UNIVERSIDAD AUSTRAL DE CHILE
Facultad de Ciencias de la Ingeniería
Escuela de Ingeniería en Computación

Construcción Módulo Web Orientado a Atención de Clientes Empresa "Procom"

Seminario de Titulación
para optar
al título de
Ingeniero de Ejecución en Computación

Profesor Patrocinante:
Srta. Claudia Zil Bontes

Christian Eduardo Aguilar Marín

Puerto Montt Chile 2002

Con cariño y agradecimiento a

Padres y Hermanos.

INDICE

1. Introducción..... 1

2. Objetivos..... 6

2.1. Objetivo General..... 6

2.2. Objetivos Específicos..... 6

3. Planteamiento del Problema..... 8

3.1. Antecedentes..... 8

3.1.1. Definición del problema..... 8

3.1.2. Identificación de esfuerzos anteriores para resolverlo..... 9

3.1.3. Solución propuesta..... 10

3.1.4. Equipo de trabajo..... 11

3.2. Justificación..... 11

3.2.1. Situación sin proyecto..... 11

3.2.2. Situación con proyecto..... 12

3.3. Delimitación..... 13

4. Metodología..... 14

4.1. Planificación de la base de datos..... 17

4.2. Definición del sistema..... 17

4.3. Análisis y colección de requerimientos..... 18

4.3.1. Análisis y colección de requerimientos para la base de datos..... 18

4.3.2. Anteproyecto de desarrollo del sitio web..... 18

4.4. Diseño..... 19
4.4.1. Diseño base de datos..... 19
4.4.2. Diseño sitio web..... 20
4.5. Selección DBMS..... 21
4.6. Prototipo..... 21
4.7. Implementación..... 21
4.7.1. Implementación base de datos..... 21
4.7.2. Implementación sitio web..... 21
4.8. Conversión y carga de datos..... 22
4.9. Prueba..... 22
4.10.Difusión sitio web..... 22
4.11.Mantenimiento operacional..... 23
4.11.1. Mantenimiento operacional de la base de datos..... 23
4.11.2. Mantenimiento o actualización sitio web..... 23
4.12.Evolución sitio web..... 23
5. Recursos..... 24
5.1. Hardware..... 24
5.1.1. Hardware de servidor..... 24
5.1.2. Hardware de empresa..... 27
5.1.3. Hardware de red..... 27
5.2. Software..... 30
5.2.1. Software del servidor..... 30

5.2.1.1. Aspectos generales software del servidor..... 30
6. Planificación de la base de datos..... 35
6.1. Evaluación de sistemas información actuales..... 35
6.2. Recursos..... 36
6.2.1. Recursos humanos..... 36
6.2.2. Recursos financieros..... 36
7. Definición del sistema..... 39
7.1. Alcance del sistema..... 39
7.2. Forma de comunicación con las áreas..... 40
8. Análisis y colección de requerimientos..... 41
8.1. Análisis y colección de requerimientos base de datos..... 41
8.2. Anteproyecto del desarrollo del sitio web..... 43
8.2.1. Entrevista con el gerente de la empresa..... 44
8.2.2. Requerimientos del usuario..... 44
9. Diseño de la base de datos..... 46
9.1. Diseño conceptual de la base de datos..... 46
9.1.1. Especificación de entidades..... 46
9.1.2. Identificación de relaciones..... 48
9.1.3. Identificación y detalle de atributos de entidades..... 49
9.1.4. Determinación de los dominios de atributos..... 61
9.1.5. Identificación de claves candidatas y elección de clave primaria para cada entidad..... 65

9.2. Diseño lógico de la base de datos..... 68

9.2.1. Transformación del modelo de datos local conceptual al local lógico..... 69

9.2.1.1. Eliminación de relaciones mucho a muchos (M:N)..... 69

9.2.1.2. Eliminación de relaciones recursivas..... 71

9.2.1.3. Derivación de relaciones del modelo de datos lógico..... 71

9.2.1.4. Normalización del modelo..... 74

9.2.1.5. Validación del modelo contra las transacciones..... 76

9.2.1.6. Diseño transacciones..... 79

9.2.1.7. Integridad y consistencia..... 84

9.2.2. Integración de la base de datos..... 88

9.2.2.1. Integración del modelo de datos lógico local al modelo global..... 88

9.2.2.1.1. Revisar nombres de entidades y sus claves primarias..... 89

9.2.2.1.2. Revisar nombres de relaciones..... 89

9.2.2.1.3. Verificar claves foráneas..... 90

9.2.2.1.4. Verificar las restricciones de integridad..... 90

9.2.2.2. Modelo E-R global..... 91

9.3. Diseño físico de la base de datos..... 93
9.3.1. Traducción del modelo de datos lógico local para el DBMS especificado..... 93
9.3.2. Diseño de relación base para un DBMS específico..... 94
9.3.2.1. Tipo de datos e integridad..... 94
9.3.2.2. Desencadenadores o triggers..... 95
9.3.2.3. Índices..... 96
9.3.3. Diseño de representación física..... 96
9.3.3.1. Estimación del espacio en disco para el almacenamiento de datos..... 96
9.3.4. Políticas de seguridad para los datos..... 97
10. Diseño del sitio web..... 98
10.1. Diseño lógico del sitio web..... 98
10.1.1. Organización de la información..... 98
10.1.1.1. Esquemas de organización..... 99
10.1.1.1.1. Esquemas de organización exactos..... 99
10.1.1.1.2. Esquemas de organización ambiguos.....100
10.1.1.1.3. Esquemas de organización híbridos..... 101
10.1.1.2. Estructura de organización..... 101
10.1.1.2.1. Estructura de organización jerárquica..... 102
10.1.1.2.2. Estructura de organización hipertexto.....103

10.1.1.2.3. Estructura de organización orientada a base de datos..... 104
10.1.2. Diseño de sistemas de navegación..... 105
10.1.2.1. Sistemas de navegación jerárquicos..... 105
10.1.2.2. Sistemas de navegación global..... 105
10.1.3. Diseño de sistemas de rotulación..... 106
10.1.3.1. Rótulos dentro de sistemas de navegación..... 106
10.1.3.2. Rótulos como términos indizados..... 108
10.1.3.3. Rótulos de vínculos..... 108
10.1.3.4. Rótulos como encabezados..... 108
10.1.4. Inventario de las páginas web..... 109
10.2. Diseño físico del sitio web..... 111
10.2.1. Generadores de aplicación web..... 111
10.2.1.1. Microsoft FrontPage 98..... 111
10.2.1.2. Editplus.....112
10.2.1.3. Macromedia Flash 5.0..... 112
10.2.2. Forma en que se cargan las páginas..... 113
10.2.3. Levantamiento del servidor web..... 114
10.2.4. Medida seguridad sitio web..... 115
11. Selección del DBMS..... 116
12. Prototipo..... 117
13. Seguridad..... 118

13.1. Acceso de datos..... 118

13.2. Seguridad de Microsoft Windows NT 4.0..... 120

13.3. Seguridad web…… 121

13.3.1. Firewalls…… 121

13.3.1.1. Beneficios de un Firewalls..... 121

13.3.1.2. Limitaciones de un Firewalls..... 122

14. Implementación..... 123

14.1. Implementación de la base de datos..... 123

14.1.1. Creación de tablas de la base de datos..... 123

14.1.2. Creación de procedimientos almacenados..... 128

14.1.3. Triggers o desencadenadores..... 141

14.2. Implementación del sitio web..... 147

14.2.1. Creación de código ASP y HTML para las páginas web..... 147

14.2.2. Vistas de distintas páginas web que forman parte del sistema..... 166

14.2.2.1. Pantalla portada principal..... 167

14.2.2.2. Formulario orden de trabajo para clientes..... 168

14.2.2.3. Pantalla calendario de trabajos..... 169

14.2.2.4. Pantalla administración calendario de trabajos..... 170

14.2.2.5. Pantalla operaciones con clientes..... 171

14.2.2.6. Pantalla administración factura..... 172

14.2.2.7. Pantalla lista de facturas pendientes..... 173

14.2.2.8. Pantalla modifica calendario de un determinado cliente.. 174

14.2.2.9. Pantalla ingreso claves..... 175

14.3. Implementación de la arquitectura final para el funcionamiento del sistema.....175

15. Conversión y carga de los datos..... 178

16. Pruebas..... 179

16.1. Prueba a procedimientos almacenados..... 179

16.2. Prueba a triggers..... 179

16.3. Prueba de seguridad..... 180

16.3.1. Verificación del acceso a través de contraseñas..... 181

16.3.2. Verificación de niveles de seguridad..... 181

16.3.3. Verificación seguridad base de datos..... 182

16.4. Prueba de aceptación visual..... 182

16.5. Prueba de rendimiento..... 182

17. Conclusión..... 187

18. Bibliografía..... 189

19. Anexos..... 190

A. Documentos recolectados en la etapa de análisis y colección de requerimientos..... 190

Tablas.

1. Equipo de trabajo en proyecto web..... 11
2. Características de computadores empresa Procom..... 27
3. Especificación de entidades..... 46
4. Identificación de relaciones..... 48
5. Identificación y detalle de atributos de entidades..... 50
6. Determinación de dominios de atributos..... 61
7. Identificación de claves candidatas y elección de clave primaria para cada entidad..... 66
8. Transacciones versus requerimientos..... 78
9. Diseño de transacciones..... 79
10. Estrategias para mantener la integridad de la base de datos..... 86
11. Situación al actualizar o eliminar en una entidad padre..... 87
12. Rótulos de la página principal del sitio..... 107

Figuras.

1. Diagrama etapas del ciclo de vida de una base de datos de Thomas Connolly..... 15
2. Diagrama metodología compuesta..... 16
3. Esquema de comunicación de la empresa con el exterior..... 26
4. Red computacional inicial de la empresa Procom..... 29
5. Flujo de datos desde un documento HTML hasta una base de datos y camino inverso..... 33
6. Información que viaja por Internet, a través de IIS..... 34
7. Áreas con las que se relaciona el sistema..... 40
8. Modelo E-R conceptual preliminar..... 67
9. Eliminación de relaciones mucho a muchos M:N..... 70
10. Mapa de transacciones..... 82
11. Construcción del E-R final local..... 83
12. Cambio nombre de relación..... 90
13. Modelo E-R lógico global..... 92
14. Modelo de organización jerárquico..... 102
15. Algunos pasos de una estructura jerárquica aplicada al sitio web..... 103
16. Estructura de organización hipertexto..... 104
17. Sistema de navegación global..... 106
18. Rótulos de vínculos..... 108
19. Rótulos de encabezados..... 109

20. Portada principal..... 167

21. Orden de trabajo para clientes..... 168

22. Calendario de trabajos..... 169

23. Administración calendario de trabajos..... 170

24. Operaciones con clientes..... 171

25. Administración factura..... 172

26. Listado de facturas pendientes..... 173

27. Modifica Calendario..... 174

28. Ingreso claves..... 175

29. Arquitectura final que permite el funcionamiento del sistema..... 177

30. Descarga página principal sitio web..... 183

31. Tiempo del servidor en crear y trasmitir código HTML..... 184

Síntesis.

El presente trabajo de titulación, está enfocado a la construcción de un módulo web orientado al cliente, para la empresa de servicio técnico PROCOM.

El objetivo que se persigue al construir este módulo es poder acercarse de mejor forma a los clientes de esta empresa y a la vez ocupar la diversidad de herramientas computacionales que existen. No obstante, se persigue también ordenar la información de los distintos clientes con que se cuenta, llevando los registros y trabajos efectuados a éstos. Por otro lado, el método de desarrollo que se utiliza proviene de una mezcla entre el diseño de una base de datos y ciclo de vida, cuyo autor es Thomas Connolly y una estrategia para el diseño de un sitio web de los autores Roxana Bassi y Eduardo E. Mercovich. Con esto se consigue obtener un método final para realizar la construcción del módulo en forma ordenada, abarcando así, una serie de objetivos y reglas que se tienen que tomar en cuenta al momento de construir un sistema.

Como conclusión de este sistema, se puede decir que el trabajo cumple con todos los objetivos que la empresa propuso. Se consigue con ésto, un sitio web, el que proporciona un medio de comunicación alternativo, cuyo objeto es acercar aún más a los clientes con la empresa y a la vez entregar un mejor servicio como respuesta.

SYNTHESIS.

The actual degree work, is focus to build a client-oriented web module, for technical services PROCOM enterprise.

The goal to be reached building this module is can be closed to the customers of this enterprise and at the same time use the diversity of computing tools existing. Nevertheless, it pursues also order the information about the distinct customers who they count, taking registers and works made to them. On the other side, the development method used come from a mixture between database design and life cycle, this by Thomas Connolly and a web site design strategy by Roxana Bassi & Eduardo E. Mercovich. With this it gets obtain a final method to do a building of the module in an ordered way, reaching thus, a series of goals and rules that it has to take in consideration at the moment of building a system.

As a conclusion of this system, it can say that this work completes with all the goals that enterprise proposed. It gets with this, a web site, that provides a way of alternative communication, whose objective is get closer to the customers with the enterprise and at the same time give a better service as an answer.

1. Introducción.

En la actualidad hay una gran competencia en el rubro computacional, por lo que las empresas de servicio técnico están en continua competencia para captar un número mayor de clientes y también retener a los actuales. Es por eso que la calidad de los servicios que se le entregan al cliente toma un carácter más preponderante, a la vez que la productividad para la empresa se transforma en un mayor crecimiento y eficiencia en ésta.

Debido a lo anterior, la empresa de servicio técnico Procom decidió la construcción de una Extranet, con el objeto de cumplir con las actuales exigencias del mercado y a la vez permitirse el interactuar con los clientes, consiguiendo así entregar un mejor servicio, como también mantener informado al cliente y conforme.

Dentro del entorno de trabajo existen tres entidades notorias:

1. **Empresa de Servicio:** Es aquella empresa cuya actividad gira en torno con el cliente, a quien proporciona servicios.
2. **Cliente:** Aquí se distinguen dos tipos de clientes. Por un lado se encuentra el **cliente externo**, el cuál puede ser una empresa que requiera de servicios. Por otro lado, está el **cliente interno**, los cuales están formados por el personal de la empresa, quienes se ven a veces en la necesidad de requerir servicios.

En la realización de este proyecto interviene una metodología compuesta por otras tres, las cuales conjugadas permiten un correcto abordamiento a los problemas para los cuales este sistema se diseñó, con el objeto de entregar una alternativa de comunicación con los clientes, como también un orden interno de la información. Esta metodología está constituida por el Ciclo de vida y el diseño de una base de datos cuyo autor es Thomas Connolly [capítulo 18, bibliografía Connolly1999], siendo complementado a la vez, por una estrategia para el diseño de un sitio web de Roxana Bassi y Eduardo E. Mercovich [capítulo 18, bibliografía Mercovich2000], consiguiendo así con esto último abordar de mejor forma lo referente a como construir un sitio web, tomando en cuenta los distintos factores, como la organización de la información, estructura de la información, sistemas de navegación entre otros.

A continuación, se describe el contenido de cada capítulo de este libro:

Capitulo 1. Este capítulo comprende la parte de introducción de esta tesis.

Capítulo 2. Se mencionan aquí, los distintos objetivos que se persiguen a través del sistema. Se da a conocer el objetivo general y los objetivos específicos.

Capítulo 3. En el capítulo 3 se da a conocer el planteamiento del problema, la definición del problema en los cuales se describen todos los elementos que determinan en definitiva la realización del sistema y por supuesto la solución de éstos, a la vez se menciona la justificación del desarrollo del sistema.

Capítulo 4. En este capítulo se encuentra la metodología, que se aplica en el desarrollo del sistema. Se nombran cada una de las etapas de ésta con sus correspondientes definiciones.

Capítulo 5. Dentro de este capítulo se nombran los recursos que intervienen en el proyecto como son los recursos de hardware y software.

Capítulo 6. Se mencionan aquí, lo referente a la planificación de la base de datos, la evaluación de sistemas actuales que existen en la empresa a la que se le desarrolla el sistema.

Capítulo 7. En este capítulo se define el sistema como también el alcance que este va a tener para la empresa y las áreas de ésta.

Capítulo 8. Dentro de este capítulo se describe todo lo referente al análisis y colección de requerimientos con el propósito de empezar a formar una idea de la base de datos y el diseño web a realizar.

Capítulo 9. Se comienza el diseño de la base de datos, lo que implica diseño conceptual, diseño lógico y diseño físico y todo lo que implica cada una de estas etapas, incluyendo por último la integración de la base de datos.

Capítulo 10. El capítulo en cuestión, contiene todo lo referente al diseño del sitio web. Esto involucra la organización de la información, el diseño de los

sistemas de navegación como también el diseño de los sistemas de rotulación. A la vez, se mencionan las herramientas que se utilizaron para la construcción del sistema.

Capítulo 11. Este capítulo involucra la selección del DBMS, donde se elige el motor de base de datos de acuerdo con las características y requerimientos del sistema.

Capítulo 12. Dentro de este capítulo se menciona el prototipo como una de las etapas de la metodología que se usa.

Capítulo 13. El contenido de este capítulo se enmarca dentro de la seguridad global del sistema. Esto es seguridad web, seguridad de acceso a datos, las políticas de seguridad que se efectúan, con el propósito de la protección de los datos, entre otros aspectos.

Capítulo 14. Aquí se hace referencia a la implementación del sistema, comprendiendo lo que es la implementación de la base de datos y sitio web. También se dan a conocer las distintas pantallas del sistema y parte del código de programación.

Capítulo 15. Aquí se menciona como se cargan los datos de la empresa al nuevo sistema para que este funcione.

Capítulo 16. En este capítulo se dan a conocer las distintas pruebas que se realizaron al sistema con el objeto de detectar errores.

