar un balance entre el costo y la efectividad del *help desk* contra el *software* que solucionará todos o la mayor parte de los requerimientos.

Al contestar cada una de las preguntas anteriores y adicionando el análisis de los requerimientos para el *help desk*, se puede comprar el software correcto. También se debe leer los descubrimientos de los especialistas en *help desk*.

3.3.1. Ejemplo.

Actualmente se ofrecen muchos sistemas para Help Desk en el mercado, en este trabajo se maneja Action Request System de Remedy, pero mostraremos las características de otra aplicación para tener un punto de comparación:

**Support Wizard.**

<b>Descripción</b>	SupportWizard es una aplicación empresarial que automatiza el proceso de un nuevo ticket con un flujo de trabajo adaptable, notifica automáticamente vía correo electrónico al cliente todo con respecto a su ticket, los usuarios definen las reglas de escalación, reportes gráficos altamente adaptables en java y sofisticado control de acceso.
<b>Categoría :</b>	100 % basado en Internet
<b>Precio:</b>	\$23,500.00 Dlls.
<b>Licencias para usuarios:</b>	Ilimitadas.
<b>Licencias Técnicas:</b>	5
<b>Prueba libre, ¿por cuanto tiempo?:</b>	90 días
<b>Costo de actualización:</b>	Incluido en el contrato anual de soporte.
<b>Costo de soporte:</b>	En la compra de la aplicación incluye 90 días de soporte gratis, si se desea contratarlo posteriormente, el costo anual es de 15% sobre el valor actual del software.
<b>Se requiere instalar software local para el cliente:</b>	Un navegador estándar.
<b>Se requiere instalar software local para los administradores:</b>	No es necesario

<b>Servidores que lo soportan:</b>	<p>Unix</p> <p>Linux</p> <p>Windows NT</p> <p>Windows 2000</p>
<b>Requerimientos del servidor:</b>	<p>MySQL</p>
<b>Estaciones de trabajo en las que corre:</b>	<p>Unix</p> <p>Linux</p> <p>Windows NT</p> <p>Windows 2000</p> <p>Windows ME</p> <p>Windows 98</p> <p>Windows 95</p> <p>Windows 3.11</p> <p>Macintosh</p> <p>Cualquier máquina que soporte un navegador estándar.</p>
<b>Características:</b>	<ul style="list-style-type: none"> <li>• Búsquedas en la base de datos</li> <li>• Puede importar y exportar datos fácilmente a otras bases de datos.</li> <li>• Genera respaldos automáticos.</li> <li>• Base de conocimientos</li> <li>• Información del cliente en la base de datos</li> <li>• Escalaciones Automáticas</li> <li>• Reportes / Estadísticas</li> <li>• Historial del cliente</li> <li>• Notificaciones vía correo-electrónico.</li> <li>• La base de datos es automáticamente instalada y configurada con la instalación de Support Wizard.</li> </ul>
<b>Características customizables:</b>	<p>Campos de datos</p> <p>Plantillas</p> <p>Apariencia de la fuente</p> <p>Tablas</p>

	Fondos Imágenes de botones Grupo de usuarios y permisos
<b>Código fuente disponible:</b>	Si

Tabla 2.1. Características del software para Help Desk "Support Wizard".

### Action Request System (Remedy)

<b>Descripción</b>	<p>El AR System es una aplicación cliente/servidor. Puede ser accesado desde cualquier lugar en una red TCP/IP. Además, el AR System es muy flexible y permite definir el flujo de trabajo, adaptando el sistema a los procesos y reglas del negocio de la organización.</p> <p>AR System permite realizar cambios sin necesidad de programar para crear o actualizar aplicaciones, el desarrollo y la optimización es fácil y rápida.</p> <p>Maneja escalaciones y notificaciones, las cuales pueden ser vía radio, correo electrónico o utilizando el Notification Tool.</p>
<b>Categoría :</b>	Cliente/servidor
<b>Precio:</b>	Aplicación \$11,850.00 Dlls. Licencias \$ 1,000.00 Dlls. cada una, mínimo 5.
<b>Licencias para usuarios:</b>	3
<b>Licencias Técnicas:</b>	2
<b>Prueba libre, ¿por cuanto tiempo?:</b>	No cuenta con prueba libre.
<b>Costo de actualización:</b>	Incluido en el contrato anual de soporte.
<b>Costo de soporte:</b>	El costo anual es de 10% sobre el valor actual del software.
<b>Se requiere instalar software local para el cliente:</b>	Sí, el AR User

<b>Se requiere instalar software local para los administradores:</b>	Si, el AR Admin (sólo en el cliente que utilizará el administrador).
<b>Servidores que lo soportan:</b>	Windows NT Windows 2000 Solaris HP-UX AIX
<b>Requerimientos del servidor:</b>	Oracle Informix Sybase Microsoft SQL Server DB2 for Windows NT Flat File (es la base de datos que trae por default AR System)
<b>Estaciones de trabajo en las que corre:</b>	Windows 95 Windows 98 Windows NT Solaris HP-UX AIX Silicon Graphics Web Browser
<b>Características:</b>	<ul style="list-style-type: none"> <li>• Búsquedas en la base de datos</li> <li>• Puede importar y exportar datos fácilmente a otras bases de datos.</li> <li>• Base de conocimientos</li> <li>• Información del cliente en la base de datos</li> <li>• Escalaciones Automáticas</li> <li>• Reportes / Estadísticas</li> </ul>

	<ul style="list-style-type: none"> <li>• Historial del cliente</li> <li>• Notificaciones vía correo-electrónico, radio o notificaciones internas.</li> <li>• La base de datos es totalmente manejada por Action Request System, por lo que es transparente para el usuario.</li> <li>• Importación y exportación de datos.</li> <li>• Altamente adaptable al flujo de trabajo de la organización.</li> <li>• Puede implementarse en un servidor WEB.</li> </ul>
<b>Características customizables:</b>	<p>Campos de datos</p> <p>Plantillas</p> <p>Apariencia de la fuente</p> <p>Tablas</p> <p>Fondos</p> <p>Imágenes de botones</p> <p>Grupo de usuarios y permisos</p>
<b>Código fuente disponible:</b>	Si

Tabla 2.2. Características del software para Help Desk "Action Request System", de Remedy.

### Comparación.

Como se puede observar, las dos aplicaciones a pesar de que son muy similares, se tienen que estudiar con cuidado para poder elegir cual se utilizará. En este trabajo no se va a tomar una decisión, porque el software ya fue comprado por la empresa, en este caso se eligió Action Request System, de Remedy, a pesar de ser más caro (se tiene que pagar cada licencia que se utiliza, ya sea de usuario o técnica), la empresa tiene un acuerdo con Remedy y le ofrece un precio especial. También hay que tomar en cuenta que Action Request System soporta un mayor número de bases de datos, puede trabajar como cliente/servidor, a través de TCP/IP, además de contar con un módulo que le permite trabajar en Web. En cambio, el Support Wizard es 100% para trabajar en Web, incrementando el costo en el servidor, aunque disminuye el costo del mantenimiento. Sólo trabaja con SQL como base de datos. También es importante considerar que uso se le dará, tal vez los procesos de la empresa son muy específicos y se tiene que customizar, o son muy generales y se puede utilizar tal cual. En el caso del Centro de Ayuda para una tienda de autoservicio es muy específico y se tienen que hacer varias modificaciones y Action Request System ofrece gran adaptabilidad y rapidez en el desarrollo de aplicaciones.

## CAPÍTULO 4

### ACTION REQUEST SYSTEM.

#### 4.1. REMEDY

Remedy® Corporation (RMDY) es un proveedor líder en software para soluciones de Tecnología de la Información en Administración de Servicios (Information Technology Service Management (ITSM)) y Manejo de las Relaciones con el Cliente (Customer Relationship Management (CRM)). Desde su fundación en 1990, miles de compañías alrededor del mundo han utilizado sus aplicaciones empresariales de fácil implementación para guiar los procesos de su negocio, mejorando su toma de decisiones. Las soluciones de Remedy ofrecen un rápido regreso de la inversión.

Al cierre del año 2000, más de 9,850 sitios<sup>6</sup> de clientes están en producción utilizando los productos de Remedy. De hecho, muchas organizaciones han desarrollado sus propias aplicaciones ITSM y CRM, utilizando el AR System. El éxito de esos productos en el mercado a lo largo de los años, han dado a Remedy una reputación como una compañía sólida y altamente confiable.

Actualmente se busca que los sistemas sean rápidos, eficientes, flexibles, seguros y provechosos. El software de Remedy ayuda a conseguir estos objetivos. Los profesionales en IT (Information Technology) utilizan Remedy para la rápida implementación de Help Desk's (Centros de Ayuda) que son más confiables, eficientes y medibles, con poca gente. Los productos de Remedy reducen costos porque administran el ciclo de vida completo de los activos de la empresa. y mejoran los procesos de administración que abarcan el crítico sistema de recursos de la compañía. De manera similar, las ventas, la publicidad y el soporte a usuarios, utilizan este software para permanecer cerca

---

<sup>6</sup> Información tomada de la página [www.remedy.com](http://www.remedy.com)

de sus clientes, y administrar todos los procesos necesarios para mantener al cliente satisfecho.

El Software de Remedy siempre ha sido conocido por su fácil uso y su notable adaptabilidad. Es decir, que habilita al cliente para reaccionar rápido y adaptarse lo mejor posible a los constantes cambios que son parte del mundo de negocios actual. También, las soluciones de Remedy, pueden ser implementadas en una empresa mucho más rápido que las soluciones ITSM y CRM de otras compañías.

En México, también muchas empresas han adquirido los productos de Remedy, de las cuales podemos mencionar las siguientes : IBM, S.A. DE C.V., GIGANTE, COMERCIAL MEXICANA, GRUPO WAL-MART, etc.

La actual incorporación con Peregrine System<sup>7</sup>, es la confirmación de que AR System continuará por un camino innovador, al sumar sus productos con los de Peregrine, Remedy asegura un acelerado crecimiento.

#### **4.2. ACTION REQUEST SYSTEM (AR SYSTEM)**

El Action Request System (AR System) es una aplicación cliente/servidor altamente adaptable a un ambiente de negocios que cambia continuamente. El AR System permite construir rápidamente un prototipo, su desarrollo, su mantenimiento y su continua optimización.

Por ejemplo, en el caso de un Help Desk o Centro de Ayuda, un administrador puede utilizar el AR System para crear un sistema flexible y adaptable al flujo de

---

<sup>7</sup> Una empresa líder en el mercado, que ofrece soluciones para la administración del ciclo de vida de los activos en una organización, de los equipos de Tecnología de la Información y de sus activos intangibles de conocimiento, así como soluciones de comercio electrónico, entre muchas cosas más.

los procesos de la organización. Cuando se introduce información en AR System, se captura una base de datos de problema/solución, creando una base de conocimientos en base a la experiencia. También se puede registrar eficientemente una variedad de activos, información y problemas.

Un objetivo del AR System es el de proveer al usuario de una interface de fácil uso para administrar requerimientos de soporte, entonces cualquier miembro del grupo de soporte o usuario final puede usar las herramientas de AR System con un mínimo aprendizaje.

#### **4.2.1 BENEFICIOS DEL AR SYSTEM**

El AR System es una multiplataforma, una aplicación cliente/servidor. Se puede adquirir el AR System Server para cualquier estación de trabajo con plataforma UNIX y Microsoft Windows NT en Intel. AR System Cliente puede estar en otras estaciones de trabajo o sistemas Windows NT, también corre en PC's Microsoft Windows, computadoras Macintosh o terminales ASCII, por lo que puede ser compatible con el hardware existente en la organización.

Para un rendimiento más alto, al instalar AR System, se puede utilizar una nueva base de datos o una ya existente en Oracle o Microsoft SQL Server bajo Windows NT, o Sybase, Oracle, Informix o Ingres bajo UNIX, preservando los valores de la base de datos

También, el AR System puede ser accesado desde cualquier lugar en una red TCP/IP. Además, el AR System es muy flexible y permite definir el flujo de trabajo, adaptando el sistema a los procesos y reglas del negocio de la organización.

El AR System separa cliente, servidor y base de datos, lo cual facilita la actualización de cada uno de ellos.

En la actualidad, el ambiente de negocios es muy cambiante, AR System permite realizar cambios sin necesidad de programar para crear o actualizar aplicaciones, el desarrollo y la optimización es fácil y rápida. Los cambios cuestan menos tiempo y esfuerzo, y se pueden desarrollar directamente en el ambiente operacional.

Las aplicaciones en AR System se implementan rápidamente, haciendo evidente un regreso de la inversión en el más valioso activo: la gente. Cada usuario no técnico, puede utilizar fácilmente la aplicación, porque AR System ofrece una interface gráfica para el usuario, minimizando los costos de capacitación y acelerando la productividad. Un nuevo administrador puede hacer cambios rápidamente cuando se requiera, minimizando costos.

Todo acerca de la interface del usuario de AR System es diseñado para optimizar su productividad, sus características guían rápidamente al usuario nuevo u ocasional a través del sistema con los pasos correctos. A través de ventanas se presenta un complemento de información concurrente, reduciendo errores y ahorrando tiempo.

A partir de una rápida y simple creación gráfica de aplicaciones, que permite automatizar el flujo de trabajo, re-utilizar objetos y controlar un robusto código fuente de interface, AR System ofrece las herramientas para crear rápidamente un mundo de soluciones para cualquier cosa que se requiera.

Finalmente, el AR System ofrece una API (Application Programming Interface) abierto, permitiendo que el usuario escriba sus propios programas de interface con el servidor de AR System, logrando una mejor integración de datos en la organización.

### **4.3. ARQUITECTURA.**

La arquitectura que utiliza el Action Request System es una arquitectura cliente/servidor, misma que se explico en el capítulo 2, en esta parte del capítulo se explica a detalle los requerimientos específicos de la herramienta.

#### **4.3.1. ARQUITECTURA CLIENTE / SERVIDOR.**

El AR System usa una arquitectura cliente – servidor. Puede correr en una sola estación de trabajo, conectada a una red TCP/IP. También corre en diferentes plataformas, incluyendo Sun, HP, IBM, NCR, SGI, Motorola, y cualquier PC con sistema capaz de correr Windows NT Server 3.51 o superior.

El AR System envía requerimientos de datos y transacciones al servidor. Todas las herramientas del AR System, incluyendo el Administration Tool, son clientes del AR System Server.

Los usuarios con estaciones de trabajo UNIX, PC's MS Windows, o computadoras Macintosh pueden correr un User Tool que es nativo de sus plataformas.

Utilizando el modelo cliente – servidor, se obtienen muchos beneficios, por ejemplo, el mantenimiento y la administración de los datos pueden ser centralizados en un servidor. El front-end (la parte que visualiza el usuario) del AR System cliente es distribuido en los sistemas del cliente, y cada usuario utiliza la interface nativa de sus plataformas particulares.

**4.3.2. REQUERIMIENTOS DEL SISTEMA.**

Para instalar y correr el AR System en un servidor UNIX, se requieren el siguiente espacio en disco duro.

UNIX Workstation	Espacio en Disco			
	Servidor	Cliente	HTML	API
NCR System 3300	26MB	24MB	11MB	4.0MB
HP 9000 Series 700/800	43MB	27MB	11MB	6.0MB
IBM RS/6000	42MB	26MB	11MB	3.0MB
SGI IRIS	34MB	32MB	11MB	4.0MB
Sun SPARC System	52MB	36MB	11MB	5.0MB

Tabla 4.1. Espacio en disco duro requerido para AR System.

Para instalar y correr el AR System en un servidor NT, se requiere el siguiente equipo:

- Una PC modelo 486 o superior
- Un monitor con una resolución en pantalla mínima de 640 x 480 (VGA) que soporte 16 colores. 800 x 600 o superior es altamente recomendable.
- Un mínimo de 20MB de espacio en disco duro disponible después de la instalación. Y durante la instalación se debe tener un mínimo de 40MB de espacio disponible.

El cliente se puede instalar en cualquiera de los siguientes ambientes de sistema operativo:

- Windows 95
- Windows 98
- Windows NT
- Solaris
- HP-UX
- AIX
- Silicon Graphics
- Web Browser

El servidor puede tener como plataforma:

- Windows NT
- Windows 2000
- Solaris
- HP-UX
- AIX

La base de datos que puede manejar AR System puede ser:

- Oracle
- Informix
- Sybase
- Microsoft SQL Server
- DB2 for Windows NT
- Flat File (es la base de datos que trae por default AR System)

Las plataformas de Administración de Redes y Systemas compatibles con Remedy son :

- BCM PATROL
- Boole & Babbage COMMAND/Post
- Bullsoft OpenMaster and System Manager
- Cabletron SPECTRUM
- HP OpenView
- IBM NetView for AIX
- Management Microsoft SMS
- OSI NetExpert
- Sun Solstice
- Tally NetCensus
- Tivoli Management Environmet

### 4.3.3. CAPAS DEL AR SYSTEM SERVER.

El AR System server está compuesto por diferentes capas, como se ilustran en la siguiente figura:

Ingreso de transacciones y requerimientos de datos.

Remedy Action Request System Server (Windows

arservr (UNIX)

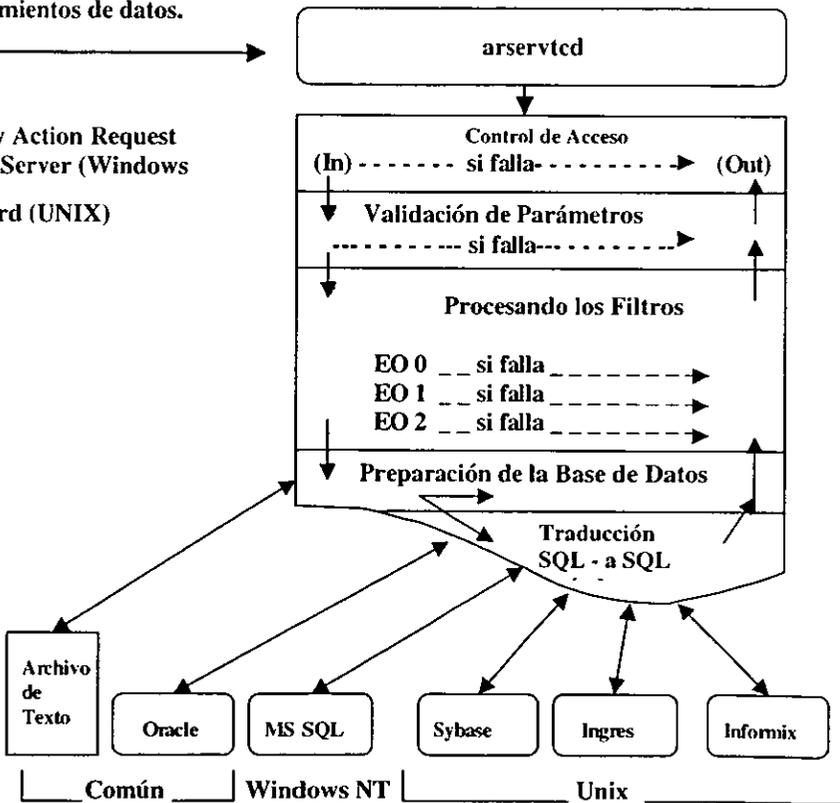


Figura 4.1. Capas del Servidor de AR System.

Las transacciones de datos del AR System utilizan RPC's (Remote Procedure Calls) para el API del AR System.

La **capa de control de acceso**, ejecuta revisiones de validación, incluyendo información del usuario, los permisos de accesos, por ejemplo, en campos específicos dentro de la transacción.

La **capa de parámetros de validación** verifica que los tipos de datos y límites del campo (incluyendo rangos numéricos) sean correctos para el valor dado, también que los campos obligatorios tengan los valores solicitados.

La transacción entonces pasa a través de la **capa de procesamiento del filtro**, la cual contiene todos los filtros definidos por el administrador. Los filtros seleccionan las transacciones que cumplen con criterios específicos y las características de la respuesta esperada.

La última capa, la **interface de la base de datos** da los comandos apropiados para cualquier base de datos que este utilizando.

#### 4.3.4. PROGRAMAS Y SERVICIOS DEL AR SYSTEM.

El AR System incluye los siguientes servicios o procesos .

- **Remedy Action Request System Server (arservtcd and arserverd):** Los procesos del servidor de AR System. El servidor de AR System tiene los procesos **arservtcd** y **arserverd**. El proceso **arservtcd** es el " administrador de procesos " el cual recibe y direcciona todas las transacciones y los requerimientos de datos de los clientes. El proceso **arserverd** es el actual servidor de datos del AR System. Pueden existir más de un **arserverd** en una máquina.
  
- **Remedy Notification Server (ntclientd and ntserverd):** El servicio que soporta el Notification Tool.
  
- **armaild:** El proceso que habilita al usuario para registrar reportes (ARs) o hacer listados de los mismos utilizando los recursos de su correo electrónico.
  
- **arnvd, arovd y arsnmd:** Estos procesos proveen de una interface entre el AR System e IBM's NetView/6000, HP's OpenView, and Sun Microsystems' SunNet Manager. Estos procesos no necesitan correr en la misma plataforma que el AR System Server.

### 4.3.5. SERVIDORES DEL AR SYSTEM

Para mejorar el desempeño del AR System, se pueden correr más de un servidor de AR System. Al utilizar la Opción de Multi-Procesamiento, se mejora el rendimiento y el desarrollo por la distribución de carga. Esta opción es escalable desde un servidor que realiza funciones simples, hasta múltiples servidores que realicen funciones específicas. Los servidores se adaptan a la configuración de parámetros definida y a la distribución de carga.

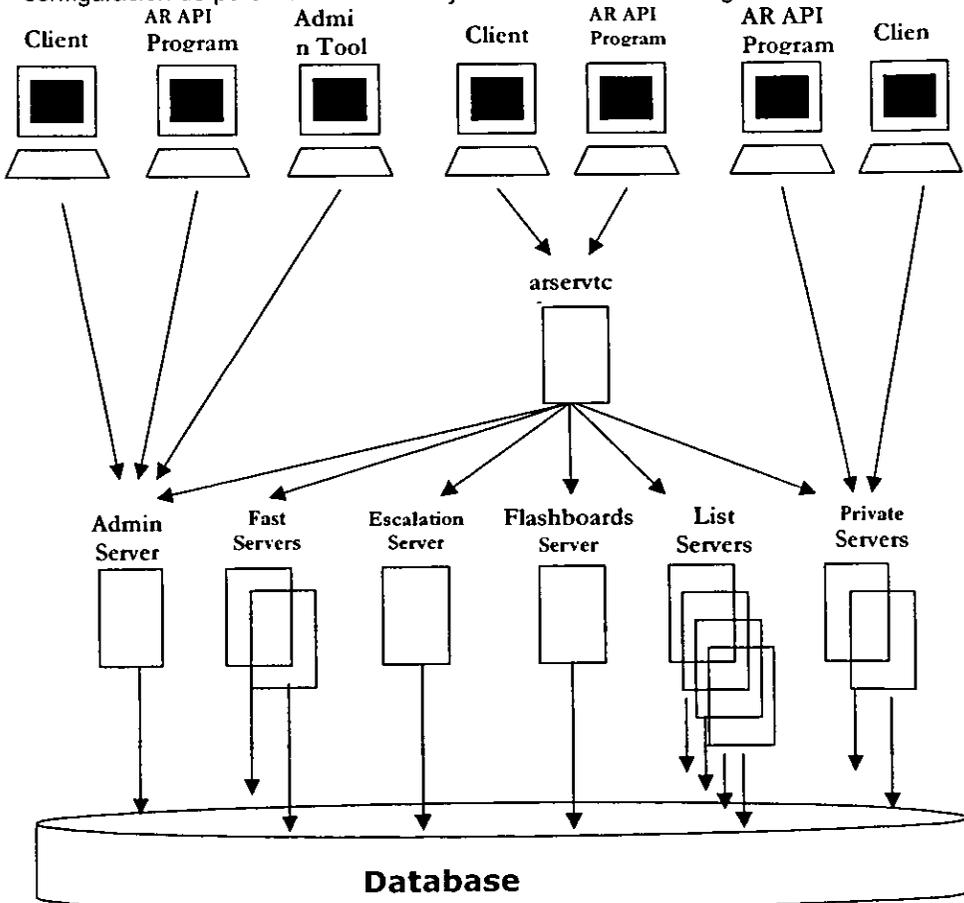


Figura 4.2. Ilustra como el controlador del servidor `arservtc` maneja los requerimientos de los clientes y los administradores de los servidores del AR System en un ambiente de servidor de multi-procesamiento.

Se pueden correr tantos servidores como sean necesarios. Hay seis tipos de servidores:

- **Admin Server.** El server administrador es un servidor del AR System que puede desempeñar cualquier operación dentro del sistema. Este realiza todas las operaciones de administración, garantizando la integridad de toda la estructura.
  
- **Fast Servers.** Los Fast Servers manejan todas las operaciones inmediatas del AR System, es decir, las operaciones que generalmente se completan rápidamente sin el bloqueo de acceso a la base de datos. Por ejemplo, las operaciones del User Tool que usan un fast server incluyen lo siguiente:
  - Los botones de Previo y Siguiente en las ventanas de Ver y Modificar.
  - Al generar reportes.
  - Generar listas de esquemas y campos.
  
- **List Servers.** Los List Servers manejan las operaciones del AR System que pueden tomar algún tiempo porque ellos tienen bloqueado el acceso a la base de datos: ARExport, ARGetListEntry y ARGetEntryStatistics. Las operaciones del User Tool que usan los list servers son las siguientes:
  - Consultas
  - Conexión inicial a un esquema.
  
- **Escalation Server.** Por lo general, el admin server maneja los escalamientos. Si se crean y manejan muchas escalaciones con intervalos cortos, es recomendable utilizar un servidor que solo maneje las escalaciones, de lo contrario se puede afectar el desempeño del admin server.

- **Flashboards.** El Flashboards server corre automáticamente si se cuenta con una licencia de servidor de Flashboards. Se utiliza para monitorear y representar gráficamente las tendencias del AR System. Las tendencias son los modelos que proporcionan la información acerca del uso de recursos del servidor en una particular hora o día. Es útil para entender los tipos y el número de operaciones que se están ejecutando en el sistema.
  
- **Private Servers.** Para los servidores fast, list y escalation, el administrador también puede crear servidores privados para usuarios específicos que tengan necesidad de accesos dedicados. Por ejemplo, un usuario que realiza operaciones críticas que no quiere ser bloqueado por otros usuarios. Los números de RPC socket usados por los servidores de multiprocesamiento no se traslapan con los números de socket utilizados por los private servers. Los usuarios no conectados específicamente a un servidor privado no puede acceder al mismo.