Capítulo 17. Se realizan las conclusiones de este proyecto, lo que significa haberlo realizado y la importancia del uso de nuevas herramientas para la empresa a la que se desarrolló el sistema.

Capítulo 18. Se da a conocer la bibliografía que se utilizó en el desarrollo de esta tesis.

2. Objetivos.

2.1 Objetivo General.

Este trabajo de titulación tiene como objetivo general, diseñar y construir un sistema orientado a web para la empresa Procom, de tal forma que permita establecer una vía de comunicación alternativa y efectiva entre el cliente y la empresa.

2.2 Objetivos Específicos.

Los objetivos específicos del siguiente trabajo son:

- Obtener una interfaz amigable para que el Administrador pueda trabajar con las tablas de la base de datos y también los clientes y personal de la empresa puedan hacer consultas a la base de datos, agilizándose así, las actividades dentro y fuera de la empresa.
- Permitir llevar a cabo consultas por parte de la empresa, del estado de cuenta de los clientes, ver pagos pendientes de estos y obtener fechas de trabajos.
- Permitir generar órdenes de trabajo por parte de los clientes, como también cotizaciones.

- Generar informes para ver la calendarización de trabajos de ciertos clientes, como también informes de cuentas de éstos.
- Mejorar la gestión administrativa de la empresa, organizando la información que maneja la empresa.
- Mejorar los tiempos de respuesta al entregar información, tanto interna como externa.
- Mantener a la empresa en un nivel competitivo en el comercio.

3. Planteamiento del problema.

3.1 Antecedentes.

3.1.1 Definición del problema.

Procom está ubicada en la calle Manuel Rodríguez 245 oficina 7 en la ciudad de Puerto Montt. Es una empresa cuya labor es prestar servicios técnicos en el área computacional, la que está conformada por tres técnicos, un vendedor, una secretaria y el gerente general, a la vez es representante de la empresa Telectronic, la que reside en Santiago y que se dedica a la asesoría tecnológica, siendo un importante proveedor e integrador de plataformas de multiservicio de comunicaciones. Por consiguiente, la empresa posee una amplia variedad de clientes que necesitan formas alternativas a las tradicionales para poder comunicarse.

Sin embargo, esta empresa enfrenta día a día un problema, que es el no llevar registros adecuados de peticiones en cuanto a trabajos, cotizaciones, pues estas peticiones realizadas en su gran mayoría por teléfono, quedan registradas en cuadernos o libretas de notas, ocasionando a la larga un mal tratamiento de la información como por ejemplo, demora en el tiempo de respuesta, pérdida de los datos anotados o incluso la pérdida al captar trabajos ya sean, por estar los teléfonos ocupados o por el hecho de no tener otro medio

de difusión que acerque aún más al cliente con la empresa y la labor que ésta desempeña. Por otro lado, la gran mayoría de la información de clientes, de trabajos y ventas, quedan registradas por medio de documentos como por ejemplo facturas. No obstante, una pequeña parte de la información se almacena en planillas Excel, lo que refleja la no existencia de registros adecuados para el tratamiento de la información.

De acuerdo a lo anterior, resulta de suma importancia la realización de este sistema, pues el gran auge de la computación y por ende de las empresas ligadas a este rubro provoca un aumento de la competencia en el área, lo que se traduce en un avance en el manejo de datos y productos, o sea una mejora en el servicio a clientes.

3.1.2 Identificación de esfuerzos anteriores para resolverlo.

Usualmente en la recolección de datos, ya sean por efecto de trabajos o ventas, van siendo registradas en forma de documentos y también en algunas planillas Excel, lo que produce lentitud en las respuestas a peticiones o trabajos determinados. Por otro lado, cuando se reciben llamados telefónicos por efecto de peticiones de trabajos o cotizaciones, éstas van siendo registradas en cuadernos, lo que no es óptimo.

Cabe mencionar además, que no existe ningún método o sistema que apoye a los técnicos en terreno, esto es llevar un historial de los computadores de clientes, para tal efecto, se está realizando un módulo que dá solución a este

problema por el Sr. Julio A. Reinoso Villarroel, con el cuál se completa el sistema orientado a web, que comprenderán dos módulos, uno orientado al cliente al que se refiere este seminario y otro orientado al servicio técnico el que forma parte del seminario titulación del señor antes mencionado.

3.1.3 Solución propuesta.

Debido al gran auge de las empresas de servicio, resulta estimulante la realización de un sistema que resuelva tanto los problemas de la empresa como de los clientes, con el propósito de mantenerse a un nivel competitivo en el mercado.

La solución comprende la realización de un sistema vía web, lo que contribuye en gran parte a mejorar la comunicación de los clientes con la empresa. Esto se construirá bajo una plataforma de Microsoft Windows NT Server 4.0, con una base de datos Microsoft SQL Server 7.0 y un servidor web (Internet Information Server), usando además como herramienta de programación EditPlus y Frontpage 98. No obstante, también se realizará una Extranet, que es una Intranet más una extensión del acceso a usuarios autorizados. Es decir, no sólo el personal de la organización tiene acceso a la información, sino también usuarios autorizados por la empresa, como por ejemplo: clientes, proveedores, empresas asociadas entre otros. El objeto de esto último es tener una interfaz para el administrador de tal forma que sea amigable en el momento de trabajar con ciertas tablas de las bases de datos.

3.1.4 Equipo de trabajo.

Este proyecto web consta de dos módulos que a continuación se describen con sus respectivos desarrolladores.

El equipo de trabajo es el siguiente:

Tabla Nº 1 Equipo de trabajo en proyecto web.

Desarrolladores	Módulos
Christian Aguilar Marín	Sist. Web orientado al cliente.
Julio Reinoso Villarroel	Sist. Web orientado al servicio técnico.

Dentro del proyecto web, existe un trabajo en conjunto con el Sr. Julio Reinoso Villarroel, en lo que respecta principalmente al diseño web. Esto se refiere a:

- Definición y estandarización de pantallas y nombres de archivos.
- Elección de plataforma de trabajo y software de desarrollo.

3.2 Justificación.

3.2.1 Situación sin proyecto.

Sin proyecto, la situación de la empresa quedaría bajo el sistema de comunicación habitual con los clientes, siendo ésta realizada por medio del

teléfono y correo electrónico. Esto provocaría un estancamiento en el desarrollo de la empresa, desencadenando así, un menor nivel en las prestaciones de servicios a clientes y por ende un bajo nivel de competencia con sus pares.

3.2.2 Situación con proyecto.

Actualmente, la empresa Procom ha decidido mejorar la comunicación con sus clientes y organizar de mejor forma la información interna que se maneja. Al construir una extranet, la que albergará el sistema a diseñar e implementar, provocará un aumento del servicio al cliente y a la vez permitirá un fácil acceso del servicio técnico a la información clasificada. Todo esto agilizará las transacciones internas de la empresa y también la productividad, ganando así tiempo valioso no sólo para el servicio técnico, sino también mejorar el tiempo de respuesta a los clientes. Con esto, se persigue que la gran gama de clientes de esta empresa se sentirán aún más interesados de solicitar servicios, consiguiéndose así, ofrecerles al empresariado y particulares, una forma distinta a la tradicional de comunicarse con la empresa y a la vez, está podrá ponerse a la par con las grandes empresas de este rubro, logrando un gran paso en el desarrollo y aprovechamiento de los recursos tecnológicos que existen hoy en día.

Por último, no cabe duda que al implementar este sistema, que en su culminación estará formado por dos módulos bien definidos, servirá de gran

ayuda en las relaciones comunicativas tanto internas como externas de la empresa.

3.3 Delimitación.

Las delimitaciones de este sistema están dadas en las etapas finales de la metodología compuesta. Por tanto las delimitaciones son las siguientes:

- Puesto que no existe un sistema previo definido como tal, en la etapa de conversión se opta por traspasar o poblar la base de datos de manera tradicional, esto quiere decir, escribiendo la información a la base de datos uno por uno.
- Otra delimitación ocurre en la etapa de prueba, pues éstas se realizan en forma limitada debido a lo complejo y lento de realizar pruebas a un sitio web. No obstante, esto no quita que se realicen pruebas clásicas para ver el comportamiento del sistema.
- Por otro lado, lo que se refiere a las etapas de mantenimiento operacional y evolución del sitio web serán efectuadas por el personal de la empresa, como también el manejo de la base de datos, a lo que se refiere llenado y actualización de ésta.

4. Metodología.

La metodología usada para el desarrollo del sistema es una mezcla de las metodologías de Thomas Connolly [figura Nº 1]: “Ciclo de vida” y “Diseño de una base de datos” con la siguiente, que es “Una Estrategia para el diseño web” de los autores Roxana Bassi y Eduardo E. Mercovich. Además, la parte del diseño web fue complementada con el libro “Arquitectura de la información para www” de los autores Louis Rosenfeld y Peter Morville.

La idea de mezclar estas metodologías [figura 2] en una sola, se debe al hecho de poder asegurar el adecuado desarrollo del sistema, permitiendo con esto abarcar de mejor forma lo que se refiere principalmente a la construcción del sitio web, complementando así las etapas normales de un sistema de información. Todo esto se debe a que no existe una metodología bien definida para la construcción de un sitio web que abarque tanto las etapas normales del desarrollo de un sistema como el diseño del sitio web.

Cabe destacar que esta metodología no se lleva a cabo en su totalidad, puesto que hay etapas como mantenimiento operacional y evolución del sitio que estarán desarrolladas por personal de la empresa.

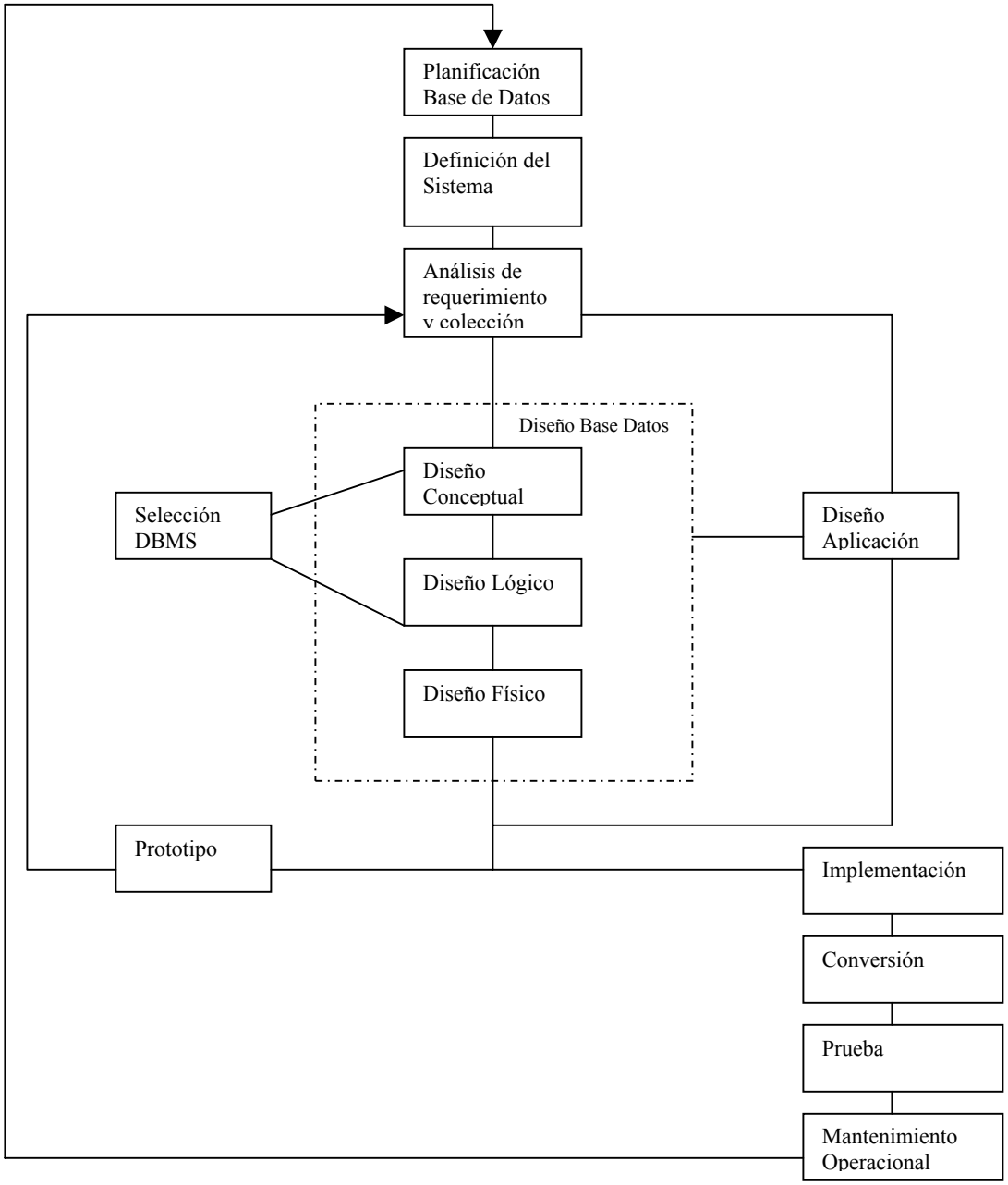


Figura Nº 1. Diagrama etapas del ciclo de vida de una base de datos de Thomas Connolly.

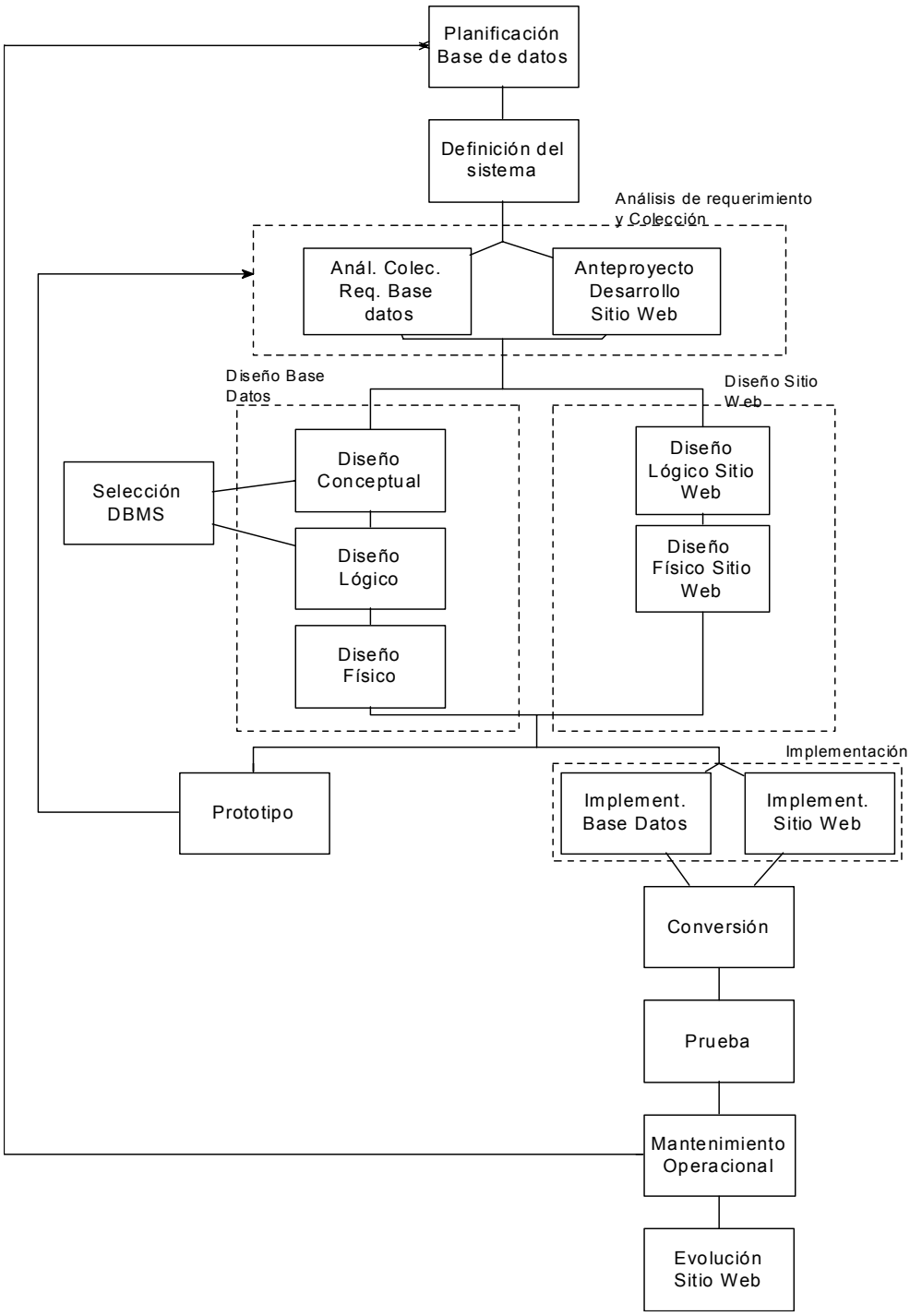


Figura 2. Diagrama metodología compuesta.

4.1 Planificación de la base de datos.

El objetivo que se persigue con esta etapa, a parte de ser el punto de partida de la ejecución del proyecto en cuestión, es tener claro el trabajo a realizar, los recursos necesitados, como también se debe tener en cuenta lo siguiente:

- Identificación de los planes de negocio y metas a obtener.
- Evaluar sistemas de información actuales.

Así mismo, con esto comienza la recolección preliminar de datos y la documentación.

4.2 Definición del sistema.

Aquí se definen los límites del sistema y la manera que se comunica con las otras fuentes de información de la empresa. También es importante definir el sistema en términos de los usuarios, no obstante con esto se obtiene de mejor forma el alcance del sistema.