#### 4.4. LA ESTRUCTURA DE LA BASE DE DATOS

El AR System oculta la base de datos al usuario, el interactúa con la misma y le proporciona la información que requiere. El Remedy soporta el acceso para lectura directamente de las tablas, pero no soporta el acceso para actualizar cualquier tabla del AR System directamente de SQL, esto se debe hacer por medio de un API.

Durante la instalación, una nueva base de datos llamada ARSystem es creada. Se solicita el tamaño de la base de datos y el nombre del dispositivo en el cual se alojara. Por default, una base de datos de 10 MB se crea. También se genera un log space, en el cual se registraran las transacciones realizadas en la base de datos, se tiene que indicar en que dispositivo se localizara. Remedy recomienda que los datos y el log se almacenen en dispositivos separados en la base de datos, y los dispositivos deberán estar en discos físicamente separados, para mejor desempeño y recuperación.

Un nuevo usuario llamado ARAdmin también es creado. Todas las tablas creadas y accesadas por el AR System, son creadas por el usuario ARAdmin y son localizadas en la base de datos ARSystem. Lo procesos de Remedy dependen de este usuario, si se modifica o se borra, no funcionarán adecuadamente.

Después de crear la base de datos y el usuario ARAdmin, Remedy crea una serie de tablas que forman el diccionario de datos del AR System. Esas tablas contienen la definición estructural de todos los esquemas, filtros, escalaciones, ligas activas, menús y extensión del administrador que son ingresadas en el sistema.

#### 4.5. OBJETOS.

El AR System cuenta con los siguientes objetos para el diseño de la aplicación :

- **Esquemas.** Cada esquema representa una tabla en la base de datos en el AR server, define como se almacenará la información en la base de datos. Todos los esquemas tienen un conjunto de campos obligatorios y se pueden adicionar tantos campos como sea necesario.
- **Menús.** Proporcionan listas ordenadas de las opciones para una selección rápida. Cada menú tienen un nombre específico y un conjunto de opciones que se define utilizando el Administrator Tool. Se pueden definir los siguientes estilos de menú:

Character. Se define el menú a través de los valores ingresados en campos de la ventana de Menú.

File: Se especifica un archivo que contiene el formato del menú.

Query: Se especifica una búsqueda que permita traer valores de otro esquema.

SQL: Se especifica un comando SQL que permita traer valores de una base de datos relacional.

Los menús son definidos y almacenados independientemente de los esquemas, lo cual permite definir un solo menú y utilizarlo donde sea en cualquier esquema.

- **Active Links.** Permiten el establecimiento de ligas entre datos contenidos en diferentes esquemas. Son más frecuentemente utilizados para cargar datos dentro del esquema actual de una fuente alterna, como un esquema secundario o una base de datos diferente. Son ejecutados en la maquina cliente. Se ejecutan por una acción del usuario, son asociados con un solo

esquema y se pueden especificar condiciones que se deben cumplir antes de ejecutarse.

- **Filtros.** Son un mecanismo de implementación en el proceso del flujo de trabajo, estos pueden ser utilizados para forzar las reglas del negocio. Los filtros prueban cada transacción del servidor con condiciones específicas y, si estas condiciones se cumplen, el servidor realiza una o más acciones. Los filtros son ejecutados en el servidor del AR System. Los procesos del flujo de trabajo son parcialmente definidos en los esquemas, pero también se definen por los filtros, los escalamientos y las ligas activas que se crean.
- **Escalaciones.** Las escalaciones comparan cada entrada con específicas condiciones en un intervalo definido. Las escalaciones funcionan de forma similar a los filtros, excepto que estas ocurren en un punto específico dentro de un intervalo de tiempo. También definen que acciones tomará el servidor cuando una transacción cumpla ciertas condiciones. Si las condiciones se cumplen, una o más acciones se realizan por el servidor.
- **Macros.** Ayudan a automatizar la ejecución de frecuentes operaciones, como por ejemplo, al crear reportes diarios o la presentación de una consulta de los reportes de alta prioridad. Cualquier usuario puede crear macros.
- **Interfaces:** Procesos externos, DDE, Notificaciones y API's del AR System.

#### 4.6. CONTROL DE ACCESO.

Una vez que se han identificado los usuarios de la aplicación del AR System y se ha identificado el tipo de información que cada uno necesita acceder, se pueden configurar los usuarios y establecer sus accesos al sistema. También se pueden definir grupos con varios usuarios

El control de acceso es un mecanismo que se utiliza para tener seguridad en los *tickets* del AR System, controlando quien puede crear, ver, modificar o borrar los *tickets*.

El control de acceso del AR System permite definir grupos de usuarios y determinar cuales de estos grupos tienen acceso a cual esquema y a cual campo dentro de un esquema.

Un usuario es una persona que tiene permisos para acceder al AR System. Los usuarios pueden ser miembros de varios grupos o pertenecer solo a uno. Los grupos son una colección de usuarios del sistema, se pueden definir en base a perfiles comunes y en conformidad con los accesos asignados.

Los grupos tienen varias funciones importantes:

- Los grupos son la base en donde todos los tipos de acceso son concedidos.
- Los grupos son importantes cuando se crean los objetos y el flujo de trabajo, por ejemplo, cuando se crea un nuevo esquema, se definen que usuarios tendrán permiso para acceder, esto en base al grupo que pertenezca.
- Los grupos definen las notificaciones que los miembros del mismo pueden recibir.

**4.6.1. Grupos explícitos e implícitos.**

Hay dos tipos de grupos:

1. **Grupos explícitos.** Son los grupos en los cuales los usuarios deben de ser explícitamente asignados.
2. **Grupos implícitos.** Son grupos definidos por circunstancias y situaciones específicas. Los usuarios pertenecen a estos grupos en base a condiciones específicas, no se pueden asignar usuarios a esos grupos.

**4.6.2. Grupos Reservados.**

Grupo	ID	Descripción
<b>Public</b>	0	Definición general de accesos de lectura o cambios en campos o esquemas. Para cualquier campo o esquema donde el grupo <b>Public</b> tenga acceso de lectura, los usuarios pueden ver el contenido de los campos del <i>ticket</i> sin importar a que grupo pertenezcan y que privilegios tengan. El grupo <b>Public</b> es especial: todos los usuarios son miembros del grupo <b>Public</b> automáticamente, aunque no estén en la lista del esquema de usuarios.
<b>Administrator</b>	1	Define a los usuarios que tienen todos los accesos al AR System. Los usuarios que pertenecen a este grupo, pueden crear, modificar y borrar objetos, incluyendo <i>tickets</i> , esquemas, menús, filtros, escalaciones y ligas activas. Ellos también pueden agregar y borrar usuarios y otros grupos.  Los usuarios asignados para ese grupo automáticamente tiene acceso para realizar cambios a todos los campos y tiene acceso a todos los sistemas.

		Cada miembro del grupo Administrator, debe tener una licencia <b>fixed</b> .
<b>Customize</b>	2	Garantiza a los usuarios los derechos de desarrollar su plan de esquemas y crear sus comandos en el User Tool. Este grupo es usado para limitar cada habilidad del usuario para cambiar los esquemas. Los administradores siempre han tenido permisos y no necesitan ser asignados a este grupo.
<b>Submitter, Assignee</b>	3, 4	Estos son grupos implícitos usados para controlar el acceso basado en el <i>login</i> del usuario. Un usuario pertenece al grupo <b>Submitter</b> automáticamente por los <i>tickets</i> que tienen su nombre en el campo de submitter. Un usuario pertenece a el grupo <b>Assignee</b> automáticamente por los <i>tickets</i> que tienen el nombre del usuario en el campo de Assigned-to.
<b>Subadministrator</b>	5	Ofrece un acceso limitado a la administración del AR System, permite realizar diferentes funciones administrativas en un solo servidor de AR System. Los usuarios que pertenecen a este grupo, tiene un acceso limitado a un conjunto de estructuras. Cada miembro del grupo Subadministrator, debe tener una licencia <b>fixed</b> .
<b>Assignee Group</b>	7	Este es un grupo implícito usado para el control de accesos en cada miembro del grupo de usuarios. Un usuario pertenece a este grupo, cuando los <i>tickets</i> en los cuales el campo de Assignee Group existe y contiene el grupo al que pertenece.

Tabla 4.2. Grupos Reservados.

### 4.6.3. LICENCIAMIENTO.

Los licenciamientos afectan el control de acceso. Cuando se crean los usuarios, se deben de asignar a cada uno un tipo de licencia. Controlando así las actividades que pueden realizar los usuarios.

#### 4.6.3.1. Tipos de Licencias.

El tipo de licencia que se asigne al usuario controla el nivel de acceso que tenga al AR System.

Licencia	Descripción
<b>Read</b>	Da acceso para buscar y desplegar <i>tickets</i> existentes y para registrar nuevos <i>tickets</i> , basadas en el grupo a cual el usuario pertenece. No puede modificar o salvar datos en <i>tickets</i> existentes.
<b>Fixed Write</b>	Proporciona los mismos accesos que la licencia <b>read</b> , y también da acceso para modificar y salvar datos para un <i>ticket</i> ya existente, basado en el grupo al cual pertenece el usuario. El administrador del AR System debe tener esta licencia, así como los usuarios que frecuentemente necesitan modificar los <i>tickets</i> .

<p><b>Floating Write</b></p>	<p>Proporciona los mismos accesos que la licencia <b>fixed write</b>, con la diferencia de que la licencia <b>floating write</b> puede adaptarse a las necesidades de acceso de escritura ocasional dentro del AR System. Estas licencias son diseñadas para permitir el acceso infrecuente para modificar o salvar datos sin necesidad de una licencia <b>Fixed Write</b>.</p>
------------------------------	---

Tabla 4.3. Tipos de Licencias.

#### 4.7. USUARIOS DEL AR SYSTEM

Cuatro tipos de usuarios están involucrados con la operación de AR System:

1. **Administrador.** Es el responsable de la administración del AR System. El define la base de datos de AR System, los derechos de acceso a los usuarios y designa el flujo de trabajo (workflow) que siguen los procesos.
2. **Subadministrador.** Es el responsable de la configuración, diseño y derechos de acceso que se darán a un grupo de esquemas permitidos por el administrador.
3. **Grupo de Soporte.** Resuelven los *tickets* utilizando el AR System configurado por el Administrador. Ellos asignan y son asignados a *tickets*, y registran sus avances en los campos apropiados. Los miembros del grupo de soporte pueden utilizar la información almacenada en anteriores *tickets*, en caso de que la falla ya se halla presentado, se puede tomar como referencia la solución que se le dio, se evita redundar esfuerzos y se dar una pronta solución al *ticket* actual.
4. **Usuarios.** Generan los *tickets* con problemas y requerimientos de soporte.

#### 4.8. COMPONENTES DEL AR SYSTEM.

El Action Request System provee de cuatro herramientas que ayudan al administrador, al grupo de soporte y a los usuarios a desarrollar sus trabajos.

- User Tool
- Notification Tool
- Administrator Tool
- Import Tool

➤ **USER TOOL.** Es generalmente utilizado por cada uno de los involucrados en el flujo del proceso. Sus funciones son :

- Registra *tickets* (ARs) describiendo problemas y requerimientos de soporte.
- Obtiene listados de la base de datos de AR System, de cualquier información o datos de registros previos.
- Modifica *tickets* previamente registrados
- Realiza reportes y cálculos estadísticos de la información de los *tickets*.
- Borra *tickets* (si tu eres administrador o subadministrador).
- Crea y corre macros para ayudar a automatizar los pasos para realizar reportes complejos y otras operaciones.
- Adapta vistas de esquemas.
- Define valores por defecto en determinados campos.

➤ **NOTIFICATION TOOL.** Al igual que el User Tool, el Notification Tool puede ser usado por cualquier persona involucrada en el flujo del proceso. El Notification Tool alerta a los usuarios de los adelantos que se llevan en la solución de su *ticket*. Los usuarios pueden ser notificados de cualquier cambio en un *ticket*.

Por ejemplo, el administrador puede haber especificado que una notificación será enviada al grupo de soporte cuando nuevos *tickets* se les asignan y también especificar que usuarios serán notificados cuando el *ticket* es cerrado.

➤ **ADMINISTRATOR TOOL.** El Administrator Tool es usado exclusivamente por el administrador para configurar el AR System que será utilizado por el grupo de soporte y los usuarios. Esto incluye lo siguiente:

- Creación de esquemas (las definiciones de la base de datos del AR System)
- Creación de accesos en esquemas.
- Creación de permisos de acceso en los campos dentro de los esquemas.
- Creación de filtros, escalaciones y ligas activas que definen el flujo de los procesos.

➤ **IMPORT TOOL.** Es usado para importar datos y registros de otro servidor AR System o de otras aplicaciones dentro del servidor de AR System. Para importar datos dentro de un esquema, se debe tener permisos para modificar los campos que se desean importar. El Import Tool tiene las siguientes funciones:

- Permite importar datos de una gran variedad de fuentes (bases de datos u hoja electrónica) y formatos (\*.csv, \*.asc y \*.arx).
- Permite mapear fuentes de datos para especificar los campos en los esquemas, aún si las fuentes de datos tienen diferentes nombres de los campos.

- Permite el manejo interactivo de los registros dañados basado en un registro-por-registro con su sofisticada capacidad para el manejo de errores.

#### 4.9. ANÁLISIS DE WORKFLOW

El Workflow o flujo de trabajo sirve para automatizar procesos de negocios con políticas y procedimientos definibles, repetitivos y claramente determinados. Al ser una herramienta workflow, el AR System se puede aplicar en donde hay frecuentes eventos, los eventos se mueven a través de estados, y las entradas son ingresadas y trabajadas por un grupo de personas. Se pueden definir los esquemas para cualquier aplicación. Se pueden crear tantos esquemas como se requieran. El uso que se de al AR System, depende de muchos factores que son específicos de las necesidades de cada empresa.

#### 4.9.1. Diseño de una aplicación en AR System.

El AR System considera dos modelos de desarrollo, el Modelo de Cascada (Waterfall Model) y el Modelo de Desarrollo Incremental (Incremental Development).

**El Modelo de Cascada.** El tradicional modelo de desarrollo de cascada, considera las especificaciones originales del usuario, pero olvida los requerimientos de cambios. Por lo que este modelo, no es muy flexible para la retroalimentación del usuario.

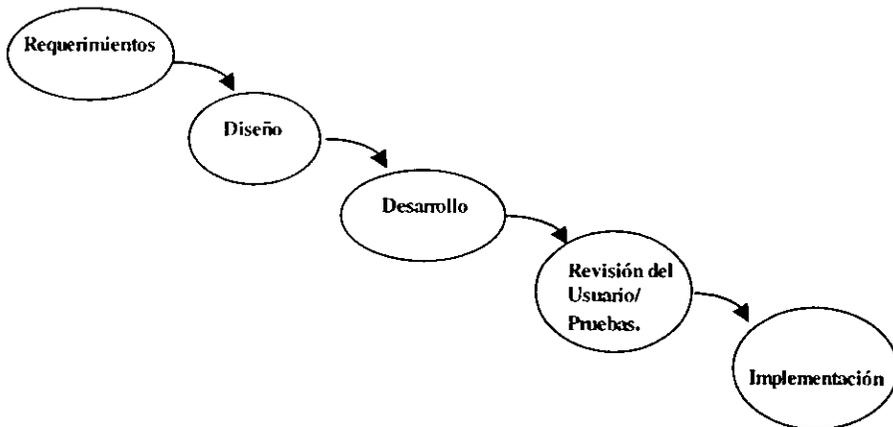


Figura 4.3. Modelo de Cascada.

**El Modelo de Desarrollo Incremental.** El modelo de desarrollo incremental, permite que se consideren las especificaciones del cliente y también adaptar la necesidad de cambios del usuario final, ofreciendo una mejor solución a los usuarios.

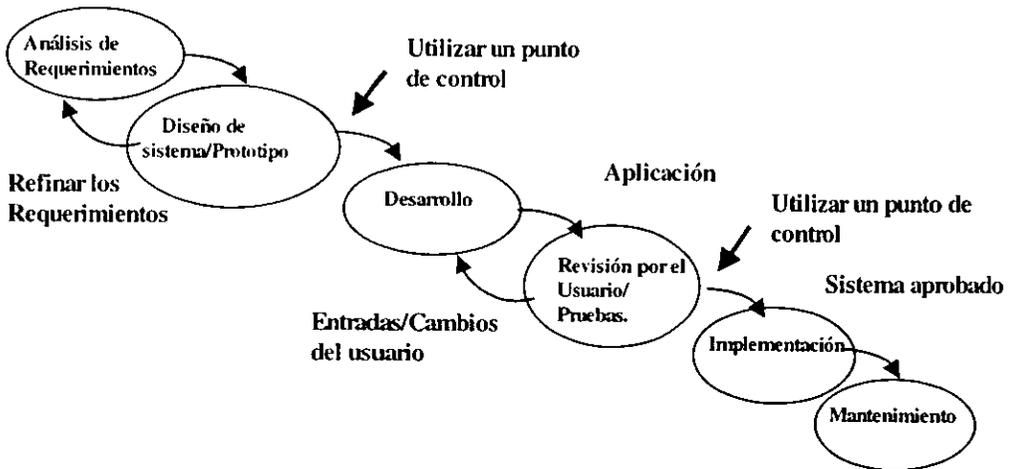


Figura 4.4. Desarrollo Incremental.

El AR System permite ofrecer una solución rápidamente, con el modelo de Desarrollo Incremental todas las decisiones no se tienen que tomar inmediatamente, por supuesto que los requerimientos deben ser claramente establecidos, pero se puede mejorar la solución después; también permite una continua retroalimentación y sugerencias de los usuarios, permitiendo que la aplicación cumpla con los requerimientos del usuario final.

#### 4.9.2. Pasos básicos para el desarrollo de una aplicación en AR System.

Hay varias facetas para el desarrollo de una aplicación en AR System.

- Definir el proceso.
- Identificar los puntos de control o estados del proceso.
- Identificar los roles involucrados.
- Identificar las mediciones o reportes.
- Definir los elementos de los datos (esquemas y campos) e interfaces externas.

#### Definir el Flujo de Trabajo.

El flujo de trabajo (workflow) es el proceso por el cual tareas individuales son coordinadas para completar una transacción (usando un proceso de negocios claramente definido) dentro de una organización. Es indispensable tener un conocimiento claro del flujo de trabajo antes de crear una aplicación.

Las preguntas que se tienen que hacer acerca del flujo de trabajo son las siguientes:

- ¿En dónde empieza el requerimiento?, ¿en dónde termina?, ¿qué sucede en el intermedio?. En otras palabras, ¿cuál es el ciclo de vida del requerimiento?.
- ¿Qué pasa a través de cada etapa de un requerimiento?
- ¿Qué acciones se deben tomar dentro de la aplicación durante el tiempo de vida del requerimiento.?

Tiene que tomar en cuenta lo siguiente:

- Los usuarios o el grupo de soporte deben ser habilitados para registrar los requerimientos rápida y completamente.
- Los requerimientos deben ser atendidos por el grupo de soporte inmediatamente.
- El flujo de trabajo debe ser capaz de reasignar los requerimientos rápidamente.
- Los requerimientos se tienen que solucionar lo antes posible, el sistema debe recordar a la persona asignada cuales continúan sin solución.
- El proceso debe tener la habilidad de seguir todos los pasos del requerimiento a través del sistema.
- El usuario final debe ser notificado inmediatamente de la solución del requerimiento.
- El proceso debe soportar la construcción de una base de conocimiento.

#### **Identificar los estados del requerimiento.**

Los estados son la clave para mover detalladamente los requerimientos a través del sistema. Los estados deben de ser la piedra angular del proceso del flujo de trabajo, por ejemplo:

- Nuevo
- Asignado
- Atendido
- Suspendido
- Cerrado

El campo de status debe registrar los estados importantes del requerimiento, desde que se recibe hasta que se cierra. Debe haber un estado para cada estado significativo en un requerimiento. Tal vez, muchos requerimientos no visitaran algún estado. Cuando se definen los estados, se deben hacer las siguientes preguntas:

- ¿Qué pasa cuando las cosas fallan?
- ¿Qué ocurre durante la transición del estado?
- ¿Existe alguna transición ilegal de estados?

Se recomienda que al identificar los estados se tome en cuenta lo siguiente:

- Tratar de obtener el menor numero posible de estados.
- Asegurar que el diagrama de estados soportará todos los posibles escenarios
- El flujo de trabajo deberá soportar la realización de mediciones.
- El flujo de trabajo deberá soportar la integración de las reglas del negocio.

Este debe:

- Guardar los cambios en las transiciones ilegales de estados.
- Identificar acciones ilegales del grupo.
- Identificar notificaciones de requerimientos.
- Identificar el tiempo de los requerimientos para las mediciones y las escalaciones.

### Identificar Grupos.

Es importante definir el control de acceso, para esto, debemos considerar lo siguiente:

- ¿Qué roles existen en el proceso?
- De acuerdo a su rol, ¿todos los usuarios tienen necesidades de acceso al sistema similares?. Si no, se deben dividir en los múltiples roles.
- ¿Qué tipos de notificaciones en grupo se requieren?. Se puede configurar una notificación en grupo.
- ¿Qué datos de permisos de acceso son necesitados para cada rol definido?

### Organización de Datos.

Acercas de los datos de la organización, es necesario realizarse las siguientes preguntas:

- ¿Qué información es necesaria para sostener los procesos del negocio?
- ¿Qué información se requiere para realizar las mediciones?
- ¿Pueden todos los datos ser guardados en un solo esquema?, ¿Se requieren muchos esquemas para organizar la información?
- ¿Se necesitan esquemas adicionales para soportar la información?
- ¿Muchos esquemas necesitan información similar?
- ¿Es conveniente unir la información para los reportes?

Cuando se crea un esquema, debe incluir campos como:

- ¿Quién solicita el requerimiento?
- ¿Cuál es el requerimiento?
- ¿Cuándo fue hecho el requerimiento?
- ¿Quién está trabajando en el requerimiento?
- ¿Cuál es su estado actual, cuándo fue cambiado el estado, y quién hizo el cambio?

La aplicación puede necesitar información adicional o diferente para soportar el proceso y los métodos de reportes. Por ejemplo, tal vez se requiera identificar el tiempo de los niveles de servicio acordados y el tiempo del servicio actual para un requerimiento.

## Capítulo 5. Caso Práctico.

En el presente trabajo se desarrolla una aplicación cuyo objetivo es cubrir las necesidades que tiene un Centro de Ayuda a Usuarios de una tienda de autoservicio, para sistematizar su información, tomando en cuenta todas las actividades que tiene que cubrir, de acuerdo a los requerimientos del cliente.

### 5.1. Planteamiento.

Una empresa internacional dedicada al diseño/desarrollo, producción, instalación y servicio post-venta de la tecnología informática, implementará un centro de ayuda a usuarios para una tienda de autoservicio, que adquirió sus equipos.

Dicha empresa garantiza sus equipos por 2 años en promedio, al término del mismo, ofrece sus servicios de soporte, previo contrato, que puede incluir todos los equipos de la empresa (aunque sean de otro proveedor) y para ello requiere la implementación de un sistema para un Centro de Ayuda a Usuarios. En el contrato se especifican los tiempos de respuesta que se tienen para la solución de los reportes.

Se cuenta con toda la tecnología necesaria, por lo cual, esto no es un limitante para el desarrollo del sistema. Esta empresa tiene un acuerdo con la empresa Remedy, la cual le ofrece sus productos a un precio menor del mercado, así como las licencias, por lo que el sistema se desarrollo con Action Request System.

Esta empresa esta certificada en ISO 9000, como consecuencia, todos sus procesos están perfectamente definidos y documentados.

## 5.2. Requerimientos.

La función principal del Centro de Ayuda a la tienda de autoservicio, será dar soporte vía telefónica a los usuarios finales con el objeto de brindarles una solución a sus problemas de operación diaria, registrándolos, de manera que se de seguimiento continuo a sus reportes hasta el cierre de los mismos.

El Help Desk dará seguimiento a los problemas reportados con la finalidad de que sean atendidos por las áreas de solución correspondientes; y soporte en sitio para problemas mayores que requieran atención en el lugar en donde se presenta la falla.

Sus principales actividades son las siguientes:

- Recepción de llamadas
- Identificación del requerimiento
- Levantamiento de reporte en el sistema
- Diagnóstico del problema
- Solución inmediata del problema y/o canalización a las áreas de solución correspondientes.
- Validación de solución y servicio con los usuarios
- Seguimiento de reportes
- Cierre de reportes

El Objetivo del Centro de Ayuda a Usuarios es proporcionar servicio de soporte telefónico que contribuya a dar una rápida solución, dentro de los tiempos comprometidos, a los problemas que se presentan en los equipos de los usuarios finales.

A continuación se define el soporte telefónico y el soporte en sitio, es importante por que el manejo que se da a los reportes es diferente:

**El Soporte Telefónico**, es el servicio que se da en primera instancia cuando se presenta una falla. Cuando el usuario tenga un problema en sus equipos de cómputo o de impresión, puede marcar al Centro de Ayuda a Usuarios donde se le pedirán sus datos para levantar un reporte y se le darán una serie de instrucciones vía telefónica para resolver su problema rápidamente.

**El Soporte en Sitio**, es el servicio que se da cuando el problema que se presenta en la operación de los equipos no pueda ser resuelto vía telefónica y requiera ser atendido personalmente, se cuenta con un equipo de soporte en sitio, quienes asistirán a revisar el equipo y a darle solución al problema. Algunos ejemplo de requerimientos atendidos por soporte en Sitio son : Instalación de software, Movimiento e Instalación de Equipos, Fallas en el Hardware y todos los problemas que no hayan podido ser resueltos con la asistencia telefónica. En este caso, el sistema debe ser capaz de darle seguimiento al requerimiento, hasta su solución.