4.3 Análisis y colección de requerimientos.

4.3.1 Análisis y colección de requerimientos para la base de datos.

En esta fase se comienza a coleccionar y analizar la información de la organización. Esta información que se recolecta es usada para identificar las necesidades de los usuarios que a la vez formarán parte de las futuras aplicaciones de la base de datos en este nuevo sistema.

La colección de los datos se realiza a través de procedimientos como por ejemplo:

- Entrevistas.
- Observación de operaciones.
- Examinación de documentos que despliegan y actualizan.
- Utilización de cuestionarios para buscar información.
- Utilización de experiencias en otros sistemas en general.

4.3.2 Anteproyecto de desarrollo del sitio web.

Aquí se determinan los requerimientos y objetivos del cliente (audiencia, lenguaje, gráfica, alcances) y basándose en estos se hace una presentación de las ideas básicas de la propuesta.

4.4 Diseño.

4.4.1 Diseño base de datos.

Esta fase comprende el diseño conceptual, diseño lógico y diseño físico.

- **Diseño conceptual:**

En esta etapa se construye un modelo de datos conceptual para cada vista de usuario. Para esto se identifican los tipos de entidades, se revisa si las entidades son relaciones o atributos, se procede también a eliminar relaciones recursivas. Con esto se comienza con el primer bosquejo del E-R, a la vez se determinan dominios de atributos, se identifican claves candidatas para cada identidad y si hay varias se elige la clave primaria.

- **Diseño lógico:**

En esta fase se comienza con la transformación del modelo de datos local conceptual al lógico. Para esto de eliminan las relaciones de mucho a muchos, relaciones recursivas y se realiza la normalización, para finalmente obtener el E-R definitivo. No obstante, también se efectúa la integración de la base de datos, obteniendo así, una base de datos lógico global.

- **Diseño Físico:**

En esta etapa se traduce el diseño lógico global para el sistema de administración de base de datos especificado. Finalmente, se realizan mecanismos de seguridad para la base de datos.

4.4.2 Diseño sitio web.

En esta fase se encuentra el “proyecto” según la estrategia del diseño web. En este se realiza un levantamiento exhaustivo de los requerimientos del sistema y la información disponible se trabaja con el cliente con el objeto de evitar contratiempos.

El proyecto comprende los siguientes pasos:

- Organización de la información.
- Diseño de los sistemas de navegación.
- Diseño de sistemas de rotulación.
- Inventarios de las páginas web.
- Diseño físico del sitio web.

4.5 Selección de DBMS.

Esta etapa ocurre entre el diseño conceptual y diseño lógico de la base de datos. El objeto de esta fase es elegir correctamente y tener bien definido los requerimientos para proceder a elegir el DBMS.

4.6 Prototipo.

Esta etapa consiste en un modelo funcional de la aplicación de base de datos. Esto permite identificar mejor los requerimientos de la empresa y a la vez se aclara e identifican mejor los rasgos del sistema. No obstante esta etapa es opcional dentro de esta metodología.

4.7 Implementación.

4.7.1 Implementación base de datos.

Se realiza la parte física de la base de datos con el diseño de aplicaciones. En otras palabras se crean los programas de aplicación que interactúan con la base de datos. A la vez se implementan ciertos controles de seguridad y integridad de las aplicaciones.

4.7.2 Implementación sitio web.

Aquí el sistema toma cuerpo real y se convierte en un objeto funcional. Se comienza el trabajo con la redacción, gráfica y programación. Se programan

las consultas a la(s) base(s) de datos a la vez, se hace toda la gráfica desde los marcos generales hasta las imágenes individuales de cada página.

4.8 Conversión y carga de datos.

En esta etapa se realiza el traspaso de los datos, en este caso de las planillas Excel. En esta etapa se procede a cargar los datos al nuevo sistema.

4.9 Prueba.

Se procede en esta parte a ejecutar el sistema con el fin de encontrar algunos errores. Las pruebas que se hacen son limitadas, ya que de lo contrario se harían muy extensas y complicadas de seguir.

4.10 Difusión sitio web.

Se realiza la difusión del sitio web por medio de algunos canales habituales como por ejemplo en listas de interés, índices apropiados dentro de Internet, como también la inclusión en boletas o facturas de la dirección electrónica.

4.11 Mantenimiento operacional.

4.11.1 Mantenimiento operacional de la base de datos.

Consiste en mantener el sistema después de la instalación. Esto comprende el monitoreo del rendimiento del sistema, como también mantener y actualizar la aplicación de la base de datos cuando sea necesario.

4.11.2 Mantenimiento o actualización sitio web.

Comprende cambios según los requerimientos que se vayan dando en el transcurso del tiempo.

4.12 Evolución del sitio web.

Es más bien un proceso. Aquí se tiene en cuenta el análisis de acceso al sitio y uso de éste.

5. Recursos.

Para llevar a cabo la realización de éste proyecto se necesitan varios recursos. Estos recursos se detallan a continuación, como recursos de hardware y recursos de software.

5.1 Hardware.

Existen distintos tipos de hardware que en conjunto permiten obtener la base para comenzar a armar el sistema.

A continuación, se detalla el hardware necesario para implementar la Intranet.

5.1.1 Hardware del servidor.

El servidor se caracteriza por lo siguiente:

- Pentium III de 700Mhz.
- Disco duro de 20Gb.
- Memoria RAM 256Mb.

- **Forma de comunicación del sistema con el exterior.**

La comunicación del sistema con el exterior se realiza a través de un enlace dedicado punto a punto (Frame Relay), que tiene asignación y activación de una IP fija.

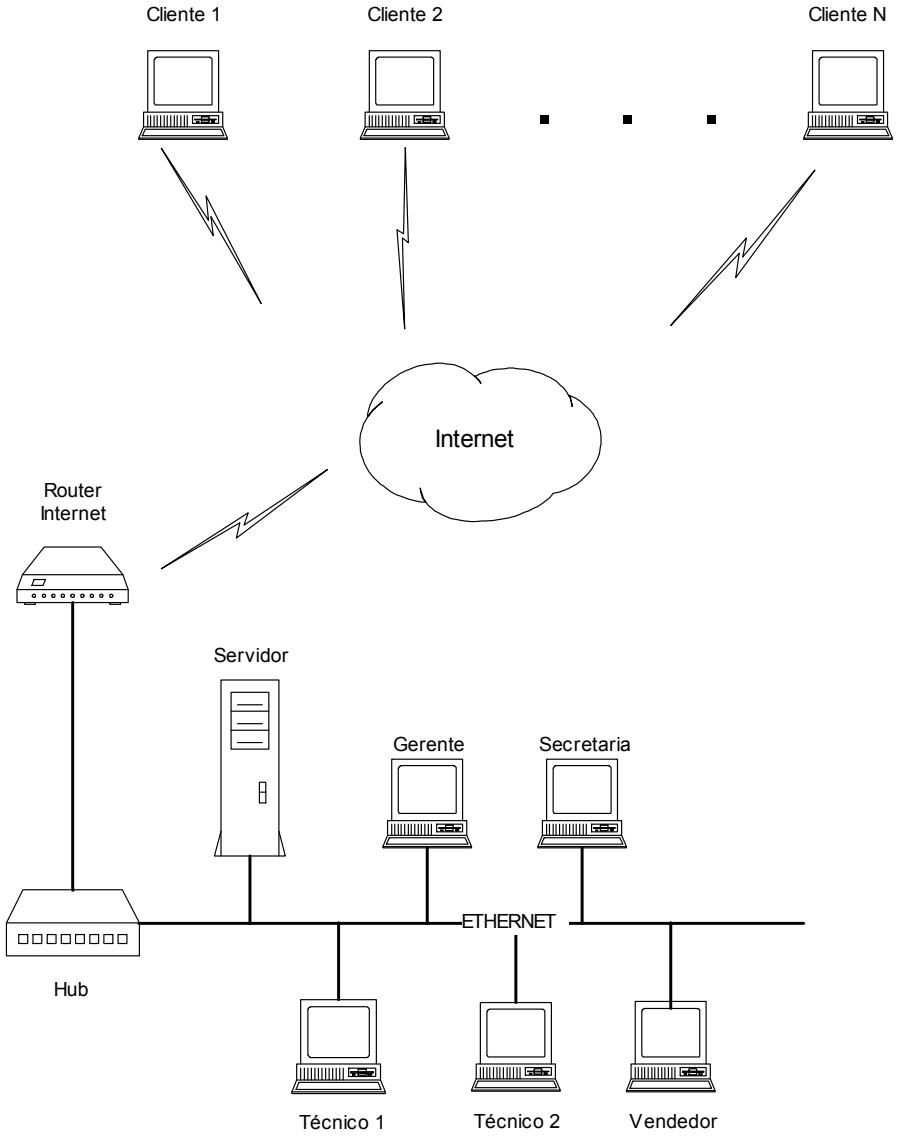
Los requerimientos básicos de este servicio son los siguientes:

- Línea digital punto a punto 128 Kbps (proporcionada por el proveedor de servicios).
- Router marca Cisco (Que es proporcionado por el proveedor de servicios)
- Red LAN Ethernet.
- Protocolo TCP/IP.
- Sistema operativo Windows NT 4.0 a 6.0, Servidor Linux Redhat 6.1 o servidor Unix.

El sistema operativo que se eligió para el sistema fue el de Windows NT Server 4.0.

A continuación, en la figura Nº 3, se muestra el esquema de comunicación de la empresa con el exterior.

Figura Nº 3. Esquema de comunicación de la empresa con el exterior.



5.1.2 Hardware de la empresa.

Este hardware se refiere a los distintos equipos computacionales que se encuentran en la empresa. Las características de estos equipos son las siguientes:

Tabla Nº 2. Características de computadores Empresa Procom.

Características de computadores Empresa Procom			
Nombre Equipo	**Procesador / velocidad**	**Memoria Ram**	**Disco Duro**
Pentium	Pentium II 400 Mhz	64 MB	4.3 GB
Acer Power	Celeron 300 Mhz	32 MB	3 GB
Acer Power	Celeron 300 Mhz	32 MB	3 GB
IBM 300 GL	Pentium MMX 200 Mhz	64 MB	3 GB
IBM 300 GL	Celeron 400 Mhz	64 MB	10 GB
Servidor Pentium	Pentium III 700 Mhz	256 MB	20GB

5.1.3 Hardware de red.

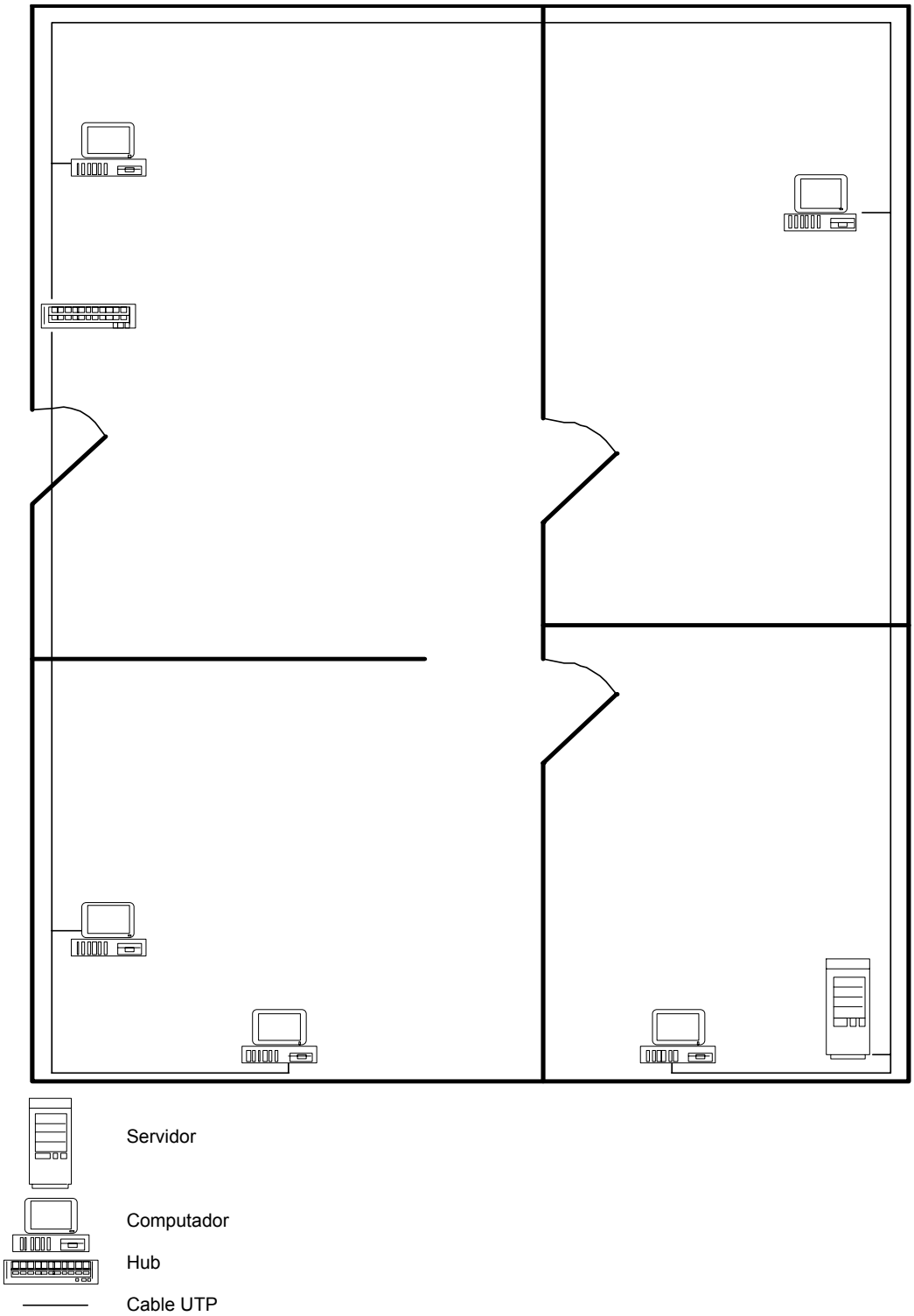
La red esta compuesta por una serie de elementos y equipos:

- Computadores.

- Servidor de red.
- Hub.
- Impresoras.
- Tarjetas de red.
- Cable UTP.
- Conector RJ 45.
- Rosetas.

A continuación, se muestra en la figura Nº 4 la red de la empresa.

Figura 4. Red computacional inicial de la empresa Procom.



5.2 Software.

Para la realización del sistema se ocuparon distintos software en el servidor.

5.2.1 Software del servidor.

- Sistema operativo: Microsoft Windows NT Server 4.0.
- Base de datos: Microsoft SQL Server 7.0.
- Servidor web: Microsoft Internet Information Server (IIS).
- Generador de aplicaciones: Microsoft FrontPage 98, EditPlus y Macromedia Flash 5.0.
- Browser: Microsoft Internet Explorer 5.5.
- Firewall ZoneAlarm.

5.2.1.1 Aspectos generales software del servidor.

- **Microsoft Windows NT Server 4.0:**

Este sistema operativo se caracteriza por estar orientado a red, ya que con él se obtiene una mayor seguridad en el tratamiento e ingreso a los datos y aplicaciones.

Los requisitos mínimos para la instalación de este sistema operativo como servidor, con SQLServer 7.0 y un servidor web son:

- Procesador Pentium II 400Mhz.
- 256 Mb en memoria RAM.
- 4Gb de espacio disponible en disco duro.
- Tarjeta de video AGP O PCI 4 Mb.
- Dispositivo de conexión a Internet.
- Tarjeta de red.

- **Microsoft SQL Server 7.0:**

Es una base de datos relacional, destinada a aceptar aplicaciones cliente servidor. Esta base de datos reside en el servidor y se caracteriza por proporcionar a los administradores de las bases de datos todas las herramientas para poder manejarla correctamente.

Ahora los requisitos mínimos que se deben de tomar en cuenta para la instalación de este software son:

- Un sistema Intel o compatible (Pentium 166 Mhz o superior), o un sistema DEC ALPHA o compatible.
- Un espacio mínimo de 65 Mb para la instalación mínima.

- Una memoria RAM mínima de 32 Mb.
- Unidad de CD.
- Adaptador de red apropiado.

- **Características de algunos componentes web.**

Con el nacimiento de Internet se abrieron muchas posibilidades. Surge el HTML (HyperText Markup Language), que es el lenguaje básico para hacer archivos de texto con hipervínculos, publicarlos en un servidor web y transportarlos a diferentes usuarios, mostrándose a través de browsers. Posteriormente surgen CGI, Java, Javascript, ASP entre otros, permitiendo realizar auténticas aplicaciones Cliente-Servidor a través de la web.

- **Active Server Pages (ASP):**

Se trata de páginas web que contienen código que se procesa en el servidor. Este código es una mezcla de HTML y código Script que van entre dos etiquetas (TAGS), ejemplo <% %>. Este código inserto puede ser Visual Basic Script o Javascript.

Una de las tareas más comunes del ASP es la conexión a bases de datos. Generalmente los usuarios hacen una petición a través de un formulario y ésta es procesada del lado del servidor, luego se procesa la petición y se devuelven los resultados como HTML puro. Ejemplo, figura Nº 5.