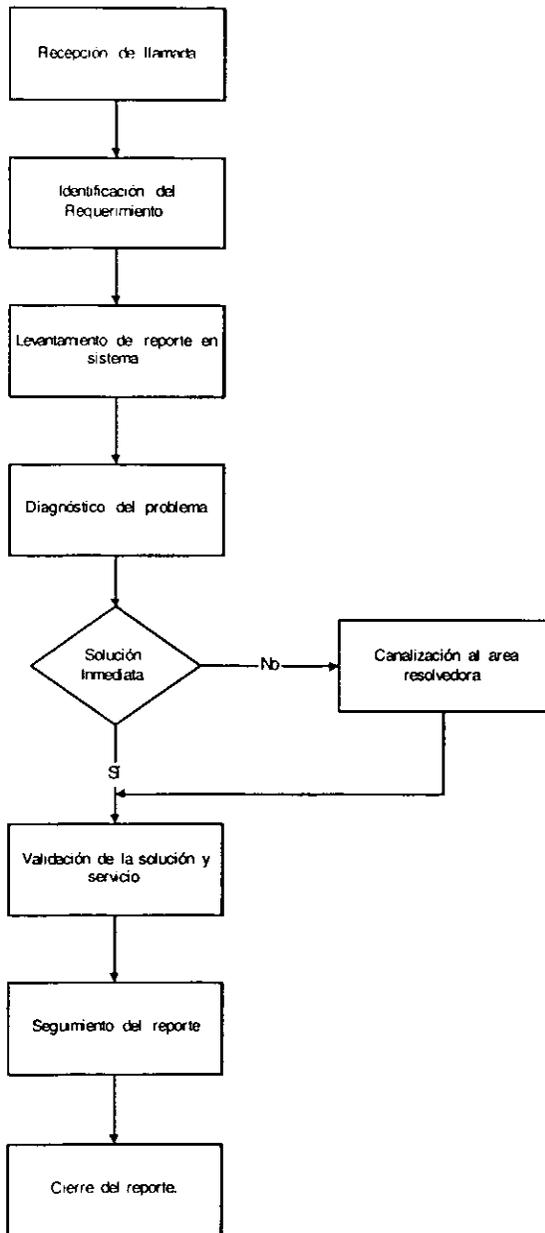
### **5.3. Análisis y diseño del Sistema.**

Siguiendo los pasos que recomienda la herramienta Action Request System, definidos en el capítulo 5, se obtiene lo siguiente:

#### **5.3.1. Definir el proceso.**

El flujo de trabajo, de acuerdo a las actividades que realiza el Centro de Ayuda, se puede representar en un diagrama de flujo, como se muestra a continuación:

Figura 5. Flujo de Trabajo del Centro de Ayuda a Usuarios de una Tienda de Autoservicio.



### 5.3.2. Identificar los puntos de control o estados del proceso.

Los estados del proceso son los siguientes:

1. Nuevo. Cuando se recibe el requerimiento, automáticamente está en un estado de nuevo, en tanto se identifique el requerimiento y se diagnostique el problema.
2. Asignado. Se asignará al personal de soporte en sitio indicado. Este estatus se utiliza, cuando el problema no puede ser resuelto vía telefónica y cambiará a atendido, hasta que el personal de soporte en sitio se reporte vía telefónica, indicando que se encuentra con el cliente.
3. Atendido. Cuando se este dando soporte vía telefónica o si el personal de soporte se encuentra en el sitio físicamente, el reporte tiene que estar en estado de atendido.
4. Suspendido. Se utilizará este estado, en caso de que el cliente solicite se atienda más tarde el reporte o si el equipo requiere algún cambio de parte.
5. Solucionado. Al terminar la atención del reporte y solucionar la falla se pone en estado solucionado.
6. Cerrado. El reporte solo podrá tomar el estado de cerrado, si se cuenta con la aprobación del usuario, es decir, que este totalmente satisfecho con el servicio.

El control de estos estados es importante, porque refleja que avance se tiene en los reportes, si están dentro del tiempo comprometido y se puede dar un buen seguimiento, manteniendo actualizado al cliente, ya que todas las acciones que se realizan, se documentan en la bitácora del sistema.

### 5.3.3. Identificar los roles involucrados.

Se determina que se utilizarán 2 grupos:

1. El grupo de soporte telefónico, llamado "Help Desk".
2. El grupo de coordinadores del Centro de Ayuda.

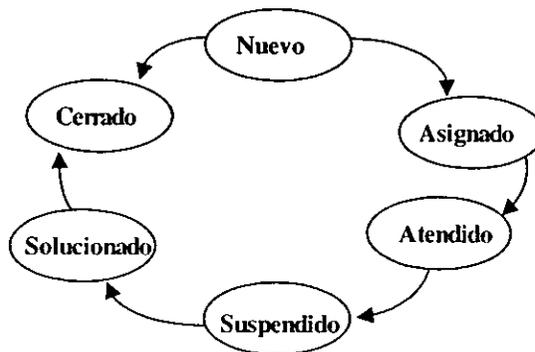
### 5.3.4. Identificar las mediciones o reportes.

1. Reportes cerrados dentro del tiempo comprometido.
2. Cuáles tiendas son las que tienen mayor número de reportes.
3. Cuáles son las fallas más reportadas
4. Reportes fuera de nivel de servicio, es decir, que se cierran después de los tiempos comprometidos.

### 5.3.5. Definir los elementos de los datos (esquemas y campos) e interfaces externas.

A continuación se describen todos los objetos que se utilizarán en la aplicación.

**Diagrama de Estados**



**Grupos**

Help Desk  
 Coordinadores

**Macros**

Nombre	Esquema	Cuando se ejecuta
HD: Agregar_solución	HD: Help Desk	Al oprimir el botón "Agregar Solución".
<b>Propósito</b>	Despliega el esquema de soluciones para que se agregue la solución dada a ese ticket y de esta forma, alimentar la base de conocimientos.	

Nombre	Esquema	Cuando se ejecuta
HD: Detalle_cliente	HD: Help Desk	Al oprimir el botón "Detalles"
<b>Propósito</b>	Despliega el esquema de clientes para mostrar la información detallada del cliente seleccionado.	

Nombre	Esquema	Cuando se ejecuta
HD: Crear_reporte	HD: Panel Principal	Al oprimir el botón "Crear Reporte".
<b>Propósito</b>	Generar un número de ticket con la información del cliente (nombre y clave)	

Active Links

Nombre	Esquema	Cuando se ejecuta
HD:CrearReporte	HD: Panel Principal	Al oprimir el botón "Crear Reporte".
<b>Propósito</b>	Verifica que los campos de 'clv_cliente', 'nom_cliente' y 'reportado_por' tengan información, si es así, ejecuta el macro de 'HD:Crear_reporte', de lo contrario despliega el siguiente mensaje: "Debe proporcionar el nombre o la clave del cliente y el nombre de la persona que esta reportando" y no ejecuta el macro.	

Nombre	Esquema	Cuando se ejecuta
HD: Agregar Solución	HD: Help Desk	Al oprimir el botón "Agregar Solución".
<b>Propósito</b>	Verifica que los campos de 'Componente', 'categoria', 'item', 'Descripcion_del_problema' y 'Problema' tengan información, si es así, ejecuta el macro de 'HD:Agregar_solución', desplegando el esquema de 'HD:Soluciones' para confirmar la introducción en la base de conocimientos de la nueva solución..	

Nombre	Esquema	Cuando se ejecuta
HD: ALcomponentes	HD: Help Desk	Al seleccionar menú "categoria".
<b>Propósito</b>	Hace visible el menú de "componentes", para presentar el listado de los componentes que pertenecen a la categoría seleccionada.	
Nombre	Esquema	Cuando se ejecuta
HD: ALItem	HD: Help Desk	Al seleccionar menú "componente".
<b>Propósito</b>	Hace visible el menú de "items", para presentar el listado de los items que pertenecen al componente seleccionado.	

Nombre	Esquema	Cuando se ejecuta
HD: ALMiembroHD	HD: Help Desk	Al seleccionar el menú "Asignar a".
<b>Propósito</b>	En caso de que seleccione 'Ing. en Sitio', oculta el campo de 'asig_a', que solo debe de ser visible en caso de que seleccione la opción 'Centro de Ayuda'.	

Nombre	Esquema	Cuando se ejecuta
HD: Detalle_cliente	HD: Help Desk	Al oprimir el botón "Detalles".
<b>Propósito</b>	Evalúa que la clave del cliente ó el nombre del cliente tengan un valor válido para ejecutar el macro de "HD: Detalle_cliente"	

Nombre	Esquema	Cuando se ejecuta
HD: Desactivar	HD: Help Desk	Al abrir el esquema "HD:Help Desk"
<b>Propósito</b>	Inhabilita los campos de 'Número de Reporte' y 'Fecha de reporte' para que no sean modificados.	

Nombre	Esquema	Cuando se ejecuta
HD: Fecha_solucion	HD: Help Desk	Al presionar la tecla 'enter' en el campo de fecha de solución.
<b>Propósito</b>	Si el estatus es 'cerrado', se despliega la fecha y la hora de solución.	

Nombre	Esquema	Cuando se ejecuta
HD: Buscar_clv_cliente	HD: Panel Principal	Al presionar el botón 'Buscar'.
<b>Propósito</b>	De acuerdo a la clave del cliente proporcionado, se realiza la búsqueda y se presenta un listado de los ticket's que cumplan con esa condición.	

Nombre	Esquema	Cuando se ejecuta
HD: Buscar_fecha	HD: Panel Principal	Al presionar el botón 'Buscar'.
<b>Propósito</b>	De acuerdo a la fecha proporcionada, se realiza la búsqueda y se presenta un listado de los ticket's que fueron creados a partir de la fecha dada.	

Nombre	Esquema	Cuando se ejecuta
HD: Buscar_Ticket	HD: Panel Principal	Al presionar el botón 'Buscar'.
<b>Propósito</b>	De acuerdo al número de ticket proporcionado, se realiza la búsqueda y se despliega el ticket que corresponda al número dado.	

### Menús

Nombre	Tipo	Propósito
HD: Asignaciones	Carácter	Lista las áreas de solución.
HD: Categorías	Carácter	Lista todas las categorías existentes
HD: Componentes	Carácter	Lista todas los componentes existentes
HD: Equipos	Carácter	Lista todos los equipos existentes en el inventario.
HD: Item	Carácter	Lista todos los item's existentes en el inventario.
HD: Marcas	Carácter	Lista todas las marcas existentes en el inventario.
HD: Menu_clv_cliente	Query <sup>a</sup>	Despliega la clave del cliente que corresponde al nombre del cliente proporcionado en el esquema 'HD: Panel Principal'.
HD: Menu_componente	Query	Lista los componentes que pertenezcan a la categoría seleccionada en el esquema de 'HD: Help Desk', de acuerdo a la información guardada en el esquema 'HD: Tipos.

<sup>a</sup> Búsqueda.

HD: Menu_item	Query	Lista los item's que pertenezcan al componente seleccionado en el esquema de 'HD: Help Desk', de acuerdo a la información guardada en el esquema 'HD: Tipos.'
HD: Menu_marca	Query	Lista las marcas que correspondan al equipo seleccionado en el esquema de 'HD: Help Desk', de acuerdo a la información guardada en el esquema 'HD: Inventario.'
HD: Menu_modelo	Query	Lista los modelos que correspondan al equipo y a la marca seleccionados en el esquema de 'HD: Help Desk', de acuerdo a la información guardada en el esquema 'HD: Inventario.'
HD: Menu_no_serie	Query	Lista los modelos que correspondan al equipo, a la marca y al modelo seleccionados en el esquema 'HD: Help Desk', de acuerdo a la información guardada en el esquema de 'HD: Inventario.'
HD:Menu_nombre_cliente	Query	Despliega el nombre del cliente que corresponda a la clave del cliente, seleccionado en el esquema 'HD: Panel Principal', de acuerdo a la información guardada en el esquema de 'HD: Clientes.'
HD: Modelo	Carácter	Lista todos los modelos existentes en el inventario.
HD: Soluciones	Query	Despliega la solución que corresponda de acuerdo a la categoría, componente e item seleccionados en el esquema 'HD: Help Desk', de acuerdo a la información guardada en el esquema de 'HD: Soluciones'

### Filtros

Nombre	Esquema	Cuando se ejecuta
HD:Bitácora_completa	HD: Help Desk	Al modificar un ticket.
<b>Propósito</b>	Verifica que siempre que se modifique un ticket, se documente en la bitácora que cambios se realizaron.	

Nombre	Esquema	Cuando se ejecuta
HD:Casos_Cerrados	HD: Help Desk	Al modificar un ticket.
<b>Propósito</b>	Evita que se modifique un ticket cerrado.	

Nombre	Esquema	Cuando se ejecuta
HD:Inventario_duplicado	HD: Inventario	Al modificar o dar de alta un equipo.
<b>Propósito</b>	Evita que se duplique la información de un equipo en el inventario.	

Nombre	Esquema	Cuando se ejecuta
HD:Registro_completo	HD: Help Desk	Al crear un ticket.
<b>Propósito</b>	Verifica que se los campos de 'categoría', 'Componente', 'item', 'Equipo', 'Marca', 'Modelo' y 'serie' contengan un valor.	

Nombre	Esquema	Cuando se ejecuta
HD:Solucion_completa	HD: Help Desk	Al cerrar un ticket.
<b>Propósito</b>	Verifica que el status del ticket este en cerrado y que los campos de 'vobo', 'fecha_solucion', 'resuelto_por', 'solucion' y 'tipo_reporte' contengan un valor.	

Nombre	Esquema	Cuando se ejecuta
HD:Tipos_duplicados	HD: Tipos	Al crear un tipo.
<b>Propósito</b>	Evita que se duplique la combinación de un tipo al darlo de alta.	

**Escalaciones**

Nombre	Esquema	Intervalos
HD: Severidad1	HD: Help Desk	30 min.
<b>Propósito</b>	Notificar al analista que recibió el ticket, cada 30 min., el estado del reporte hasta que este cerrado, para que él tome las acciones necesarias.	
<b>Severidad</b>	1	
<b>Tiempo de solución</b>	3 horas	

Nombre	Esquema	Intervalos
HD: Severidad2	HD: Help Desk	1 hora.
<b>Propósito</b>	Notificar al analista que recibió el ticket, cada hora, el estado del reporte hasta que este cerrado, para que él tome las acciones necesarias.	
<b>Severidad</b>	2	
<b>Tiempo de solución</b>	4 horas	

Nombre	Esquema	Cuando se ejecuta
HD:Severidad3	HD: Help Desk	1:30 horas.
<b>Propósito</b>	Notificar al analista que recibió el ticket, cada 1:30 horas el estado del reporte hasta que este cerrado, para que él tome las acciones necesarias.	
<b>Severidad</b>	3	
<b>Tiempo de solución</b>	6 horas	

**Esquemas:**

Nombre del Esquema: HD: Clientes.

Nombre del Campo	Descripción	ID	Requerido?	Tipo de dato	Longitud	Acceso a cambios	Menú?
clv_cliente	Clave de Cliente	536870924	Requerido	Carácter	50	Coordinadores	No
nom_cliente	Nombre de Cliente	536870920	Requerido	Carácter	50	Coordinadores	No
Contacto	Contacto	536870923	Opcional	Carácter	50	Coordinadores	No
Direccion	Dirección	8	Requerido	Carácter	128	Coordinadores	No
Estado	Estado	536870921	Opcional	Selección		Coordinadores	Sí
Telefono	Teléfonos	536870922	Opcional	Carácter	50	Coordinadores	No
fecha_registro	Fecha de registro	3	Sistema	Fecha/Hora			No
ultima_modificación_por	Última Modificación por	5	Sistema	Carácter	30		No
fecha_modificacion	Fecha de modificación	6	Sistema	Fecha/Hora			No
registro	Registró	2	Requerido	Carácter	30		No

Nombre del Esquema: HD: Panel Principal.

Nombre del Campo	Descripción	ID	Requerido?	Tipo de dato	Longitud	Acceso a cambios	Menú?
nom_cliente	Nombre de Cliente	536870913	Opcional	Carácter	60	Help Desk	No
clv_cliente	Clave de Cliente	4	Opcional	Carácter	30	Help Desk	No
reportado_por	Reportado por:	536870916	Opcional	Carácter	50	Help Desk	No
Recibio	Recibió	2	Requerido	Carácter	30		No
No.Ticket	No. de Ticket	536870919	Opcional	Carácter	15	Coordinadores Help Desk	No
clv_clienteb	Clave de Cliente	536870920	Opcional	Carácter	15	Coordinadores Help Desk	No
Mes	Mes	536870921	Opcional	Carácter	2	Coordinadores Help Desk	No
Día	Día	536870923	Opcional	Carácter	2	Coordinadores Help Desk	No
Año	Año	536870925	Opcional	Carácter	2	Coordinadores Help Desk	No

Nombre del Esquema: HD: Help Desk.

Nombre del Campo	Descripción	ID	Requerido?	Tipo de dato	Longitud	Acceso a cambios	Menú?
No_reporte	Número de Reporte	1	Sistema	Carácter	15		No
reportado_por	Reportado por	536870961	Opcional	Carácter	50	Help Desk	No
clv_cliente	Clave del Cliente	536870921	Opcional	Carácter	15	Help Desk	No
nom_cliente	Nombre del Cliente	536870922	Opcional	Carácter	60	Help Desk	No
Fecha_de_reporte	Fecha de reporte	3	Sistema	Fecha/Hora			No
Nom_usuario	Nombre del Usuario	536870920	Opcional	Carácter	50	Help Desk	No
Telefono	Teléfono	536870923	Opcional	Carácter	15	Help Desk	No
departamento	Departamento	536870924	Opcional	Carácter	50	Help Desk	No
Status	Status	7	Requerido	Selección		Help Desk	No
severidad	Severidad	536870934	Opcional	Selección		Help Desk	No
categoria	Categoría	536870933	Opcional	Carácter	69	Help Desk	Sí

Componente	Componente	536870931	Opcional	Carácter	69	Help Desk	Sí
item	Item	536870932	Opcional	Carácter	69	Help Desk	Sí
Problema	Problema	8	Requerido	Carácter	128	Help Desk	No
Descripcion_del_problema	Descripción del Problema	536870935	Opcional	Carácter	255	Help Desk	No
bitacora	Bitácora	536870952	Opcional	Diario		Help Desk	No
Asignado_a	Asignado a:	4	Opcional	Carácter	30	Help Desk	Sí
	asig_a	536870964	Opcional	Carácter	20	Help Desk	No
Ultima_modificación_por	Última Modificación por	5	Sistema	Carácter	30		No
fecha_modificación	Fecha de modificación	6	Sistema	Fecha/Hora			No
Submitter	Recibió	2	Requerido	Carácter	30		No
posible_solucion	Solución propuesta	536870936	Opcional	Carácter	50	Help Desk	No
Equipo	Equipo	536870943	Opcional	Carácter	30	Help Desk	Sí
Marca	Marca	536870944	Opcional	Carácter	30	Help Desk	Sí
Modelo	Modelo	536870945	Opcional	Carácter	30	Help Desk	Sí
serie	Serie	536870960	Opcional	Carácter	30	Help Desk	Sí
resuelto_por	Resuelto por	536870953	Opcional	Carácter	50	Help Desk	Sí
fecha_solucion	Fecha de Solución	536870954	Opcional	Fecha/Hora		Help Desk	No

tipo_reporte	Tipo de Reporte	536870955	Opcional	Selección		Help Desk	No
vobo	Vo.Bo.	536870956	Opcional	Carácter	50	Help Desk	No
solucion	Solución	536870957	Opcional	Carácter	100	Help Desk	No
desc_solucion	Descripción de la Solución	536870959	Opcional	Carácter	255	Help Desk	No

Nombre del Esquema: HD: Inventario.

Nombre del Campo	Descripción	ID	Requerido?	Tipo de dato	Longitud	Acceso a cambios	Menú?
clv_equipo	Clave de Equipo	1	System	Carácter	15		No
equipo	Equipo	536870919	Requerido	Carácter	30	Help Desk Coordinadores	Sí
marca	Marca	536870920	Requerido	Carácter	30	Help Desk Coordinadores	Sí
modelo	Modelo	536870922	Requerido	Carácter	30	Help Desk Coordinadores	Sí
Serie	Serie	536870924	Requerido	Carácter	30	Help Desk Coordinadores	Sí
Fecha_registro	Fecha de registro	3	Sistema	Fecha/Hora			No
Submitter	Registró	2	Requerido	Carácter	30		No

Nombre del Esquema: HD: Soluciones.

Nombre del Campo	Descripción	ID	Requerido?	Tipo de dato	Longitud	Acceso a cambios	Menú?
problema	Problema	536870922	Requerido	Carácter	50	Help Desk	No
problema_detallado	Problema Detallado	536870919	Opcional	Carácter	255	Help Desk	No
solucion	Solución	536870917	Requerido	Carácter	100	Help Desk	No
Solución Detallada	Solución Detallada	536870918	Opcional	Carácter	255	Help Desk	No
categoria	Categoría	536870921	Requerido	Carácter	50	Help Desk	Sí
Componente	Componente	536870915	Requerido	Carácter	25	Help Desk	Sí
Item	Item	536870920	Requerido	Carácter	30	Help Desk	Sí
Status	Status	7	Requerido	Selección		Help Desk	No

Nombre del Esquema: HD: Tipos.

Nombre del Campo	Descripción	ID	Requerido?	Tipo de dato	Longitud	Acceso a cambios	Menú?
clv_tipo	Clave del Tipo	1	Sistema	Carácter	15	Help Desk	No
Status	Status	7	Requerido	Selección		Help Desk	No
Categoría	Categoría	536870913	Requerido	Carácter	50	Help Desk	Yes
Componente	Componente	8	Requerido	Carácter	25	Help Desk	Yes
Item	Item	4	Requerido	Carácter	30	Help Desk	Yes
fecha_creacion	Fecha de Creación	3	Sistema	Fecha/Hora			No
ultima_modificación_por	Última Modificación por	5	Sistema	Carácter	30		No
fecha_modificacion	Fecha de modificación	6	Sistema	Fecha/Hora			No
registro	Registró	2	Requerido	Carácter	30		No

## 5.4. Desarrollo

Después de definir todas las propiedades de todos los objetos, se puede empezar con el desarrollo de la aplicación, para lo cual utilizaremos la herramienta "Administration Tool". Solo los usuarios con permiso de administrador o customizador pueden ingresar a esta herramienta.

Como primer paso, es necesario firmarse en el sistema, como se muestra en la figura 5.1

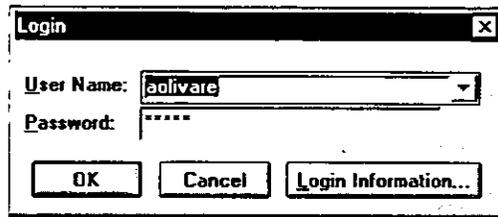


Figura 5.1

Se introduce el nombre del usuario y la clave del mismo, el sistema evalúa esta información. Si es válida, ya se puede empezar a trabajar.

Posteriormente se despliega la ventana del servidor, la cual muestra los objetos guardados en el servidor al que ingresamos. Como se puede observar en la figura 5.2, en la parte izquierda, hay un listado de los objetos que tiene el sistema, si damos un click en cualquier objeto, como los esquemas, los filtros, los menús o alguno de ellos, del lado derecho se desplegarán todos los existentes.

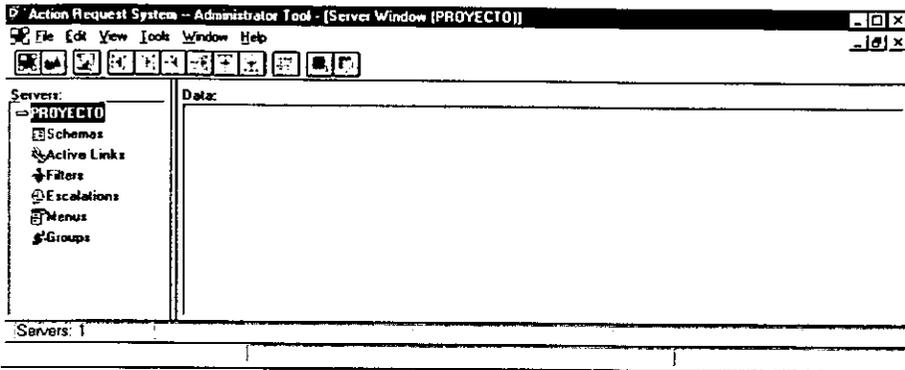


Figura 5.2

Por ejemplo, si damos click en "Groups", se desplegará un listado de los grupos que están definidos, como vemos en la figura 5.3

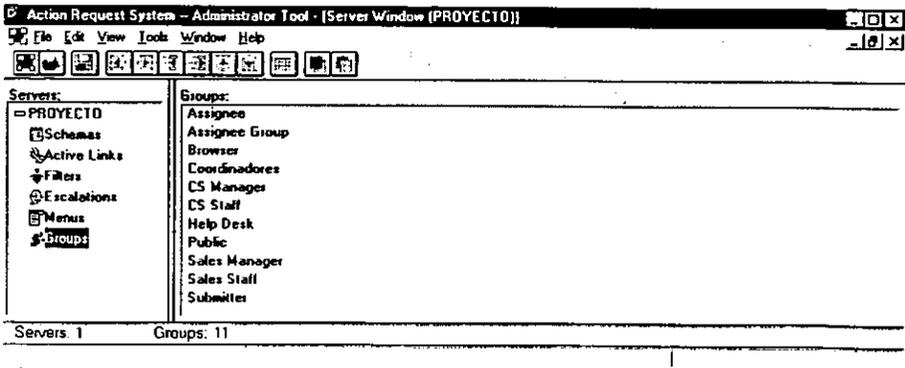


Figura 5.3

Para crear un nuevo objeto, damos click en el icono "New Server Object" , se desplegará una pequeña ventana en donde se puede seleccionar el objeto que deseamos. Ver figura 5.4.

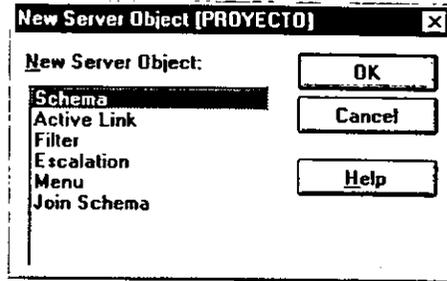


Figura 5.4

Si elegimos esquema, se mostrará una pantalla de diseño para el esquema. Como se muestra en la figura 5.5

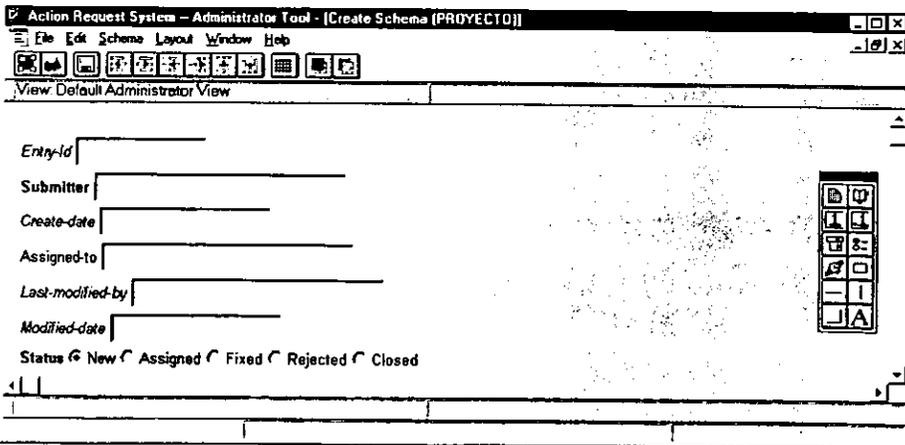


Figura 5.5

Podemos observar que por omisión aparecen los 7 campos obligatorios, estos se pueden ocultar o cambiar la etiqueta para guardar la información que se quiera, también se puede ver una barra de herramientas del lado derecho de la pantalla, que a continuación definiremos:

Icono	Descripción.
	Crea un campo de texto.
	Se utiliza para crear un diario o bitácora.
	Para un nuevo campo de números enteros.
	Crea un campo de números reales
	Crea un campo de selección en forma de lista.
	Crea un campo de selección.
	Crea un campo de fecha / hora
	Crea un nuevo botón.
	Crea una línea horizontal o vertical.
	Crea una caja
	Crea una etiqueta.