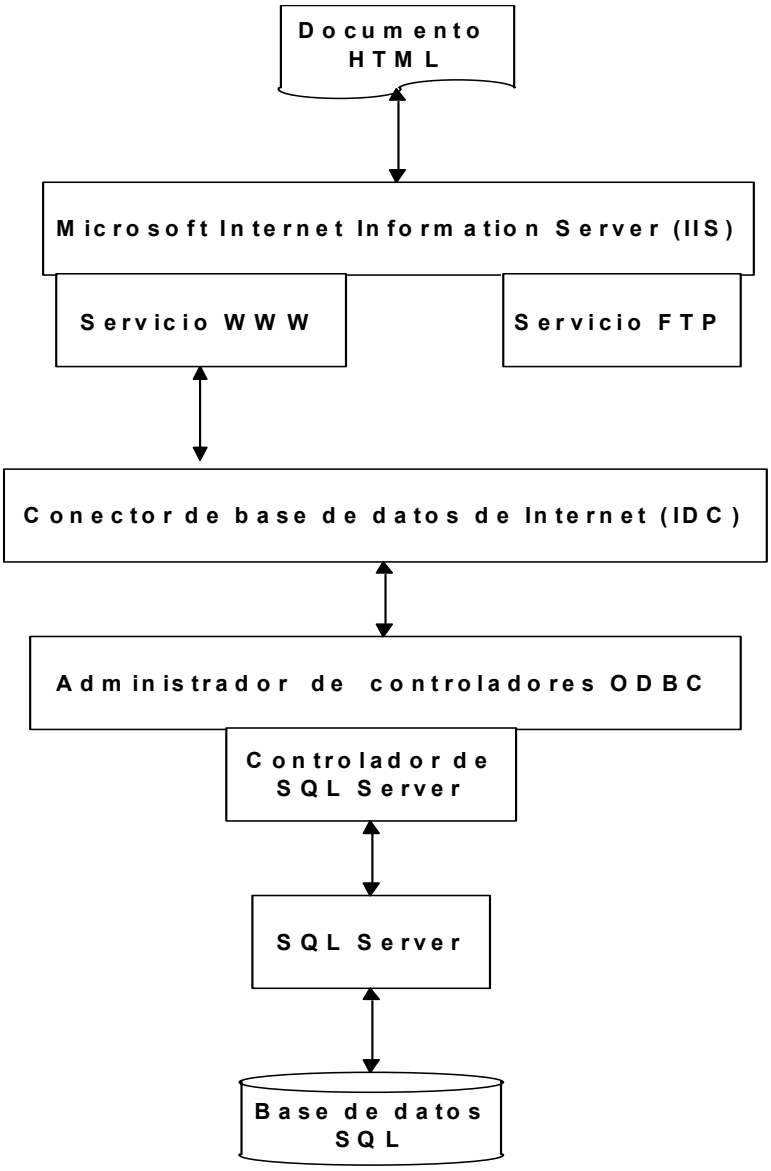


Figura 5. Flujo de datos desde un documento HTML hasta una base de datos y el camino inverso.

- **Internet Information Server (IIS):**

El IIS es un servidor web que soporta comunicaciones en Internet sobre el sistema operativo WindowsNT.

Los servicios web se utilizan cuando un usuario o cliente envía una solicitud al servidor web con el fin de obtener una página web, ejecutar una consulta, modificar un registro o cualquier otra solicitud de información. En otras palabras los servicios web interpretan la solicitud enviada al servidor y devuelven una respuesta apropiada, ya sea ésta otra página web, la confirmación de un cambio entre otras. Ejemplo figura Nº 6.

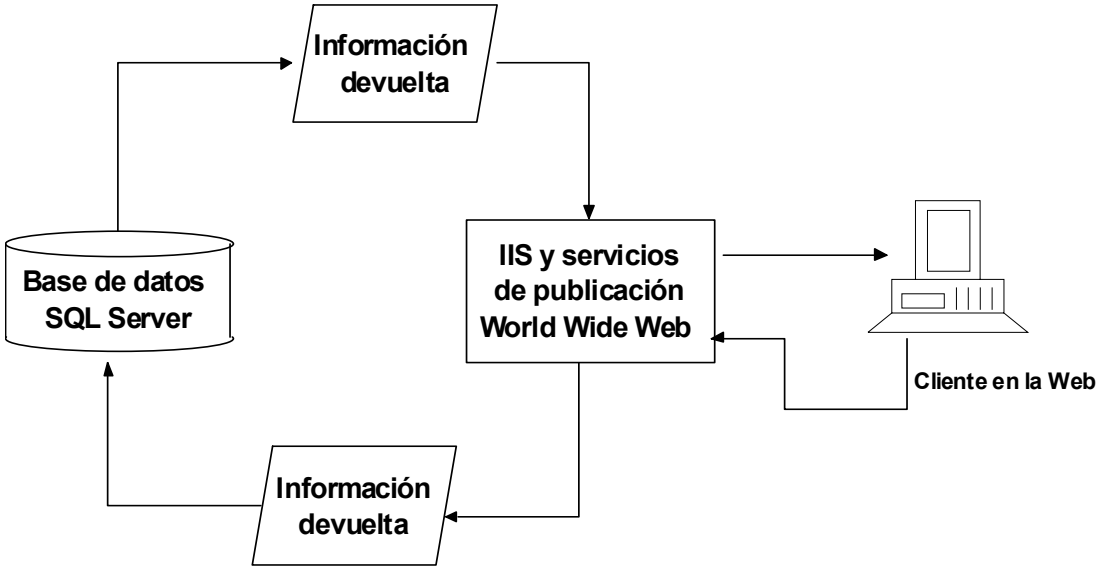


Figura 6. Información que viaja por Internet, a través de IIS. Este envía la solicitud a la base de datos, la que devuelve una respuesta a través del mismo camino, la que se ve por un Browser.

- **Protocolo TCP/IP.**

Este es un protocolo de comunicación que permite conectar la más grande de las redes que es Internet.

6. Planificación de la base de datos.

Para la planificación de la base de datos se debe tomar en cuenta los objetivos que se persiguen en el proyecto. Estos objetivos, que fueron descritos con anterioridad, permiten desarrollar y enfocar correctamente lo que se refiere a las etapas posteriores del sistema.

Lo que se persigue dentro de esta etapa es tener claro los recursos con los que se cuenta para la construcción del sistema, no obstante también se evalúan sistemas de información actuales para obtener así, cuáles sistemas son los que operan, si los hay, y ver que es lo que cubren en cuanto a las necesidades actuales de la empresa.

6.1 Evaluación de sistemas de información actuales.

Al momento de la evaluación dentro de la empresa se constató de que en ella el tratamiento de la información se hace en torno a planillas Excel, por lo tanto, esta empresa no contaba con un sistema que mejore las respuestas a diversos trabajos que se desarrollan. No obstante, se considera que almacenar datos importantes en planillas Excel es relevante, sin embargo, éste no es el método más adecuado para agilizar las búsquedas y respuestas de información; debido a esto surge este sistema.

En lo referente a las planillas Excel, éstas contenían información con respecto a situación de pagos de los clientes, listas de precios, información relevante de clientes entre otras cosas.

6.2 Recursos.

Para poder dar vida a un sistema se deben tener en cuenta algunos recursos que permitan cumplir con los diversos objetivos que se plantean al comienzo del desarrollo de un proyecto.

Estos recursos comprenden tanto los recursos humanos como los recursos monetarios.

6.2.1 Recursos Humanos.

Para llevar adelante el cometido, en este caso el sistema, se debe tomar en cuenta el recurso humano.

Dentro del desarrollo de un sistema intervienen diversas personas en formas distintas, no solamente los que realizan el sistema. Esto quiere decir, que intervienen y aportan al desarrollo personas de distintas áreas de trabajo, o sea desde niveles superiores a inferiores.

6.2.2 Recursos financieros.

Este tipo de recursos se traduce en dinero.

Al momento de construir un sistema se debe tener en claro los costos que intervienen en él. Estos costos en un lapso de tiempo se traducen o transforman en beneficios.

Costos:

- Costos de desarrollo.

  Este es el costo del equipo asignado al desarrollo del sistema.

  Costos del personal encargado del desarrollo del sistema(horas-hombre).

- Costos de instalación del sistema.

  Dentro de estos costos se tiene:

  - Costos de capacitación del usuario, esto corresponde al personal de la empresa.
  - Costo de poblamiento de la base de datos, lo que debería realizarse sólo una vez para la base de datos centralizada.
  - Costo de instalación, esto es referente al hardware y software que se emplean para poder instalar el sistema en la empresa.
  - Costo de la puesta en marcha del sistema.

- Costos operacionales.

  Estos consideran costos de mantenimiento tanto al software como hardware. Dentro de esto se encuentra el costo al mantenimiento de la base de datos, que por cierto se reduce considerablemente al tratarse de una base de datos única.

Beneficios:

Todo costo tiene como objetivo, generar un beneficio. Dentro de los beneficios que surgen se encuentran:

- Beneficios relacionados con el mantenimiento y búsqueda de registros.
- Obtención de registros en forma más rápida, lo que contribuye al aumento en la velocidad de las actividades.
- Posibilidad de cambios de registros en la base de datos, lo que contribuye a la reducción de costos.
- Mejoras de la seguridad del almacenamiento de datos, reduciendo así errores y costos.
- Posibilidad de analizar registros almacenados, contribuyendo a la reducción de errores.
- Aumento de la capacidad para almacenamiento de registros en términos de espacio y costo.

7. Definición del sistema.

Este es un sistema orientado a web, cuyo objetivo principal es ser una alternativa de comunicación entre clientes y servicio técnico.

Ahora, más allá de una definición concreta, hay que ver cuál es el alcance de este sistema, como también la forma que se comunica con las áreas.

7.1 Alcance del sistema.

El alcance que este sistema tiene, parte desde beneficiar a clientes como a los empleados de la empresa.

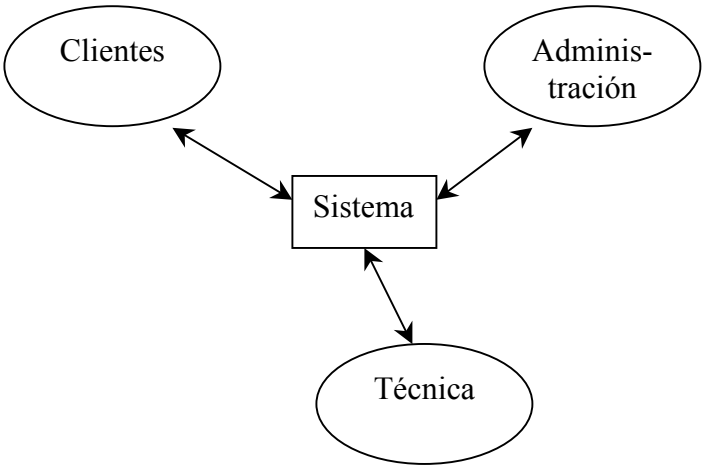
Esto en términos más específicos, quiere decir que los clientes pueden realizar diversas consultas y también solicitar algunos trabajos. Esto posibilita que el cliente se acerque de otra forma, como es la vía web, a la empresa; abarcando así una mayor comunicación. No obstante, el tener un sistema que permita interactuar vía web, ayuda a difundir de otra forma el servicio que se ofrece.

Por otro lado, un alcance más de este sistema se refleja en el hecho de que la empresa en cuestión se incorpora a una serie de empresas que utilizan la vía web para realizar sus trabajos, produciendo esto una mayor igualdad y competitividad con sus pares, a lo que se refiere a la entrega de un producto, siendo en este caso, el servicio al cliente.

7.2 Forma de comunicación con las áreas.

Dentro de la empresa se pueden apreciar tres áreas Figura Nº 7, las cuales son: Administración, técnica y clientes.

Figura Nº 7. Áreas con las que se relaciona el sistema.



8. Análisis y colección de requerimientos.

8.1 Análisis y colección de requerimientos base de datos.

Los pasos que se utilizan al momento de la recolección de datos son los siguientes:

- Observación:

La observación permite ver como se opera dentro de la empresa, con el objeto de comenzar a constatar visualmente algunas anomalías, si las hubiera. Esto claro, sin tener contacto o interacción con los empleados. Luego se procede a realizar algunas entrevistas con el objetivo de reunir información de las operaciones que se hacen y por consiguiente, obtener requerimientos que posteriormente son verificados a través de un seguimiento.

- Entrevista:

El tipo de entrevista que se procede a realizar es en base a un cuestionario con preguntas fáciles de responder. Estas preguntas fueron dirigidas a las áreas involucradas en la empresa (Administración y técnica).

Durante las entrevistas se procedió:

1. Confirmar la función que desempeña dentro de la empresa a la persona que se le pregunta.

2. Se consultó a cada entrevistado la forma en que éstos operan tras distintos acontecimientos que éstos deben de enfrentar según su puesto de trabajo.
3. También se consultó por la documentación con la que trabaja cada persona y los procedimientos que éste efectúa.
4. Se procede a consultar por la forma en que guardan los datos que se van obteniendo después de cada operación.
5. Se preguntó a la vez, por las formas de interacción que se tienen con clientes.
6. Se aclaran también en estas entrevistas por respuestas dudosas o no entendidas.
7. Dentro de estas entrevistas, se procede a recolectar documentos, ya sean estos copias de facturas, boletas, planillas u otro documento con datos relevantes.

A continuación, se muestra el formato del cuestionario.

- Identificación de la persona entrevistada.

  Nombre:

  Cargo:

  Unidad:

  Fecha:

- Numeración encuesta.

  Numero encuesta:

- Descripción y objetivo de la encuesta.

- Contenido del cuestionario.

  Pregunta 1
  *
  *
  Pregunta n.

- Observaciones.

En el **anexo 1**, se muestra una serie de documentos recolectados con los cuales la empresa realiza trabajos.

8.2 Anteproyecto del desarrollo del sitio web.

En el anteproyecto del desarrollo del sitio web, lo que se pretende es obtener requerimientos pensando en el diseño del sitio web. Bajo este punto de vista, se comienza a realizar audiencias con el cliente, con el propósito de obtener una idea clara de lo que se pretende. Esto quiere decir que las preguntas van orientadas al tipo de gráfica, lenguaje entre otras cosas.

Finalmente, se le entrega al cliente las ideas básicas de lo que sería la realización del sistema.

8.2.1 Entrevistas con el gerente de la empresa.

En la primera reunión, el gerente expone la forma del diseño aproximada que quiere del sistema, dando a conocer ciertos elementos de la gráfica como por ejemplo colores, ciertas opciones en cuanto a la configuración del menú del sitio entre otras cosas.

Una vez que se recolecta los objetivos e impresiones que persigue la persona interesada para su empresa, se está en condiciones para darle a conocer las herramientas con las que se trabajará. Esto es el tipo de lenguaje que se usará, la plataforma, herramientas de programación y los requisitos mínimos que tiene que tener el servidor para dar vida al sistema, como por supuesto dar a conocer los tipos de conexión a Internet que se pueden usar.

En las posteriores reuniones se dan a conocer los avances de la gráfica del sitio web y a la vez se toman o recolectan las opiniones, con el objeto de ir modificando el sitio según los requisitos del cliente.

8.2.2 Requerimientos del usuario.

A continuación, se procede a enumerar los siguientes requerimientos como resultado de las entrevistas y observación de documentos.

Enumeración de requerimientos (**R**):

R(1): Permitir la búsqueda de información de clientes externos, como calendario de trabajos, respuestas de cotizaciones, trabajos.

R(2): Permitir la búsqueda de información de personal de la empresa, por productos, ventas, facturas, órdenes de trabajos, cotizaciones, pedidos, fechas, calendarios.

R(3): Permitir generar una comunicación externa de datos a clientes por consultas, cotizaciones, productos, trabajos.

R(4): Generar información interna con los datos de clientes externos, en cuanto a ventas, trabajos, cotizaciones.

R(5): Generar información interna con datos de productos, pedidos, facturas.

9. Diseño de la base de datos.

9.1 Diseño conceptual de la base de datos:

Se caracteriza porque maneja las entidades del mundo real como entidad. El modelo relacional implica varias cosas:

1. Entidades.
2. Relación.
3. Atributos.

9.1.1 Especificación de entidades.

En la siguiente **Tabla Nº 3** se identifican y describen entidades.

Tabla Nº3. Especificación de entidades.

Nombre Entidad	Descripción	Ocurrencia
CLIENTE	Empresas que solicitan servicios habituales.	Ingreso de empresas que necesitan apoyo técnico.
CALENDARIO	Muestra fechas de mantenciones empresas u trabajos.	Ingreso de fecha de trabajo a realizar.
PRODUCTO	Entidad que muestra productos a vender	Ingreso de un nuevo ítem producto.

COTIZACION	Entrega información que requiere el cliente.	Ingreso de inquietudes de clientes acerca de productos.
ORDEN_DE_TRABAJO	Muestra servicios que requiere el cliente.	Ingreso de una petición de trabajo.
PEDIDO	Entrega información de fecha de pedidos y situación de estos, como también los proveedores a quienes se les compra.	Pedido de productos para la venta.
FACTURA	Entrega información del número y que origina el documento entre otras cosas.	Ingreso de numeración del tipo de documento forma pago, estado y fecha de realización de factura.
DETALLE_FACTURA	Muestra información de productos que se venden y trabajos a clientes.	Detalle de la realización de venta o trabajo realizado.
VENTA_DIRECTA	Muestra el número de ventas a clientes.	Ingreso de numeración y cliente que compra.

autorizacion	Esta entidad guarda información de los usuarios que trabajan y ocupan el sistema dentro de la empresa.	Ingreso de un nuevo usuario con sus debidos permisos.

9.1.2 Identificación de relaciones.

Las relaciones son asociaciones entre entidades. Estas asociaciones cuyo nombre se conoce como cardinalidad pueden ser de varios tipos como por ejemplo, Uno a Uno (1:1), Uno a Muchos (1:N), etc. Por otro lado, la Existencia se refiere si la relación es mandatoria (M), la cuál indica que siempre debe ir, o opcional (O), la que puede o no ir.

En la siguiente Tabla Nº 4, se muestra las características de cada relación.

Tabla Nº 4. Identificación de relaciones.