En caso de elegir crear un nuevo "Active Link", se presentará una ventana que contiene todas las propiedades que debe contener, como se despliega en la figura 5.6.

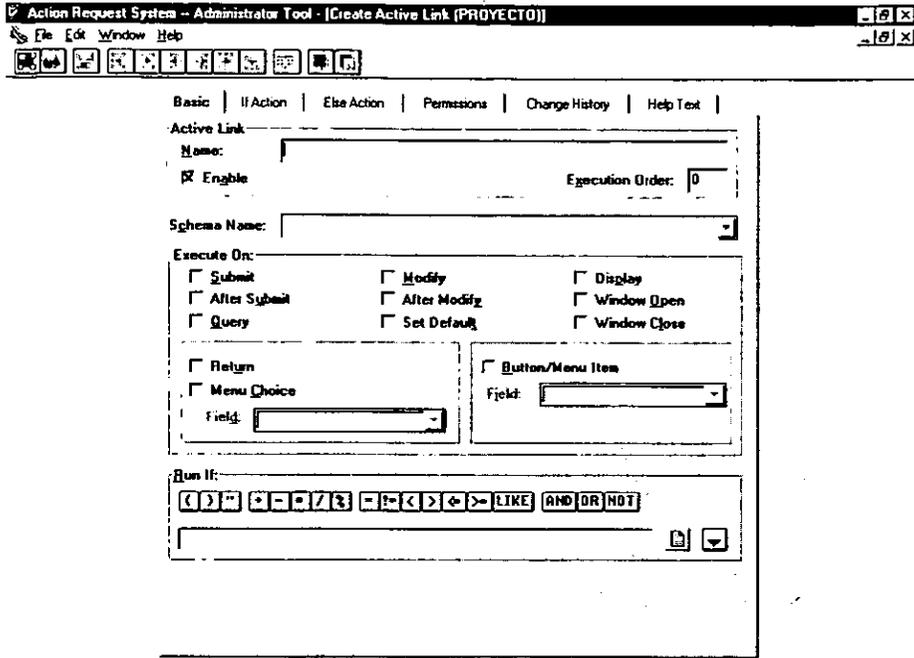


Figura 5.6

Así también si se elige un nuevo filtro, una nueva escalación o un nuevo menú. Cada uno con sus propias características, pero las ventanas que se despliegan son similares.

### 5.4.1. Descripción del Sistema.

En el AR User, es en donde los usuarios tendrán acceso al sistema, a continuación se describe como utilizarlo. Es importante mencionar, que no esta dentro del alcance de este trabajo, la implementación de este sistema.

Para iniciar una sesión en el AR User, es necesario que el usuario se firme al sistema, proporcionando su usuario y su password (los cuales fueron asignados por el administrador del sistema), como se muestra en la figura 5.7.

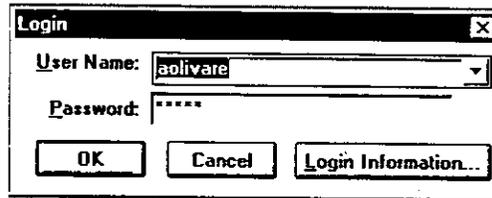


Figura 5.7.

Es importante describir la barra de herramientas de la aplicación, como se ve en la figura 5.8.

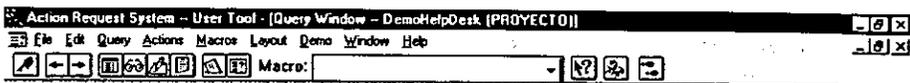


Figura 5.8.

Los iconos que la conforman son los siguientes:

**Icono Descripción.**

-  Se utiliza para guardar los cambios realizados en el registro.
-  En caso de que no se hallan hecho cambios, el icono se presenta inhabilitado.
-  Se utilizan para desplazarse de un registro a otro, en caso de seleccionar varios a la vez.
-  Se utiliza para realizar un listado que cumplan con las características que se introducen en una ventana de búsqueda.
-  Presentar un registro sólo para lectura, sin permitir realizar cambio alguno, seleccionado desde un listado o en una ventana de búsqueda.
-  Se utiliza para modificar un ticket o cualquier registro que se seleccione en un listado o en una ventana de búsqueda.
-  Con este icono, se puede realizar un reporte a partir de varios registros seleccionados o de acuerdo a la ventana de búsqueda seleccionada.
-  Al dar click en este icono, se presenta un listado de los esquemas a los que el usuario tiene acceso.
-  Este icono, despliega las ventanas de búsqueda que el usuario puede utilizar para realizar sus listados o buscar un reporte con características más específicas.
- Macro** En esta parte se muestran todos los macros realizados en la aplicación.

Al dar click en este icono y después en un campo de algún esquema, se despliega una ventana que muestra todas las propiedades de dicho campo de texto o de selección.
- 

Posteriormente, ya que el usuario se firmó al sistema, ahora podrá ingresar a los esquemas que tiene acceso. Para ello, puede desplegar el listado de estos esquemas dando click en el icono ya mencionado (  ), se abrirá la ventana mostrada en la figura 5.9.

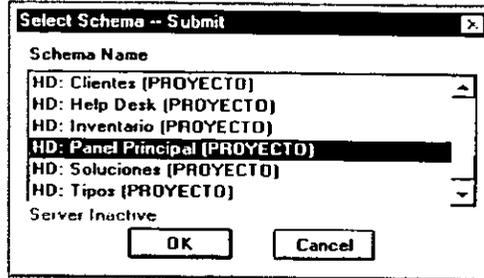


Figura 5.9.

En este caso, el esquema inicial para el Centro de Ayuda es el “HD: Panel Principal”, se selecciona y se da click en el botón “OK”. En este esquema tienen acceso los usuarios del grupo “Help Desk” para dar de alta un ticket o realizar búsquedas de ticket's ya existentes para su actualización. Lo anterior se visualiza en la figura 5.10.

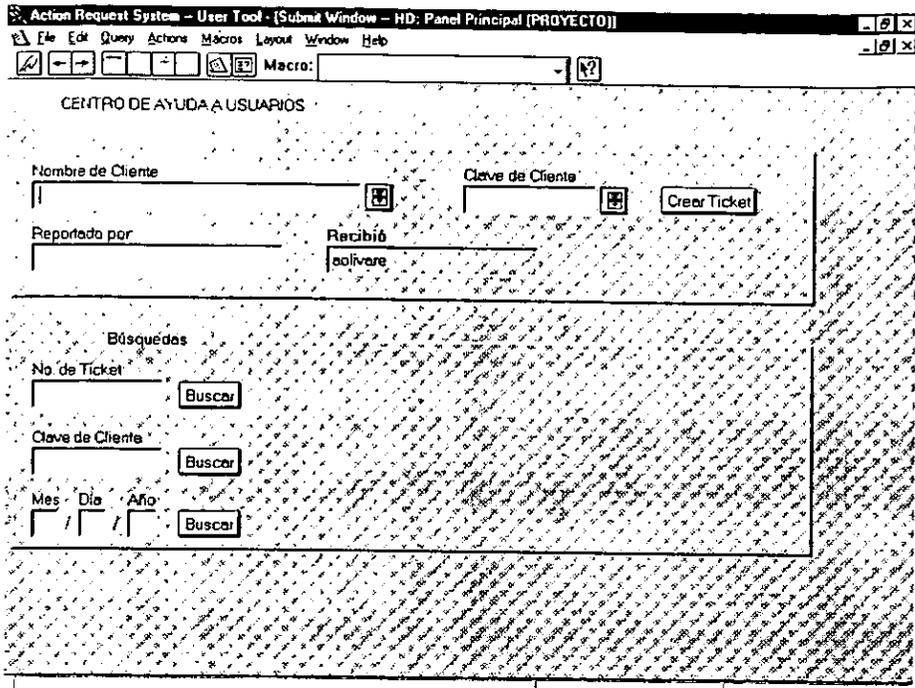


Figura 5.10.

Para realizar las búsquedas se puede hacer de tres formas:

1. Por número de ticket.
2. Por la clave del cliente.
3. A partir de una fecha, hasta la fecha actual.

Por otra parte, para que el analista abra un nuevo ticket, debe ingresar el nombre y la clave del cliente, así como el nombre de la persona que reporta, y posteriormente presionar el botón de "Crear reporte", entonces se desplegará el esquema de "HD: Help Desk", con los datos proporcionados y en donde el analista podrá documentar todos datos del caso que proporcione el cliente. También puede hacer uso de la base de conocimientos, con las soluciones sugeridas por el sistema de acuerdo a la categoría, al componente y al item

dado. El esquema de "HD:Help Desk", mostrado en las figuras 5.11 y 5.12 se divide en cinco partes:

1. Información del cliente. Contiene los datos del cliente, como son el nombre, la clave, el teléfono, la persona que reporto, así como un botón en donde ingrese a la información más detallada del cliente. Ver figura 5.13.
2. Información del caso. En esta área se documentarán los detalles del caso, cual es el problema y su descripción, así como la categoría, el componente y el ítem al que corresponde, también es en esta sección en donde se asigna un área de solución, que pueden ser el analista o un ingeniero de sitio, si se requiere.
3. Información del equipo. Aquí se registran los datos del equipo que tiene la falla y sólo los equipos que estén dados de alta en el inventario se desplegarán en los menús, porque están contrato, de ahí la importancia de esta parte.
4. Tips. De acuerdo a los datos introducidos en la información del caso, el analista podrá consultar posibles soluciones a la falla.
5. Solución. Al cerrar el ticket, es indispensable llenar los datos de la solución, porque si falta alguno de ellos, el ticket no se cerrará. Si el analista considera que la solución dada a este ticket se puede agregar a la base de conocimiento, dispone de un botón llamado "Agregar solución".

En este esquema se dará seguimiento a la vida de un ticket, documentando cualquier cambio realizado en la bitácora o historial.

Action Request System - User Tool - (Modify Individual - HD: Help Desk (PROYECTO))

File Edit Query Actions Macros Layout Window Help

Macro: [ ]

CENTRO DE AYUDA A USUARIOS

Información del Cliente

Número de Reporte	Reportado por	Clave del Cliente	Nombre del Cliente
000000000000007			
Fecha de reporte	Nombre del Usuario	Teléfono	Departamento
Sunday, July 29, 20			

Detalles

Información del Caso

Status

Nuevo  Asignado  Atendido  Suspendido  Solucionado  Cerrado

Severidad [ 1 ]

Categoría [ ] Componente [ ] Item [ ]

Problema [ ] Bitácora [ ]

Descripción del Problema [ ]

Asignado a: [ ]

Number 1 of 1

Figura 5.11

Action Request System - User Tool - [Modify Individual - HD: Help Desk (PROYECTO)]

File Edit Query Actions Macros Layout Window Help

Macro: [ ]

Última modificación por: [activare] Fecha de Modificación: [Sunday, August 05, 2007] Recibió: [activare]

Tips

Solución propuesta

Possible solución de acuerdo a la información proporcionada en los campos de Categoría, Componente e Item.

Información del Equipo

Equipo: [ ] Marca: [ ] Modelo: [ ] Serie: [ ]

Solución

Resuelto por: [ ] Fecha de Solución: [ ] Tipo de Reporte: [Servicio]

Vó Bo: [ ] Solución: [ ]

Descripción de la Solución: [ ]

Agregar Solución

Number 1 of 1

Figura 5.12.

La información de los clientes, se maneja con el esquema "HD: Clientes", en este esquema se pueden dar de alta o modificar los datos de los clientes. La clave de cada cliente se conforma con las dos primeras letras del cliente y un número consecutivo, misma que se introduce en el sistema como se muestra a continuación. Ver figura 5.13.

Action Request System - User Tool - [Submit Window - HD: Clientes (PROYECTO)]

File Edit Query Actions Macros Layout Window Help

Macro: [ ]

CENTRO DE AYUDA A USUARIOS

Cliente

Clave de Cliente: [ ] Nombre de Cliente: [ ] Contacto: [ ]

Dirección: [ ] Estado: [ ] Teléfono: [ ]

Fecha de registro: [ ] Última Modificación por: [ ] Fecha de modificación: [ ] Registro: [activare]

Figura 5.13

Para dar de alta el inventario, tenemos el esquema de "HD: Inventario" y se muestra en la figura 5.14. El coordinador y los miembros del grupo de "Help Desk" tienen acceso a este esquema para dar de alta equipos, el sistema verifica automáticamente que no se dupliquen.

Figura 5.14.

La base de datos de conocimiento, se llena con el esquema de "HD: Soluciones" y se muestra en la figura 5.15.