Entidad	**Relación**	**Entidad**	**Cardina-lidad**	**Existen-cia**
CLIENTE	consulta	CALENDARIO	1:N	O:O

CLIENTE	genera	ORDEN_DE_TRABAJO	1:N	M:O
CLIENTE	realiza	COTIZACION	1:N	O:M
ORDEN_DE_ TRABAJO	genera	FACTURA	N:1	M:M
FACTURA	tiene	DETALLE_FACTURA	1:N	M:M
PEDIDO	requiere	PRODUCTO	N:M	M:O
CLIENTE	genera	VENTA_DIRECTA	1:N	M:O
VENTA_DIRECTA	tieneasoc	FACTURA	N:1	M:M
PRODUCTO	incorpora	DETALLE_FACTURA	N:1	M:M

9.1.3 Identificación y detalle de atributos de entidades.

Nomenclatura:

Valor Nulo	Longitud	Valor Defecto	Derivado	Multivalorado
VN	**L**	**VD**	**D**	**M**

En la siguiente Tabla Nº 5, se describen los atributos de las distintas entidades que forman parte de este sistema.

Tabla Nº 5. identificación y detalle de atributos de entidades.

Entidad	Nombre Atributo	Descripción	Tipo Dato	VN	L	VD	D	M
CLIENTE	rut_ cliente	Rut de la Empresa.	numeric	no	8	no	no	no
	dig_ cliente	Dígito verificador.	char	no	1	no	no	no
	nombre _cliente	Nombre empresa.	char	no	30	no	no	no
	direc ción_ cliente	Dirección empresa.	char	no	30	no	no	no
	ciudad_ cliente	Ciudad de la empresa.	char	no	20	no	no	no
	contacto _cliente	Persona a contactar.	char	no	30	no	no	no
	teléfono _cliente	Teléfono de empresa.	numeric	no	10	no	no	no
	email_ cliente	E-mail de la empresa.	char	sí	30	no	no	no

	pass word_ cliente	Clave secreta para el cliente.	char	no	10	no	no	no
CALENDARIO	correl	Correlativo.	numeric	no	8	no	no	no
	fecha	Fecha de trabajos.	datetime	no	8	no	no	no
	tipo_ trabajo	Tipo de servicios.	char	no	35	no	no	no
	equipo	Hardware de la empresa.	char	no	35	no	no	no
	rut_ cliente	Rut cliente empresa	numeric	no	8	no	no	no
PRODUCTO	cod_ prod	Código del producto.	numeric	no	10	no	no	no
	tipo_ producto	Productos de computación.	char	no	30	no	no	no
	detalle_ producto	Característi-ca del producto.	char	no	110	no	no	no

	valor_ original_ SI	Valor proveedor.	real	no	24	no	no	no
	valor_ original_ CI	Valor proveedor con IVA.	entero	no	10	no	no	no
	valor_ procom_ CI	Valor del producto de Procom.	entero	no	10	no	no	no
COTIZACION	numero _doc	Número del documento.	numeric	no	10	no	no	no
	fecha	Fecha de la cotización.	datetime	no	8	no	no	no
	detalle_ cotiza ción	Detalle del producto que se cotiza.	text	no	16	no	no	no
	estado_ cotiza ción	Situación de la cotización.	char	no	9	sí	no	no

	res puesta_ cotiza ción	Respuesta de la cotización por parte de la empresa.	text	sí	16	no	no	no
	rut_ cliente	Rut cliente empresa.	numeric	no	8	no	no	no
PEDIDO	num_ pedido	Número del pedido efectuado.	numeric	no	10	no	no	no
	provee dor	Nombre del proveedor al que se le piden productos.	char	no	30	no	no	no
	situa ción_ pedido	Situación del pedido si esta recibido o pendiente.	char	no	9	sí	no	no

	fecha_ pedido	Fecha en que se hace el pedido.	datetime	no	8	no	no	no
ORDEN_DE_ TRABAJO	num_ orden	Número de la orden de trabajo que se emite.	numeric	no	6	no	no	no
	fecha_ emisión	Fecha en que se emite la orden de trabajo.	datetime	no	8	no	no	no
	tipo_ trabajo	Tipo de trabajo que se efectuaría.	char	no	30	no	no	no
	detalle_ orden	Muestra la especifica- ción de la orden de trabajo.	text	no	16	no	no	no

	solicitado_por	Persona que solicita el trabajo.	char	no	30	no	no	no
	nombre_técnico	Nombre del técnico que realiza el trabajo.	char	sí	30	no	no	no
	trabajo_realizado	Detalle por parte del técnico del trabajo realizado.	text	sí	16	no	no	no
	fecha_trabajoi	Fecha en que el técnico comienza a realizar el trabajo.	datetime	sí	8	no	no	no
	fecha_trabajot	Fecha en que el técnico termina el trabajo.	datetime	sí	8	no	no	no

	aprobado_por	Nombre de persona que aprueba el trabajo.	char	sí	30	no	no	no
	validez_orden	Muestra si la orden es válida o no.	char	no	1	sí	no	no
	situación_trabajo	Describe si el trabajo está pendiente o no.	char	no	9	sí	no	no
	situación_pago	Describe si el pago está pendiente o ha sido facturado.	char	no	9	sí	no	no
	rut_cliente	Rut pertenece al cliente de la empresa.	numeric	no	8	no	no	no

	num_ doc	Número factura	numeric	no	10	no	no	no
FACTURA	num_ doc	Número del documento que se emite (factura).	numeric	no	10	no	no	no
	origen_ doc	Se refiere a que originó la factura (orden de trabajo o venta directa).	char	no	13	no	no	no
	fecha	Fecha en que se realiza factura.	datetime	no	8	no	no	no
	rut_ cliente	Rut que pertenece a clientes de la empresa.	numeric	no	8	no	no	no

	neto	Valor original del producto.	real	no	24	no	no	no
	iva	Porcentaje que se agrega por conceptos de servicios por ley.	real	no	24	no	no	no
	total	Valor final que se cobra por conceptos de servicios y ventas.	entero	no	10	no	no	no
	estado	Estado de la factura.	char	no	9	no	no	no
	forma_ pago	Forma en que se realizan los pagos a la empresa.	char	sí	10	no	no	no

DETALLE_ FACTURA	num_ doc	Número del documento (factura).	numeric	no	10	no	no	no
	cod_ prod	Codigo del producto.	numeric	no	10	no	no	no
	detalle_ producto	característica del producto.	char	no	110	no	no	no
	cantidad	Cantidad de productos.	numeric	no	4	no	no	no
	valor_ procom	Valor que procom le da a los productos.	entero	no	10	no	no	no
	espec ficación	Especifica-cación más concreta para ciertos item.	texto	sí	16	no	no	no
VENTA_ DIRECTA	num_ venta	Número de venta actual.	numeric	no	6	no	no	no

	rut_ cliente	Rut perteneciente al cliente de la empresa.	numeric	no	8	no	no	no
	num_ doc	Número perteneciente a la factura.	numeric	no	10	no	no	no
autorizacion	login	Nombre de seguridad del usuario.	char	no	13	no	no	no
	pass-word	Contraseña del usuario.	char	no	10	no	no	no
	nivel	Nivel de seguridad para cada usuario.	numeric	no	5	no	no	no

9.1.4 Determinación de los dominios de atributos.

El dominio consiste en un rango o conjunto de valores que puede tomar cierto atributo.

A continuación, en la tabla Nº 6 se muestra el dominio de los atributos.

Tabla Nº 6. Determinación de dominios de atributos.

Nombre dominio	Característica	Ejemplo
rut_cliente	numérico	12935066, 13123040
dig	alfanumérico	0,1,k,2.....
nombre_cliente	alfabético	Dolly, Mundo Joven, Weiztler....
dirección_cliente	alfanumérico	Urmeneta 320, Antonio Varas 440...
ciudad_cliente	alfabético	Pto Montt, Osorno,.....
contacto_cliente	alfabético	Victor Vargas,..
teléfono_cliente	numérico	322410, 285847, ...
email_cliente	alfanumérico	dolly@telsur.cl, jorge@telsur.cl
password_cliente	alfanumérico	Rojo2, scorpion,…

correl	numérico	1,2,3,4,5,6.....
fecha	fecha	12-03-1999
tipo_trabajo	alfabético	Mantención, inst. de software
equipo	alfabético	Impresora ,computador...
cod_prod	numérico	5,300,301,302....
tipo_producto	alfabético	Tarjeta red, impresora,...
detalle_producto	alfanumérico	Motherboard 748LMRT….
valor_original_SI	numérico	4680, 15990…
valor_original_CI	numérico	325444, 39990,….
valor_procom_CI	numérico	43250, 19990....
numero_doc	numérico	1,2,3,4.....
fecha	fecha	13-11-1985
detalle_cotización	alfanumérico	Computador Pentium III 700 MHZ
estado_cotización	alfabético	Pendiente, Enviada

respuesta_cotización	alfanumérico	Computador Pentium valor $500000
num_orden	numérico	1,2,3,4,5,.....
fecha_emisión	fecha	12-03 -1999
tipo_trabajo	alfabético	Mantención impresoras....
detalle_orden	alfanumérico	Instal. Tarjeta red 10/100 encore
solicitado_por	alfabético	Alberto Cienfuegos
nombre_técnico	alfabético	Juan Pérez
trabajo_realizado	alfanumérico	Configuración equipo secretaria...
fecha_trabajoi	fecha	18-09-2002
fecha_trabajot	fecha	19-09-2002, ...
aprobado_por	alfabético	Juan Carlos Aravena, ...
validez_orden	numérico	0 , 1
situación_trabajo	alfabético	Pendiente, Realizado
situación_pago	alfabético	Facturado, Pendiente

num_pedido	numérico	1,2,3,4,5,....
proveedor	alfabético	Reflychile,....
situación_pedido	alfabético	Recibido, Pendiente
fecha_pedido	fecha	10-02-2000
num_doc	numérico	1,2,3, ……..
origen_doc	alfabético	Orden trabajo, venta directa
fecha	fecha	19-08-2002
neto	decimal	20000,00
iva	decimal	24500,00
total	numérico	350000
estado	alfabético	Emitida, Pagada, Nula
forma_pago	alfabético	Efectivo, Cheque
cantidad	numérico	1,2,3,...
especificación	alfanumérico	19-05-2002

num_venta	numérico	1,2,3,4,5...
usuario	alfabético	c_aguilar, rpinto,...
password	alfanúmerico	cyeg1j, ekjypvyj..
nivel	numérico	1,2,3,4

9.1.5 Identificación de claves candidatas y elección de clave primaria para cada entidad.

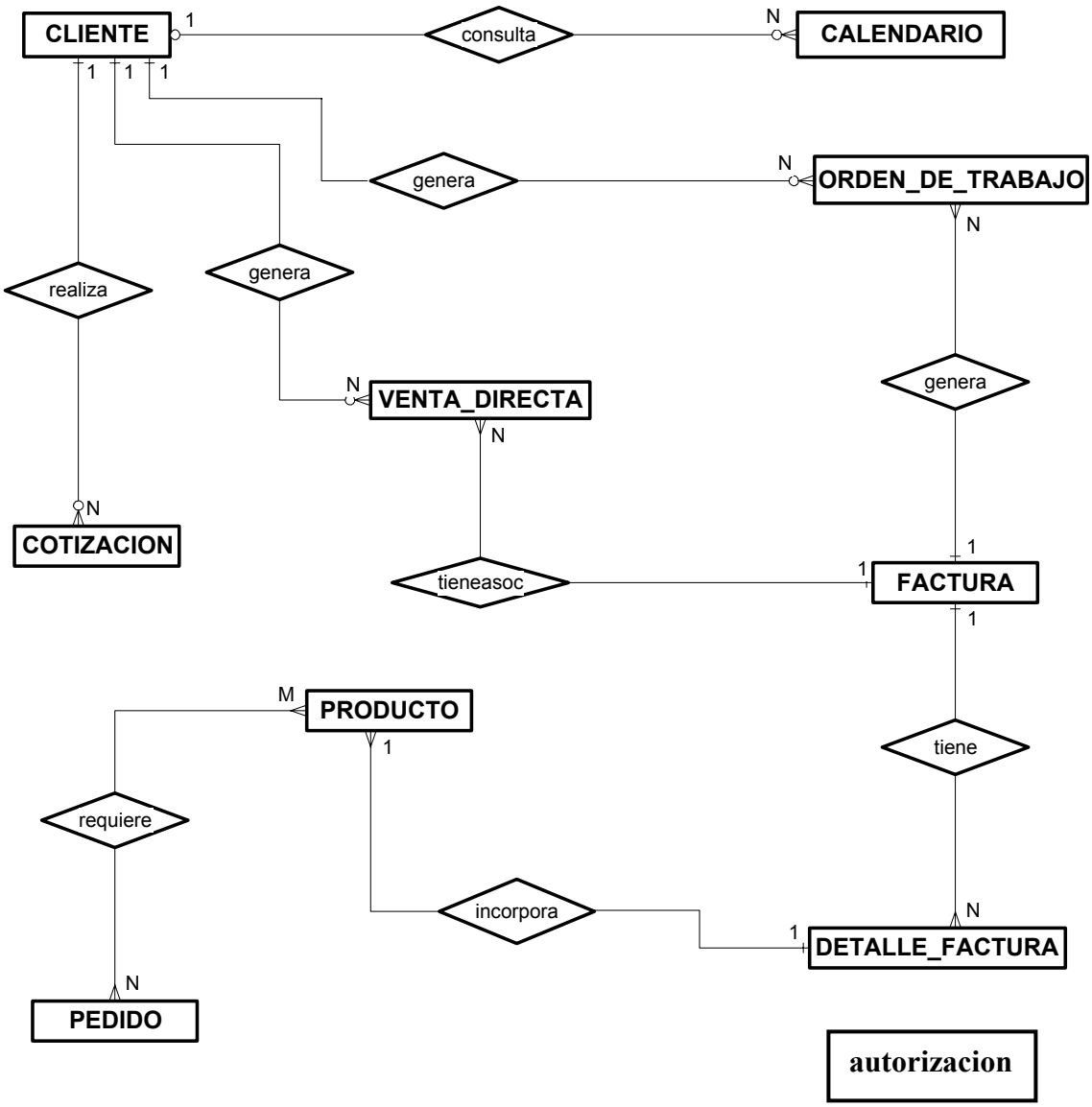
Las claves candidatas son aquellas claves que están formadas por un conjunto mínimo de atributos. De este conjunto mínimo de atributos se elige aquella clave que tenga la menor posibilidad de cambiar y a la vez la que tenga la posibilidad de mantener la condición de ser única. Una vez que se ubica esta clave se procede a nominarla como clave primaria.

A continuación, en la Tabla Nº 7, se procede a identificar claves candidatas y claves primarias.

Tabla Nº 7. Identificación de claves candidatas y elección de clave primaria para cada entidad.

Entidad	Claves Candidatas	Clave Primaria
CLIENTE	rut_cliente	rut_cliente
CALENDARIO	correl, fecha	correl
PRODUCTO	cod_prod, tipo_producto	cod_producto
COTIZACION	numero_doc	numero_doc
ORDEN_DE_TRABAJO	num_orden, tipo_trabajo	num_orden
PEDIDO	num_pedido, fecha_pedido	num_pedido
VENTA_DIRECTA	num_venta	num_venta
FACTURA	num_doc, origen_doc	num_doc
autorizacion	usuario	usuario

Figura Nº 8. Modelo E-R Conceptual preliminar.



9.2 Diseño Lógico de la base de datos:

En el diseño lógico de la base de datos, se comienza con la transformación del modelo de datos local conceptual al lógico. Para esto se debe eliminar las relaciones de mucho a mucho, eliminar relaciones M : N complejas de 3 ó más entidades; también se debe quitar las relaciones recursivas si es que las hay, como también reexaminar relaciones de uno a uno. Luego de esto, comienza la normalización, siendo éstas:

- 1FN (Primera Forma Normal): Esta elimina grupos repitentes.
- 2FN (Segunda Forma Normal): Elimina dependencias parciales en la clave primaria.
- 3FN (Tercera Forma Normal): Elimina dependencias transitivas en la clave primaria.

Posteriormente, se diseña el modelo E-R (Entidad – Relación), se definen limitaciones de integridad como también se destacan las claves primarias y foráneas.

Para finalizar se lleva a cabo la integración de la base de datos donde se obtiene una base de datos lógico global.

9.2.1 Transformación del modelo de datos local conceptual al local lógico.

El objetivo de esta transformación es constatar y refinar el modelo de datos conceptual local y remover aquellas características no deseables, si es que las hay, para posteriormente transformar este modelo al modelo de datos lógico local.

9.2.1.1 Eliminación de relaciones mucho a muchos (M:N).

La eliminación de las relaciones mucho a muchos consiste en lo siguiente:

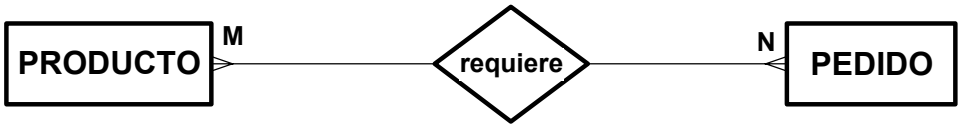
Se descompone la relación identificando una entidad intermedia. Luego, se reemplaza la relación M:N con dos relaciones 1:M asociadas con la nueva entidad.

En el modelo de datos conceptual local, figura Nº 8, se aprecia que existe este tipo de anomalía.

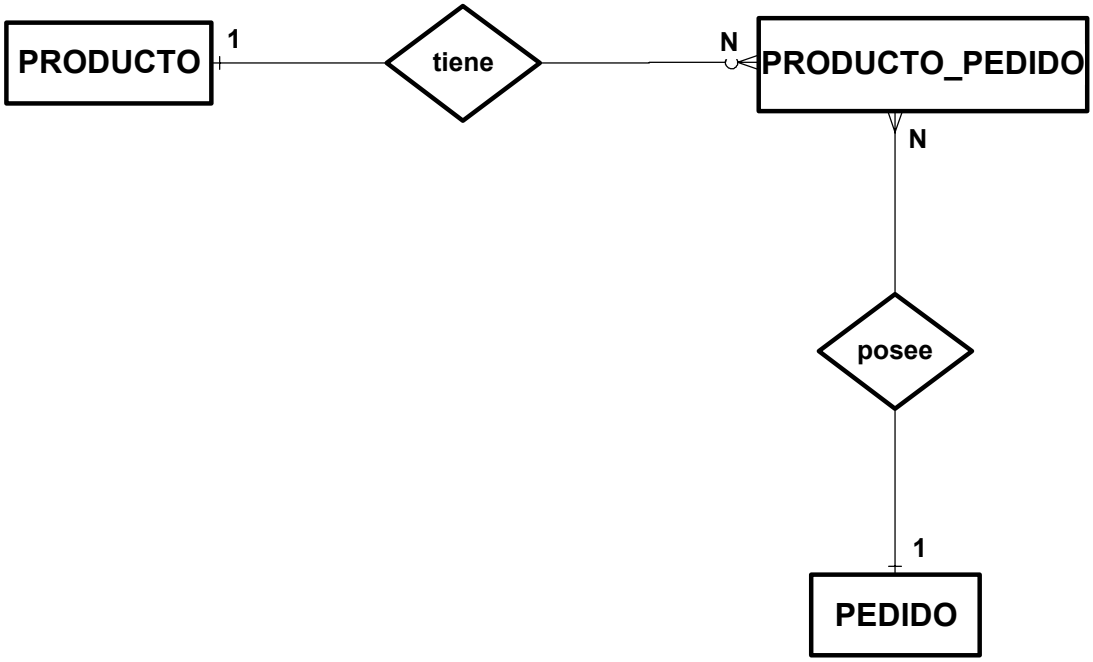
A continuación, en la figura Nº 9, se presenta la solución a la respectiva anomalía.

Figura Nº 9. Eliminación de relación Mucho a Muchos M:N.

Anomalía M:N



Solución



Una vez realizados los pasos correspondientes, se dá solución a esta problemática, cumpliéndose así, el objetivo descrito anteriormente que es la transformación del modelo de datos.

Así como se eliminó las relaciones mucho a muchos, se debe proceder a hacer lo mismo con las relaciones M:N complejas de 3 ó más entidades, pero en esta ocasión no existe tal condición en el modelo E-R conceptual.

9.2.1.2 Eliminación de relaciones recursivas.

Para proceder a eliminar este tipo de relación, se comienza a descomponer la relación identificando una entidad intermedia. Si es una relación recursiva de 1:M, entonces se reemplaza con la entidad intermedia y una nueva relación adicional de uno a uno.

Tomando en cuenta esto se procede a revisar el modelo conceptual local en busca de esta anomalía, la que no se encontró.

9.2.1.3 Derivación de relaciones del modelo de datos lógico.

El objetivo es describir la relación utilizando Database Definition Language (DBDL), o lenguaje de definición de base de datos.

Lo importante de esto es destacar las claves primarias y foráneas. Así se identifican las entidades padres e hijos, siendo las entidades padres las que depositan sus claves primarias en las entidades hijos.

CLIENTE(rut_cliente, dig_cliente, nombre_cliente, dirección_cliente, ciudad_cliente, contacto_cliente, telefono_cliente, email_cliente, password_cliente)

Primary key (rut_cliente)

CALENDARIO(correl, fecha, tipo_trabajo, equipo, rut_cliente)

Primary key (correl)

Foreign key (rut_cliente) reference **CLIENTE** (rut_cliente)

COTIZACION(numero_doc, fecha, detalle_cotizacion, estado_cotizacion, respuesta_cotizacion, rut_cliente)

Primary key (numero_doc)

Foreign key (rut_cliente) reference **CLIENTE** (rut_cliente)

FACTURA(num_doc, origen_doc, fecha, rut_cliente, neto, iva, total, estado, forma_pago)

Primary key (num_doc)

Foreign key (rut_cliente) reference **CLIENTE** (rut_cliente)

ORDEN_DE_TRABAJO(num_orden, fecha_emision, tipo_trabajo, detalle_orden, solicitado_por, nombre_tecnico, trabajo_realizado, fecha_trabajoi, fecha_trabajot, aprobado_por, validez_orden, situacion_trabajo, situacion_pago, rut_cliente, num_doc)

Primary key (num_orden)

Foreign key (rut_cliente) reference **CLIENTE** (rut_cliente)

Foreign key (num_doc) reference **FACTURA** (num_doc)

PEDIDO(num_pedido, situacion_pedido, fecha_pedido, proveedor)

Primary key (num_pedido)

PRODUCTO(cod_prod, tipo_producto, detalle_producto, valor_orignal_si, valor_original_ci, valor_procom_ci)

Primary key (cod_prod)

VENTA_DIRECTA (num_venta, rut_cliente, num_doc)

Primary key (num_venta)

Foreign key (rut_cliente) reference **CLIENTE** (rut_cliente)

Foreign key (num_doc) reference **FACTURA** (num_doc)

DETALLE_FACTURA(num_doc, cod_prod, detalle_producto, cantidad, valor_procom, especificacion)

Primary key (num_doc, cod_prod)

Foreign key (num_doc) reference **FACTURA** (num_doc)

Foreign key (cod_prod) reference **PRODUCTO** (cod_prod)

autorizacion (usuario, password, nivel)

Primary Key (usuario)

9.2.1.4 Normalización del modelo.

El diseño de esquema para generar base de datos relacionales, debe considerar el objetivo de almacenar información sin redundancia innecesaria, pero que a la vez permite recuperar información fácilmente.

- **Ventajas de la normalización.**

La normalización está para prevenir anomalías de actualización, inconsistencia en los datos. El objetivo principal de una normalización es desarrollar una estructura de datos donde la redundancia es minimizada y la consistencia es maximizada.

- **Desventajas de la normalización.**

El problema o la desventaja que tiene la normalización es el acceso, ya que este se torna más lento, por lo que afecta el rendimiento en la base de datos.

El proceso de normalización incluye:

- Primera forma normal 1FN que elimina grupos repitentes.
- Segunda forma normal 2FN que elimina dependencias parciales en la clave primaria.
- Tercera forma normal 3FN que elimina las dependencias transitivas en la clave primaria.

- **Primera forma normal 1FN.**

Una tabla se dice que está en primera forma normal (1FN), si todas las columnas tienen valores atómicos; eso es que no existen grupos repitentes (columnas) dentro de la fila.

La ventaja de la primera forma normal es que las tablas son más simples.
La desventaja es que se duplican datos.

En derivación del modelo de datos lógico, se puede ver que se cumple esta condición, por lo que se cumple con la primera forma normal.

- **Segunda forma normal 2FN.**

Una tabla se dice en segunda forma normal (2FN) si está en primera forma normal (1FN) y cada atributo no-clave es completamente dependiente de la clave primaria.

En la derivación del modelo de datos lógico, se aprecia que cada atributo no clave depende de la clave primaria.

- **Tercera forma normal 3FN.**

Una tabla está en tercera forma normal (3FN), si sólo sí está en 2FN y todos los atributos no clave dependen de manera no transitiva de la clave primaria.

En el tema tratado anteriormente acerca de la derivación del modelo de datos lógico, se puede ver que las distintas tablas cumplen con la 3FN, ya que todas están en 2FN y a la vez cumplen con la condición de que todos los atributos no clave dependen de manera no transitiva en la clave primaria.

9.2.1.5 Validación del modelo contra las transacciones.

El desarrollo de la validación del modelo contra las transacciones se realiza con el objeto de comprobar que el modelo de datos lógico local sea acorde con las transacciones que requieren los usuarios.

Debido a lo anterior, se procede a identificar las transacciones:

T(1): Ingreso de órdenes de trabajo por parte de clientes.

T(2): Ingreso de cotizaciones por parte de clientes.

T(3): Ingreso de clientes.

T(4): Ingreso de productos.

T(5): Ingreso de informes de trabajos.

T(6): Ingreso de facturas.

T(7): Ingreso de calendario de trabajos.

T(8): Búsqueda de trabajos efectuados por parte de clientes.

T(9): Búsqueda de calendario de trabajos por parte de los clientes.

T(10): Búsqueda de respuesta a cotizaciones por parte de clientes.

T(11): Búsqueda de informes técnicos de trabajos por parte de clientes.

T(12): Búsqueda de situación de pagos por clientes.

T(13): Búsqueda de facturas de pagos pendientes.

T(14): Búsqueda de productos y valores por personal de empresa.

T(15): Búsqueda de trabajos realizados y fechas por parte de la empresa.

T(16): Búsqueda de situación pedidos a proveedores.

T(17): Revisión de facturas por concepto de ventas directas.

T(18): Revisión de facturas por concepto de trabajos.

T(19): Búsqueda de situación de trabajos.

Una vez que se identifican las transacciones, se realiza una muestra entre las transacciones y los requerimientos que fueron enumerados en el capítulo Nº 8, sección 8.2.2. En la tabla Nº 8, se muestra las transacciones versus los requerimientos.

Tabla Nº 8. Transacciones versus requerimientos.

T \ R	R(1)	R(2)	R(3)	R(4)	R(5)
T(1)			X		
T(2)			X		
T(3)				X	
T(4)					X
T(5)				X	
T(6)					X
T(7)				X	
T(8)	X				
T(9)	X				
T(10)	X				
T(11)	X				
T(12)	X				
T(13)		X			
T(14)		X			

T(15)		**X**			
T(16)		**X**			
T(17)		**X**			
T(18)		**X**			
T(19)		**X**			

9.2.1.6 Diseño de transacciones.

A continuación, en la tabla Nº 9, se mencionan para cada transacción sus datos de entrada, características funcionales y las salidas correspondientes.

Tabla Nº 9. Diseño de transacciones.

Transacción	Datos de Entrada	Características Funcionales	Salidas
T(1)	Rut del cliente, petición de trabajo, fecha emisión, solicitante.	Se ingresa información por parte de clientes, de órdenes de trabajo, con el objeto de atender posteriormente estos trabajos.	Formulario de ingreso para órdenes de trabajo para el uso de Clientes.
T(2)	Rut del cliente, fecha cotización, detalle de cotización.	Se ingresan los datos que el cliente solicita, para posterior respuesta por parte de la empresa.	Formulario de ingreso de cotizaciones para el uso de Clientes.
T(3)	Rut del cliente, nombre, dirección, ciudad, contacto, teléfono,email , password	Se ingresan registros con la información más relevante de los clientes.	Formulario ingreso de Clientes.

T(4)	Tipo producto, detalle producto, valor producto original.	Se ingresan los productos para su posterior utilización.	Formulario ingreso de productos para el uso del personal Empresa.
T(5)	Nombre del técnico, trabajo realizado, fechas	Se realiza el ingreso de los detalles del trabajo realizado por parte de la empresa para los clientes.	Formulario ingreso de informes de trabajos.
T(6)	Número de factura, origen del documento,fecha, rut cliente, neto, iva, total, forma de pago.	Se realiza el ingreso de las facturas correspondientes a trabajos y ventas.	Formulario para el ingreso de facturas por parte de la Empresa.
T(7)	Fecha de trabajo, tipo de trabajo, equipo o ubicación del trabajo, rut cliente	Se ingresan los calendarios de trabajos a los distintos clientes que los solicitan.	Formulario para el ingreso de calendarios de trabajos.
T(8)	Rut del cliente	Permite a los clientes revisar los distintos tipos de trabajos realizados por parte de la empresa.	Formulario que muestra una lista de los trabajos que a ellos solicitan.
T(9)	Rut del cliente	Se realiza la búsqueda por parte de los cliente, de los distintos trabajos que se realizarán. Mostrándose estos por medio de un calendario.	Formulario que muestra el calendario de trabajos para cada Cliente.
T(10)	Rut del cliente	Se realiza la búsqueda de las respuesta de cotizaciones que la empresa genera a causa de una petición realizada por un cliente.	Formulario que muestra las respuestas de cotizaciones a Clientes.

T(11)	Rut del cliente	Permite búsqueda de los informes técnicos por parte de clientes de aquellos trabajos que se han realizado completamente.	Formulario que contiene el informe técnico de los trabajos efectuados a cada Cliente.
T(12)	Rut del cliente Situación trabajo	Se realiza la búsqueda por los clientes acerca de su situación de pagos.	Formulario que muestra la situación de pagos de cada Cliente.
T(13)	Rut del cliente estado	Búsqueda por parte de la empresa de aquellos clientes con pagos pendientes.	Formulario que contiene las facturas pendientes de Clientes.
T(14)	producto	Búsqueda por parte de la empresa de aquellos productos y valores que suelen ser solicitados por clientes.	Formulario que contiene los productos que se venden y sus precios.
T(15)	Rut del cliente	Se realiza la búsqueda de los trabajos realizados a clientes específicos y las fechas en que se solicitaron.	Formulario que contiene los trabajos realizados y las fechas de éstos.
T(16)	Situación pedido	Se busca la situación en que se encuentran los pedidos realizados a proveedores.	Formulario que muestra la situación de pedidos a proveedores.
T(17)	Origen ventas	Se revisan las facturas por concepto de ventas directas.	Formulario que muestra las facturas por concepto de ventas directas.
T(18)	Origen trabajos	Se revisan las facturas por concepto de trabajos realizados a clientes	Formulario que muestra las facturas por concepto de trabajos.
T(19)		Se realizan búsquedas de la situación de los trabajos.	Formulario que muestra la situación de trabajos.

Figura Nº 10. Mapa de Transacciones

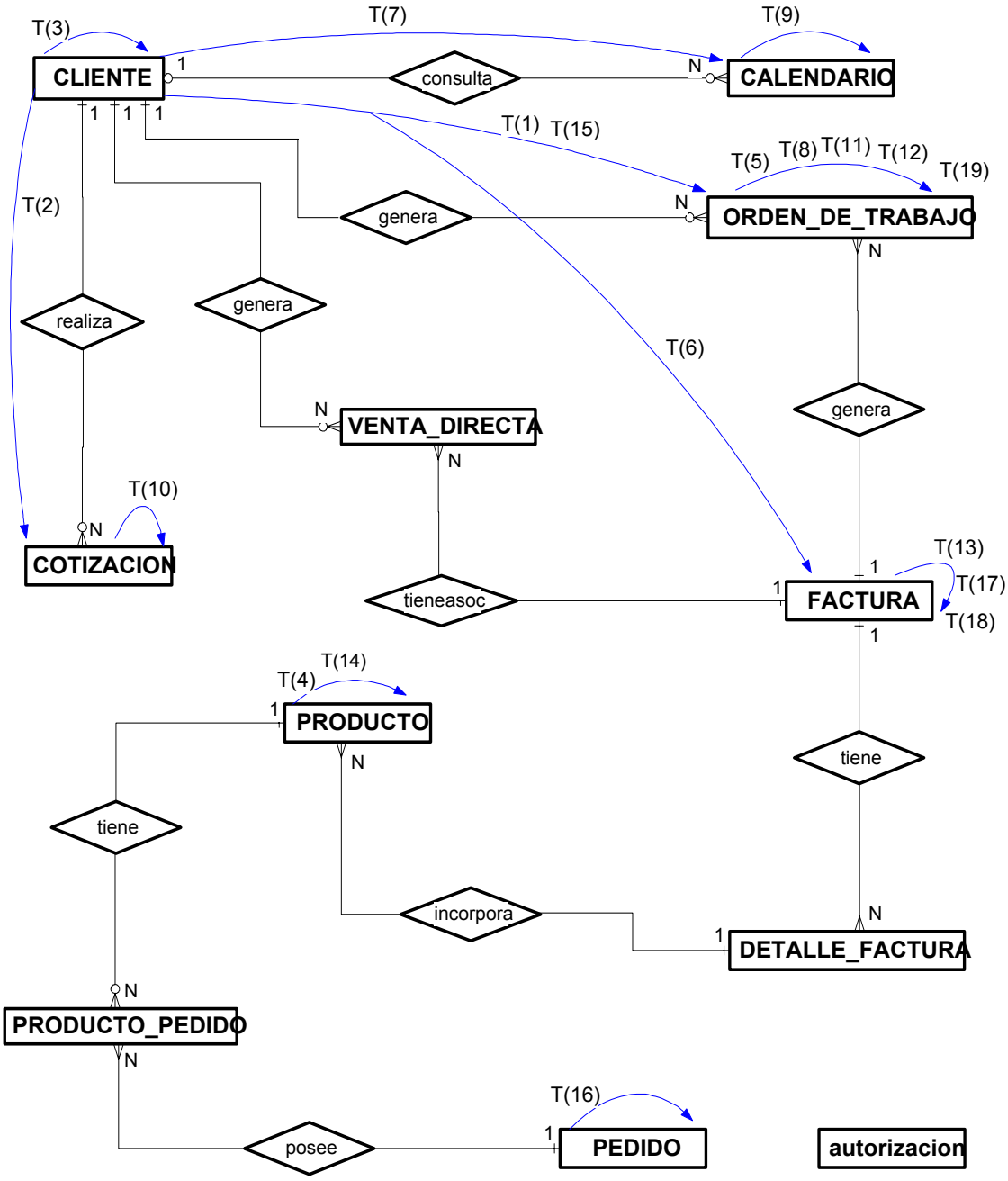
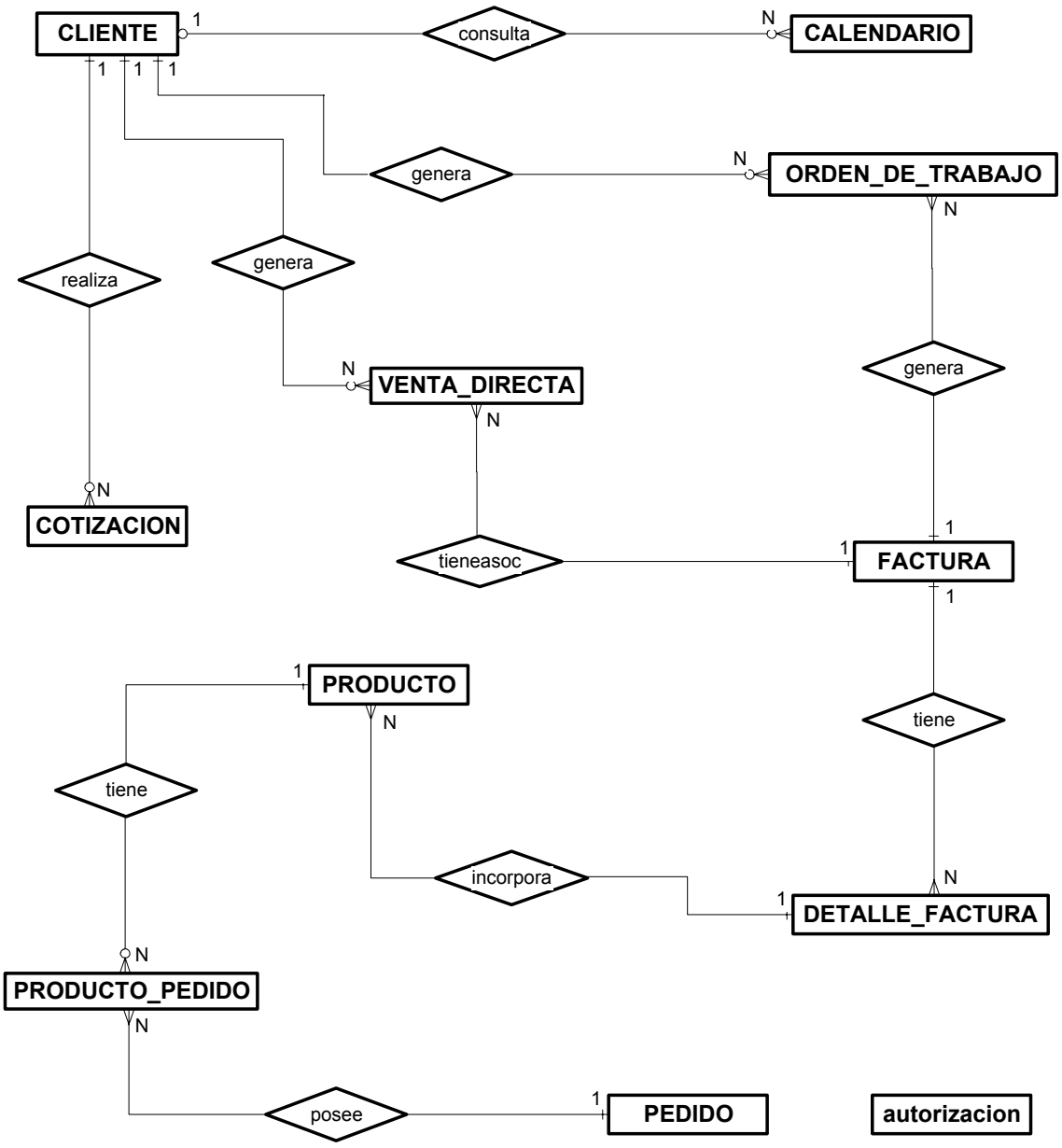