Figura 5.15.

Las soluciones están ligadas a la categoría, componente e ítem que se seleccione. El analista puede decidir si agrega una solución o no a la base de datos, al cerrar un reporte.

También tenemos el esquema de "HD: Tipos", en donde se dan de alta las combinaciones de categorías, componentes e ítems que se pueden presentar en los tickets reportados al Centro de Ayuda. Este esquema se muestra en la figura 5.16.

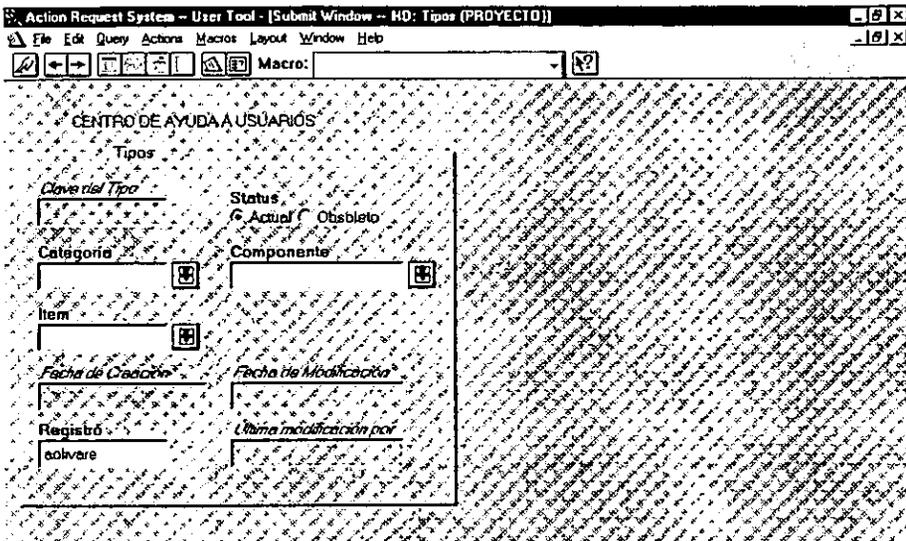


Figura 5.16.

Otra herramienta importante es el "Notification Tool", a través del cual se notificará a los miembros del Help Desk que un ticket aún no esta resuelto, para que tome las acciones necesarias para que el ticket se cierre dentro del tiempo comprometido. Esta herramienta se muestra en la figura 5.17.

The screenshot shows a window titled "Notification Tool - aplicare" with a menu bar containing "File", "Edit", and "Help". Below the menu bar is a table with three columns: "Date", "Source", and "Message". The table contains three rows of notification data. At the bottom of the window, there is a status bar that reads "Total number of notifications : 207".

Date	Source	Message
8/13/01 11:56	AR	Reporte 000000000000010 pendiente de solución. URGENTE I. Recibido el 8/5/01 12:47:22 PM
8/13/01 11:56	AR	Reporte 000000000000015 pendiente de solución. URGENTE I. Recibido el 8/5/01 9:52:19 PM
8/13/01 11:56	AR	Reporte 000000000000016 pendiente de solución. URGENTE I. Recibido el

Figura 5.17

También tenemos las ventanas de búsquedas, las cuales son unas pantallas con distintivos azules que muestran todos los campos pertenecientes al esquema seleccionado y tiene una barra de búsqueda que se muestra en la figura 5.18, en la cual se pueden formar sentencias en SQL para realizar búsquedas.

The screenshot shows a search query builder interface. It features a toolbar with various operators and logical connectors: parentheses, minus, plus, asterisk, slash, equals, less than, greater than, less than or equal to, greater than or equal to, LIKE, AND, OR, NOT, and a "Fields..." button. Below the toolbar is a text input field labeled "Query:".

Figura 5.18

Para realizar búsquedas de inventario (figura 5.19), tipos (figura 5.20), soluciones (figura 5.21) y clientes (figura 5.22) se utilizarán las ventanas de búsqueda que se presentan a continuación:



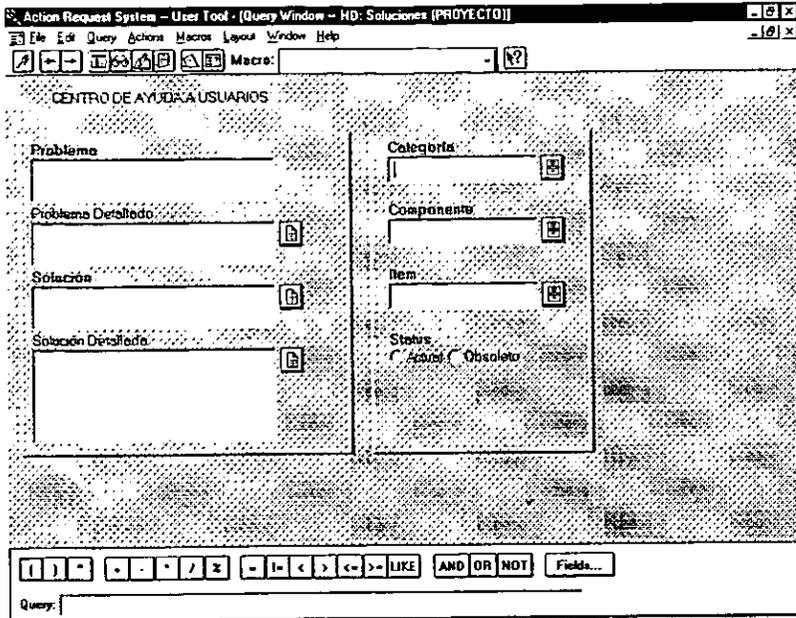


Figura 5.21

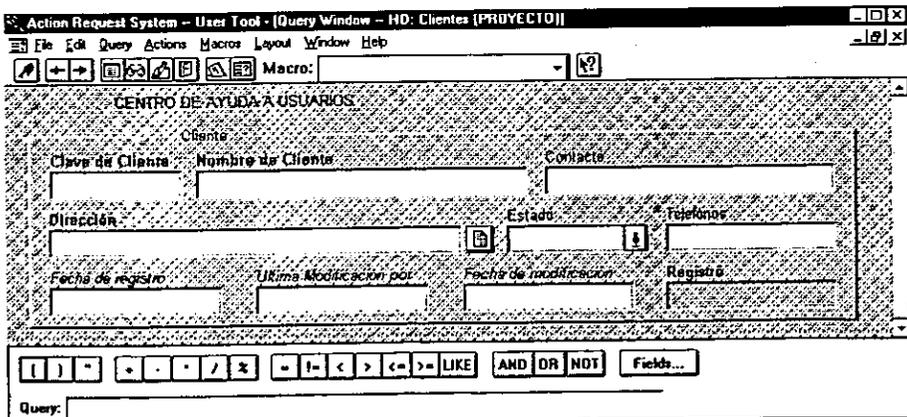


Figura 5.22

Y para hacer búsquedas más específicas de los tickets, también se puede utilizar su ventana de búsqueda, la cual se muestra en las figuras 5.23, 5.24 y 5.25.





Para abrir las ventanas de búsqueda mencionadas anteriormente, se tendrá que dar click en el icono de "open query"  se abrirá una ventana con el listado de las que están disponibles, como se muestra en la figura 5.26., se elige la que se necesita, presionando el botón de "OK." y se abrirá la ventana de búsqueda seleccionada.

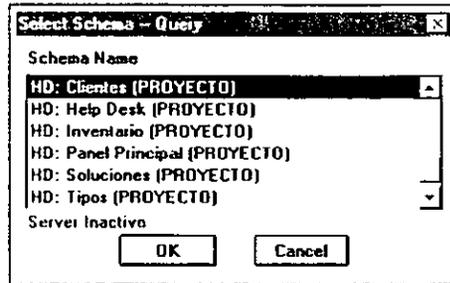


Figura 5.26

Por último, tenemos los reportes, el AR User cuenta con una herramienta para realizarlos, como se muestra en la figura 5.27, que despliega todos los campos que componen el esquema seleccionado, se eligen los que se desean incluir en el reporte, ordenándolos, realizando estadísticas, si se desea, y generando el reporte. Se puede salvar en un disco flexible o bien en disco duro, con formato \*.arx (para realizar respaldos o importaciones de datos) o \*.csv (de texto, separando los datos por comas) y además también se puede desplegar en pantalla, utilizando el AR System User Tool, en Excel o en algún procesador de texto. También se puede enviar a una impresora local o remota.

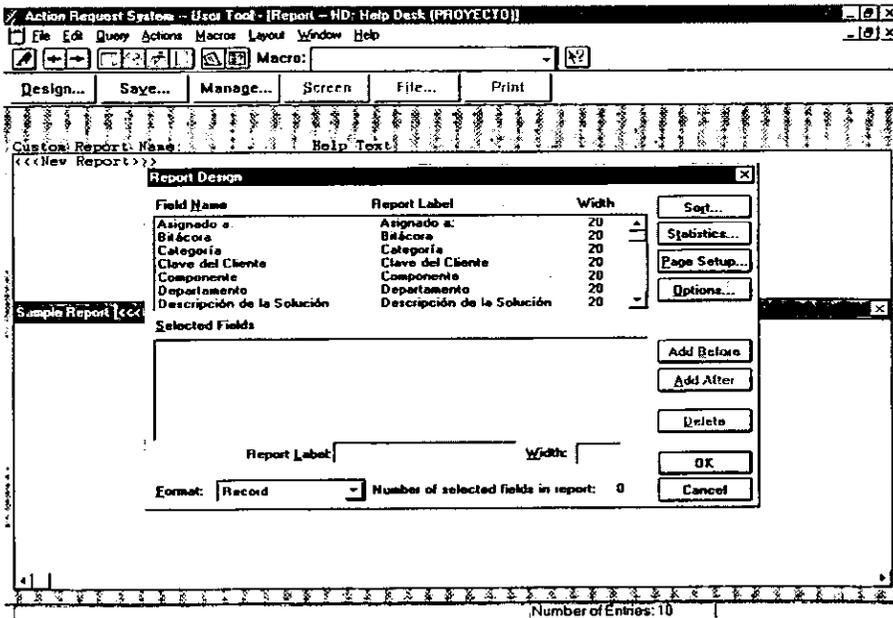


Figura 5.27

## CONCLUSIONES.

Los sistemas de información, actualmente están ocupando un sitio muy importante en las empresas. de ahí que sea fundamental el saber elegir el adecuado. Específicamente en este trabajo, se desarrollo un sistema para un centro de ayuda a usuarios de una tienda de autoservicio, se explicaron las actividades que realizan y lo benéfico que es contar con un sistema de información. También se puso en manifiesto lo importante que es el tiempo para los centros de ayuda, mismo que se reduce considerablemente si se cuenta con el sistema adecuado.

El Action Request System es una herramienta muy robusta que permite el desarrollo rápido y eficaz de un sistema para un centro de ayuda, adaptándose perfectamente a sus necesidades. Además, permite una mejora continua y la importación masiva de datos. Posteriormente, se pueden anexar más herramientas de Remedy a este sistema, como es su integración en web o el envío automático de mensajes al radio localizador.

El sistema desarrollado cumple con las expectativas del cliente, maneja la información de los *tickets*, los clientes, los equipos que están en contrato, así como una base de datos de conocimientos, manteniendo la información actualizada, permite realizar diversas búsquedas y generar reportes, y si se requiere de algún cambio, no es necesario invertir demasiado tiempo en ello. Además, al utilizar una aplicación con arquitectura cliente/servidor, se obtiene la centralización de la información, evitando que se duplique y permitiéndolo una actualización más eficiente.

Por último, es importante recalcar que la implementación de este sistema no esta dentro del alcance de este trabajo, sin embargo, en el capítulo 4 de "Action Request System", se describen las características que debe de tener el equipo para soportar la instalación del mismo.

## BIBLIOGRAFÍA

1. **“Cliente/servidor. Guía de supervivencia.” 2ª ed.**  
Orfali, Robert; Harkey, Dan y Jeri, Edwards.  
McGrawHill, 1996, E.U.  
676 pp.
  
2. **“Análisis y diseño de sistemas”. 3ª ed.**  
Kendall & Kendall  
Prentice Hall Hispanoamericana, S. A., 1997, E. U.  
913 pp.
  
3. **“Análisis y diseño de sistemas de información.” 2ª ed.**  
A. Senn, James.  
McGrawHill, 1992, México  
942 pp.
  
4. **“Sistemas de información para los negocios.  
Un enfoque de toma de decisiones”. 3ª ed.**  
Cohen, Daniel y Asín, Enrique.  
McGrawHill, 2000, México.  
413 pp.
  
5. **“Action Request System 3.0,  
Administrator’s Guide, Volumen 1”**  
Remedy Corporation, 1997, U.S.A.  
520 pp

6. **“Action Request System 3.0,  
Administrator’s Guide, Volumen 2”**  
Remedy Corporation, 1996, U.S.A.  
350 pp.
  
7. **“A System for Users,  
Student Guide”**  
Remedy Corporation, 1998, U.S.A.  
226 pp.
  
8. **“Administering the AR System from UNIX/Motif”**  
Remedy Corporation, 1998, U.S.A.  
505 pp.
  
9. **“Action Request System 3.0  
Error Messages Guide”**  
Remedy Corporation, 1998, U.S.A.  
188 pp.

[www.helpdesk.com](http://www.helpdesk.com)

[www.remedy.com](http://www.remedy.com)

[www.helpdesks.com](http://www.helpdesks.com)