Figura Nº 11. Construcción del E-R final local.



9.2.1.7 Integridad y consistencia.

La integridad se refiere a la precisión y consistencia de los datos. En pocas palabras, la integridad se utiliza para proteger la base de datos contra la inconsistencia.

En cuanto a la integridad y consistencia se pueden enumerar algunos puntos a seguir para garantizarlas:

- Las claves primarias o partes de claves primarias no deben aceptar valores nulos (integridad). Este punto se conoce como integridad de entidades.
- Restricción de dominio de atributo. Esto consiste en que cada atributo tiene un dominio que forman un conjunto de valores legales.
- Toda ocurrencia de una clave foránea en una tabla, debe estar ligada a la existencia de ésta como clave primaria de otra tabla (consistencia). Esta parte también se conoce como integridad referencial, que significa que si una clave foránea tiene un valor, ese debe referirse a una instancia específica en la entidad padre.

En este punto de la integridad referencial se debe asegurar ésta por medio de restricción de existencia, en las que se analizan distintos casos, los que se nombran a continuación:

- Ocurrencia de inserción en la relación hijo.
- Ocurrencia de eliminación de la relación hijo.
- Actualizar clave foránea en ocurrencia de hijo.
- Ocurrencia de eliminación de la relación padre.
- Actualizar clave primaria de ocurrencia de padre.

Para concretar esto (inserción, eliminación, actualización) existen varias estrategias:

1. No action.
2. Cascade.
3. Set null.
4. Set default.
5. No check.

Las estrategias mostradas en la tabla Nº 10, se refieren al caso cuando se elimina una relación padre.

Tabla Nº 10. Estrategias para mantener la integridad de la base de datos.

Estrategias	Resultado
No Action	Prevenir que se elimine la relación padre si existe un hijo.
Cascade	Cuando se elimina el padre, automáticamente se elimina cualquier referencia del hijo.
Set Null	Cuando se elimina una instancia del padre, la clave foránea en el hijo es automáticamente actualizado con un null.
Set Default	Cuando se elimina una instancia del padre, las claves foráneas de todos los hijos referenciados son puestos con su valor defecto. Esto es si los atributos que componen las claves foráneas tienen valores defecto.
No Check	Esta estrategia dice que se elimina una instancia padre “no haga nada” para asegurar que la integridad se mantenga.

En la tabla Nº 11 que se muestra a continuación, se procede a identificar las entidades padres y las entidades hijos, con el objeto de ver lo que sucede al momento de actualizar una entidad padre y que ocurre al eliminar una de estas entidades.

Tabla Nº 11. Situación al actualizar o eliminar en una entidad padre.

Entidad padre	Entidad hijo	On Delete	On Update
CLIENTE	CALENDARIO, COTIZACION, FACTURA, ORDEN_DE_TRABAJO, VENTA_DIRECTA	Cascade Cascade Cascade Cascade Cascade	Restrict Restrict Restrict Restrict Restrict
FACTURA	ORDEN_DE_TRABAJO, DETALLE_FACTURA, VENTA_DIRECTA	Cascade Cascade Cascade	Restrict Restrict Restrict
PRODUCTO	DETALLE_PRODUCTO, PRODUCTO_PEDIDO.	Cascade Cascade	Restrict Restrict
PEDIDO	PRODUCTO_PEDIDO,	Cascade	Restrict

9.2.2 Integración de la base de datos.

9.2.2.1 Integración del modelo de datos lógico local al modelo global.

Este procedimiento se efectúa debido a que existe otro módulo web que está orientado al servicio técnico Procom, realizado por el alumno Julio A. Reinoso Villarroel, y que forma parte de un seminario de titulación. Debido a esto, el objetivo es juntar ambos modelos de datos lógico local en uno solo, lo que se traduce en un modelo de datos lógico global.

Para llevar a cabo esto se procede a lo siguiente:

- Revisar nombres de entidades y sus claves primarias. Con esto se evita que dos entidades tengan el mismo nombre, o que éstas sean las mismas entidades con nombres distintos.
- Revisar nombres de relaciones, con el objeto de evitar relaciones iguales con nombres distintos entre otras cosas.
- Verificar claves foráneas, con el propósito de verificar que estas todavía sean válidas en las entidades hijos.
- Verificar las restricciones de integridad. Esto es con el fin de evitar conflictos y asegurar la consistencia de los datos.

9.2.2.1.1 Revisar nombres de entidades y sus claves primarias.

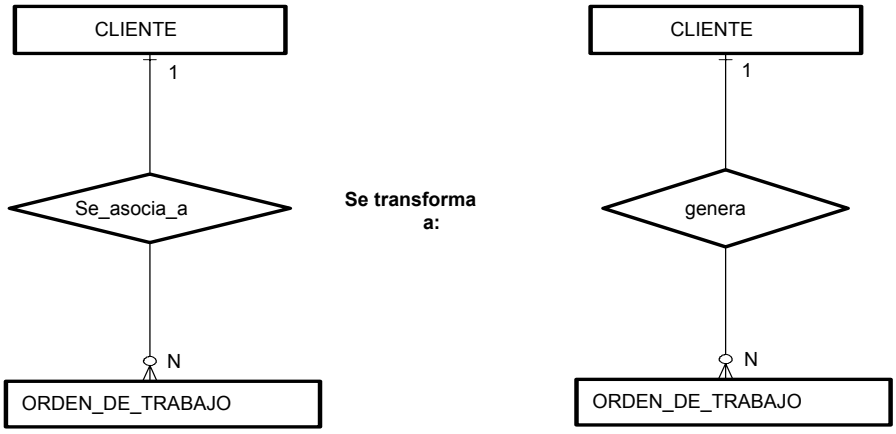
En este paso se comienza a revisar los nombres de entidades y las claves primarias en ambos modelos E-R. Una vez que se verifican ambos modelos, se encontraron algunas entidades similares con nombres distintos. Estas entidades a las que se hace referencia, son la entidades Cliente y Orden_de_trabajo.

Con respecto a las claves primarias no hubo problemas, pues en aquellas tablas o entidades iguales, las claves primarias eran las mismas, esto es en cuanto a escritura y tipo de datos.

9.2.2.1.2 Revisar nombres de relaciones.

El objetivo de este punto es verificar si existen relaciones iguales con nombres distintos. En el proceso de revisión se encontró lo siguiente, figura Nº 12:

Figura Nº 12. **Cambio nombre de Relación.**



Esta relación se deja en forma definitiva con el nombre de **genera** en el modelo E-R global.

9.2.2.1.3 Verificar claves foráneas.

En el proceso de verificar claves foráneas se constató que la totalidad de ellas estaban escritas correctamente, esto quiere decir, sin variación en cuanto al tipo de datos. Por este motivo, en este proceso no hubo cambios al momento de integrar ambos modelos E-R.

9.2.2.1.4 Verificar las restricciones de integridad.

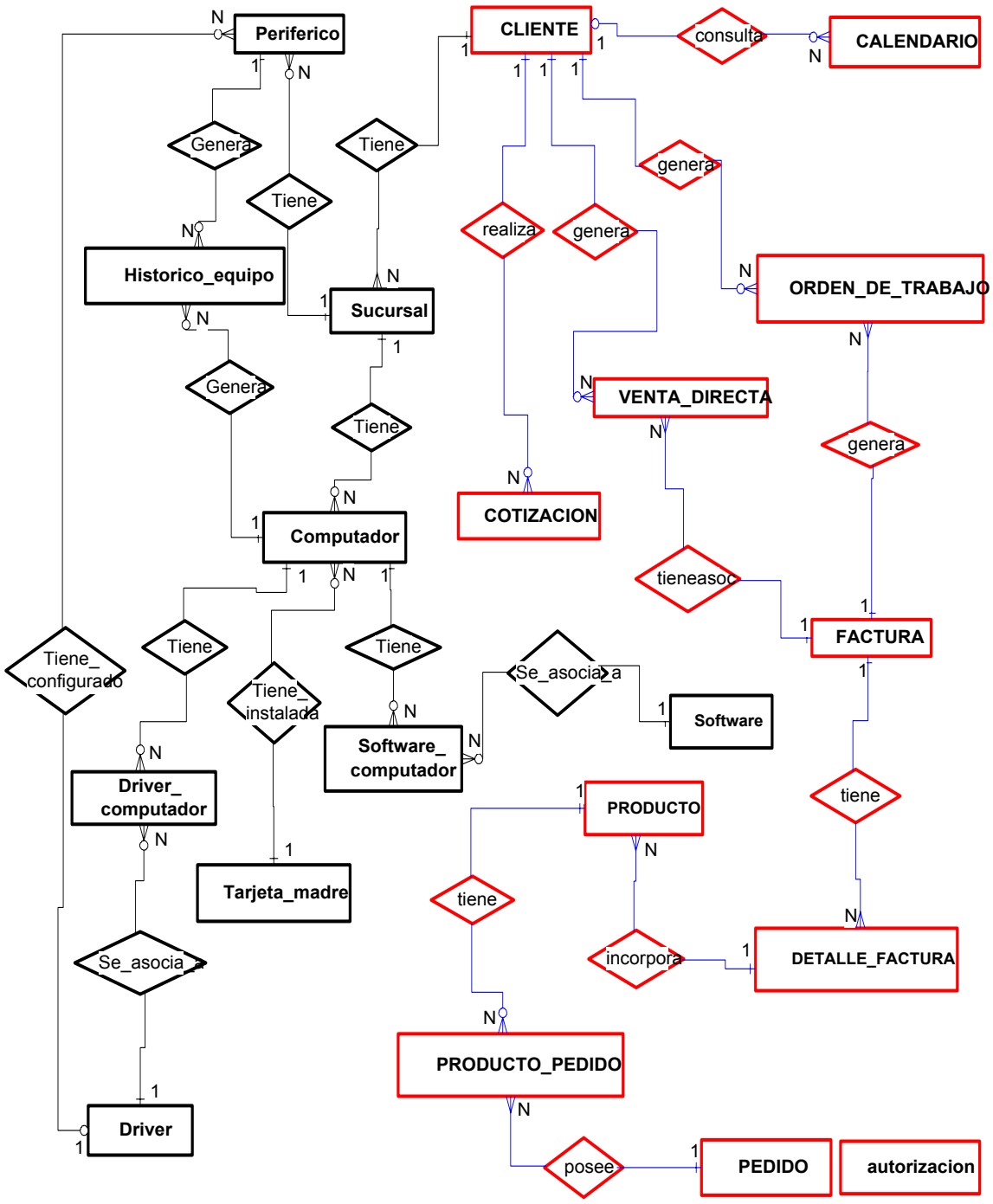
En ambos modelos se revisan las restricciones de integridad. Esta revisión comprende:

1. Restricciones de dominios de atributos: El objeto es evitar que atributos iguales tengan dominios diferentes. En este proceso no se encontraron anomalías.
2. Integridad de entidades: Aquí se verifica que cada clave primaria cumpla con la condición de no ser nula.
3. Datos requeridos: El propósito es revisar datos null o not null, y así asegurar que estos sean iguales en aquellos atributos de entidades concordantes.

9.2.2.2 Modelo E-R global.

A continuación, en la figura Nº 13, se juntan los dos modelos de datos lógico local, para dar vida al modelo de datos lógico global.

Figura Nº13. Modelo E-R lógico global.



9.3 Diseño Físico de la base de datos:

Esta parte del diseño consiste en traducir el modelo de datos lógico local para el DBMS especificado. Específicamente, el diseño físico es el tercer paso del proceso de diseño de base de datos, durante el cuál se debe decidir como la base de datos debe ser implementada, por lo que el diseño físico debe ajustarse a las especificaciones del DBMS.

En general, la meta que se persigue en el diseño físico es:

- Proceder a traducir el modelo lógico local al DBMS.
- Diseñar relaciones base para el DBMS especificado.
- Diseño de representación física.
- Políticas de seguridad para los datos.

9.3.1 Traducción del modelo de datos lógico local para el DBMS especificado.

Lo que se persigue en este paso es traspasar e interpretar de la mejor forma posible lo realizado en el diseño lógico, con el objeto de llegar a concretar esto en el DBMS, y así tener un funcionamiento óptimo del modelo de datos lógico local.

9.3.2 Diseño de relación base para un DBMS específico.

9.3.2.1 Tipos de datos e integridad.

Una base de datos está compuesta por tablas, y a la vez estas están formadas por distintos atributos. Una de las propiedades de los atributos es su tipo de datos, que define el tipo de dato que estos atributos pueden almacenar. Entre los tipos de datos más comunes que el SQL Server 7.0 poseen, se encuentran:

a) Binary.	e) Char.	i) Datetime.	m) Varchar.
b) Float.	f) Int.	j) Text.	n) Smalldatetime.
c) Bit.	g) Numeric.	k) Decimal.	ñ) Ntext.
d) Smallint.	h) Real.	l) Nchar.	o) Image.

Los tipos de datos definidos hacen que la estructura de las tablas tengan mayor significado, contribuyendo así, a asegurar que los atributos que almacenan clases de datos similares tengan el mismo tipo de datos.

También, un dominio, que es el conjunto de valores permitidos por un atributo, no sólo incluye el tipo de datos que se tienen que almacenar, si no que también se puede exigir la asignación de valores permitidos, mediante mecanismos como restricciones **check** y **desencadenadores**. Además, los atributos pueden aceptar o rechazar valores NULL, siendo NULL un valor

especial que en la base de datos representa el concepto de un valor desconocido. Así como existe un NULL, hay un NOT NULL donde se obliga a tener un valor. Existen otras condiciones como primary key, unique, default, que en conjunto ayudan a mantener la integridad de los datos.

9.3.2.2 Desencadenadores o triggers.

Es un procedimiento almacenado de tipo especial que se invoca automáticamente cada vez que se modifican los datos de una tabla. Algunos se programan y otros que vienen en la base de datos. Los desencadenadores se utilizan en respuesta de inserciones, actualizaciones o eliminaciones en tablas de la base de datos.

El objetivo de estos desencadenadores es permitir la integridad de la información y a la vez la consistencia de la base de datos. Por ejemplo, cuando se realizan transacciones que contienen errores graves, los desencadenadores deshacen automáticamente toda la transacción. Por otro lado, si se producen eliminaciones de tablas padres, estos procedimientos se activan con el objeto de que también sean eliminados los correspondientes registros en las tablas hijos, con el fin de mantener la consistencia de los datos. De esta misma forma estos procedimientos actúan en las situaciones de inserción y actualización.

9.3.2.3 Índices.

En el proceso de diseño de la base de datos se advierten dos tipos de índices, siendo éstos las llaves primarias y llaves foráneas.

Una llave primaria es uno, o conjunto de atributos que permiten valores únicos, es decir no ocurren más de una vez.

Una llave foránea es un atributo que hace referencia a una llave primaria de otra relación, concluyendo con esto que en una relación pueden haber varias llaves foráneas.

9.3.3 Diseño de representación física.

9.3.3.1 Estimación del espacio en disco para el almacenamiento de datos.

El espacio en disco que se requiere para poder almacenar los distintos datos generados tanto por clientes como por el personal de la empresa no es alto, debido a que el número de clientes estables con que generalmente trabaja esta empresa no es excesivo y a la vez el almacenamiento de datos que éstos producen sería bajo. En concreto, la mayoría de los datos que se almacenan corresponden a datos de clientes, productos que se venden, trabajos que se realizan, como también los datos de las diversas facturas por conceptos de ventas y trabajos que se efectúan. Por lo tanto, el espacio que se requiere para esta joven base de datos es el óptimo, lo que no debería producir problemas en sus inicios.

9.3.4 Políticas de seguridad para los datos.

Este paso se tratará más adelante en el capítulo Nº 13, incorporando preferentemente la seguridad en el acceso de datos y la seguridad web.

10. Diseño del sitio web.

En esta fase se encuentra el “proyecto”, cuyo propósito es tratar la información que se ha recolectado para luego trabajar con ella junto al cliente con el objeto de realizar un sitio web que se ajuste a sus necesidades, evitando así contratiempos.

El proyecto está formado por:

- Diseño lógico del sitio web.
- Diseño físico del sitio web.

10.1 Diseño lógico del sitio web.

El diseño lógico del sitio web comprende los siguientes pasos:

1. Organización de la información.
2. Diseño de los sistemas de navegación.
3. Diseño de los sistemas de rotulación.
4. Inventario de las páginas web.

10.1.1 Organización de la información.

Los sistemas de organización se componen de esquemas de organización y estructuras de organización.

10.1.1.1 Esquemas de organización.

Los esquemas de organización se preocupan de ver las características más comunes de los elementos del contenido, para luego ver de que forma se pueden agrupar estos elementos de manera lógica.

El objetivo de los esquemas de organización, es facilitar el acceso a los usuarios cuando éstos navegan.

Existen esquemas de organización como:

- Esquemas de organización exactos.
- Esquemas de organización ambiguos.
- Esquemas de organización híbridos.

10.1.1.1.1 Esquemas de organización exactos.

Estos esquemas dividen la información en secciones bien definidas y excluyentes entre sí.

La desventaja de este tipo de organización es que el usuario debe saber el nombre específico del recurso que anda buscando.

Algunos ejemplos de estos esquemas son, esquemas alfabéticos, cronológicos y geográficos.

10.1.1.1.2 Esquemas de organización ambiguos.

Estos esquemas dividen la información en categorías que se resisten a la definición exacta. Una característica de este tipo de esquema es que posibilita al usuario a encontrar de mejor forma lo que busca, debido al agrupamiento de los temas, pues los usuarios muchas veces no saben lo que andan buscando.

Algunos esquemas de organización ambiguo son:

- **Temático:** Aquí se organiza la información por materia o por temas.

- **Funcional:** Este tipo de esquema organiza el contenido y las aplicaciones en torno a un conjunto de procesos, funciones o tareas. Es apropiado para organizar tareas de alta prioridad que realizan los usuarios, pues aquí se integran aplicaciones poderosas junto con el contenido.

- **Para públicos específicos:** El enfoque de este esquema se dá en donde existen dos o más públicos. Esto nos permite dividir el sitio en minisitios específicos para cierto tipo de público. Los públicos específicos pueden ser abiertos o cerrados. En los primeros se permiten que los miembros de un tipo de público entren al contenido dirigido a otro tipo de público. El segundo evita que los miembros de cualquier público se muevan entre secciones dirigidas a otros. En este caso se usa a menudo medidas de seguridad.

10.1.1.1.3 Esquemas de organización híbridos.

Este esquema combina una serie de esquemas de organización. Por un lado esto puede ser confuso, pero si se separa cada esquema y estos se organizan, se puede tener varios esquemas sin causar confusión al usuario cuando este navega.

Cabe destacar por último que el diseño web para este sistema se centra en este tipo de esquema (esquema híbrido), debido que involucra tanto el aspecto temático, funcional como también el para públicos específicos.

10.1.1.2 Estructura de organización.

Las estructuras de la organización nos muestra la forma en que los usuarios navegarán en forma primaria a través del sitio. Las estructuras de organización más importante son:

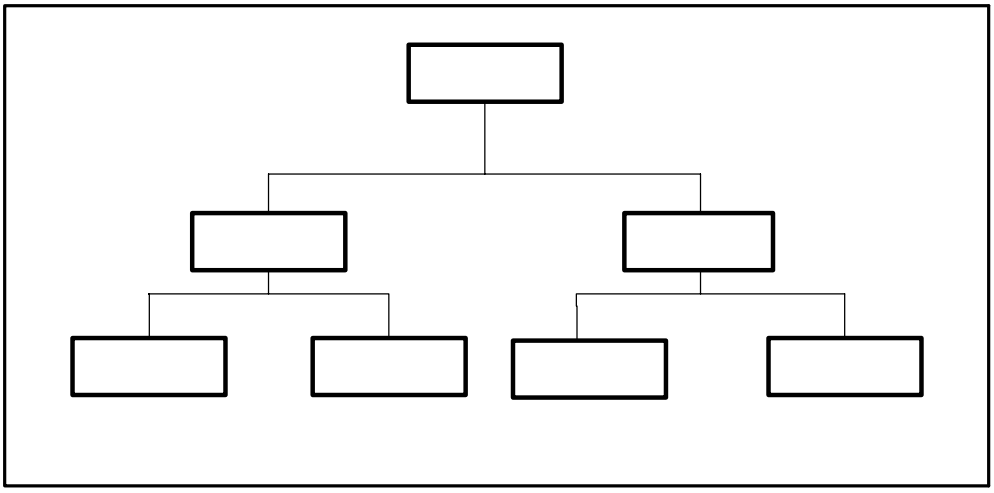
- Estructura de organización jerárquica.
- Estructura de organización de hipertexto.
- Estructura de organización orientada a base de datos.

10.1.1.2.1 Estructura de organización jerárquica.

Esta estructura permite de una forma simple y conocida organizar la información. Su enfoque de arriba abajo otorga una guía del tamaño del sitio, de esta forma se identifican áreas del contenido más relevantes que en conjunto con los esquemas de organización, darán acceso al contenido.

A continuación, en la figura Nº 14 se muestra el modelo de organización jerárquica.

Figura Nº 14. Modelo de organización jerárquica.



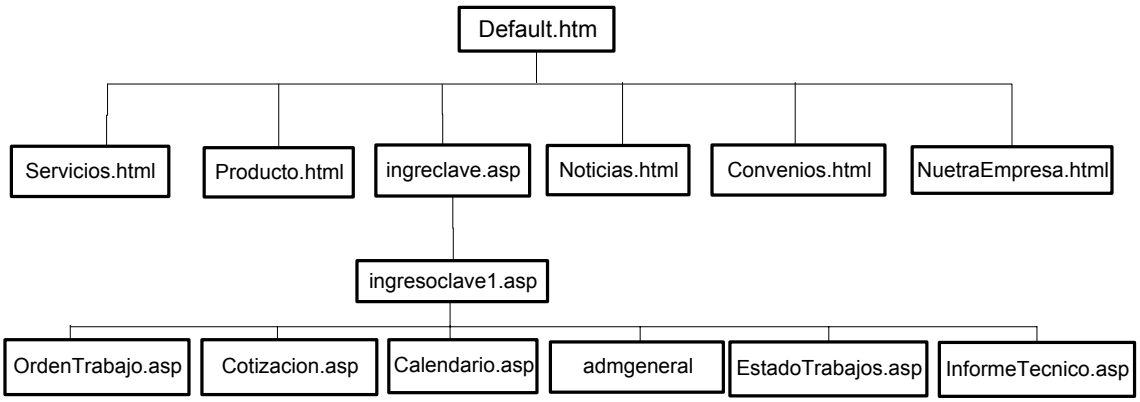
El diseño de jerarquía se caracteriza por tener 2 elementos de suma importancia, siendo estos la amplitud y profundidad.

Para que un sitio tenga una acogida aceptable por el usuario, este debe cumplir con un equilibrio entre estos 2 elementos. Esto quiere decir que el

modelo de organización no debe ser tan amplio (que no tenga tantas opciones por nivel) y que a la vez no sea tan profundo (que no tenga tantos niveles).

En la figura Nº 15 se muestran algunas de estas características correspondientes al sitio web.

Figura Nº 15. Algunos pasos de una estructura jerárquica aplicada al sitio web.

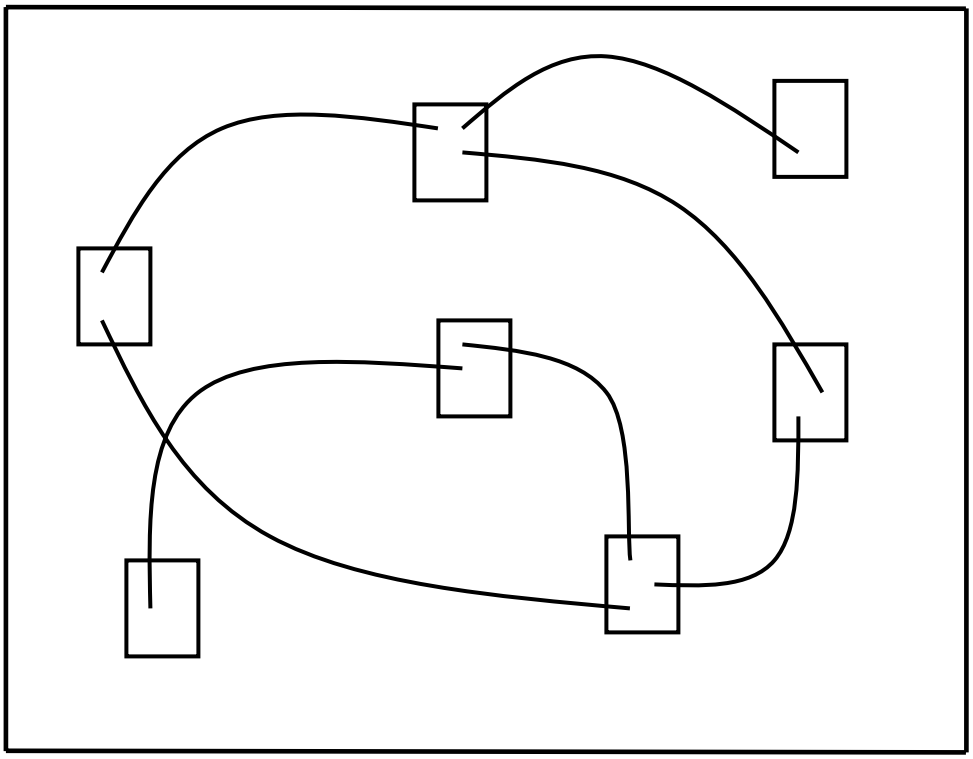


10.1.1.2.2 Estructura de organización de hipertexto.

Es una forma de estructurar la información. El hipertexto se puede utilizar para complementar estructuras jerárquicas o de base de datos. No obstante, también permite establecer relaciones útiles entre elementos y áreas de la estructura jerárquica.

En la figura Nº 16, se muestra un ejemplo breve de una estructura de organización hipertexto.

Figura Nº 16. Estructura de organización hipertexto.



10.1.1.2.3 Estructura de organización orientada a base de datos.

Este tipo de organización tiene un enfoque de abajo hacia arriba. Esta estructura permite utilizar un mismo contenido de varias maneras y formatos para distintos usuarios. Una desventaja de esta estructura es que puede llegar a ser muy cara y a la vez requerir demasiado tiempo.

10.1.2 Diseño de sistemas de navegación.

Los sistemas de navegación comprenden varios elementos como barras de desplazamiento, menús, tablas de contenido, mapas del sitio entre otros. El objetivo de los sistemas de navegación es evitar que los usuarios se extravíen.

Algunos tipos de sistemas de navegación son:

10.1.2.1 Sistemas de navegación jerárquicos.

Este es el sistema de navegación principal, donde se va desde la página principal a las páginas de destino.

10.1.2.2 Sistemas de navegación global.

El sistema de navegación global complementa un sistema de navegación jerárquico, permitiendo mayor capacidad de desplazamiento vertical y lateral en todo el sitio.

Un ejemplo sencillo de un sistema de navegación global es colocar una barra de navegación gráfica en cada página del sitio, de esta forma se mantienen las opciones principales en el sitio web.

Un ejemplo [Figura Nº 17] de esto aplicado al diseño web del sistema en cuestión sería:

Figura Nº 17. sistema de navegación global.

10.1.3 Diseño de los sistemas de rotulación.

Los sistemas de rotulación son una forma de representación. Se usan para representar partes de información en los sitios web.

Los tipos de sistemas de rotulación habitualmente se presentan en forma textual o por íconos, estando estos como índices o rótulos para vínculos y también como encabezados que dividen o identifican partes de información en una misma página.

Algunos sistemas de rotulación son:

10.1.3.1 Rótulos dentro de sistemas de navegación.

Estos rótulos son aquellos que se mantienen dentro del sitio web, con el fin de mantener aquellos rótulos de la página principal en las demás páginas del sitio, con el objeto de que el usuario este familiarizado y sepa el significado de cada uno de estos.

En la tabla Nº 12, se muestra los rótulos de la página principal del sitio y su respectivo significado.

Tabla Nº 12. Rótulos de la página principal del sitio.

Rótulo	Significado
Nuestra Empresa	Características de la empresa Procom y parte de su historia.
Servicios	Muestra los distintos servicios que la empresa ofrece a los clientes.
Productos	A través de este rótulo se puede ver los tipos de productos con que la empresa trabaja y característica de estos.
Noticias	Se da a conocer distintos tipos de noticias referentes a la empresa y desde luego noticias referente a informática.
Convenios	Por medio de este rótulo se da a conocer los tipos de convenios que un cliente puede acoger con la empresa.
Acceso Restringido	Acceso de clientes y personal de la empresa Procom, por medio de claves secretas, a las distintas operaciones que estos puedan realizar.

10.1.3.2 Rótulos como términos indizados.

Este tipo de rótulos tienen como función mejorar la posibilidad de que un documento sea hallado mediante un sistema de búsqueda o también apoyar la exploración de un sitio web.

10.1.3.3 Rótulos de vínculos.

Estos rótulos se emplean como vínculos de texto dentro del cuerpo o del texto de una parte de la información.

Un ejemplo de este tipo de rótulo en el sitio web es el siguiente, figura Nº 18:

Figura Nº 18. Rótulos de vínculos.

10.1.3.4 Rótulos como encabezados.

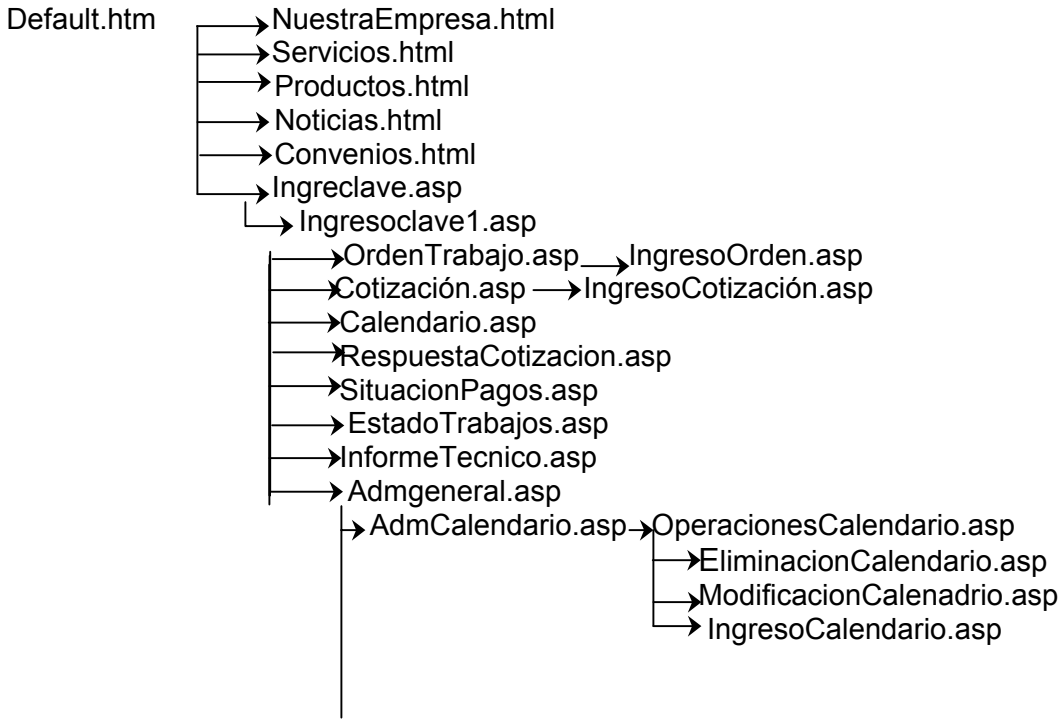
El objetivo de estos rótulos es atraer al usuario hacia la parte de la información que les corresponda o anden buscando.

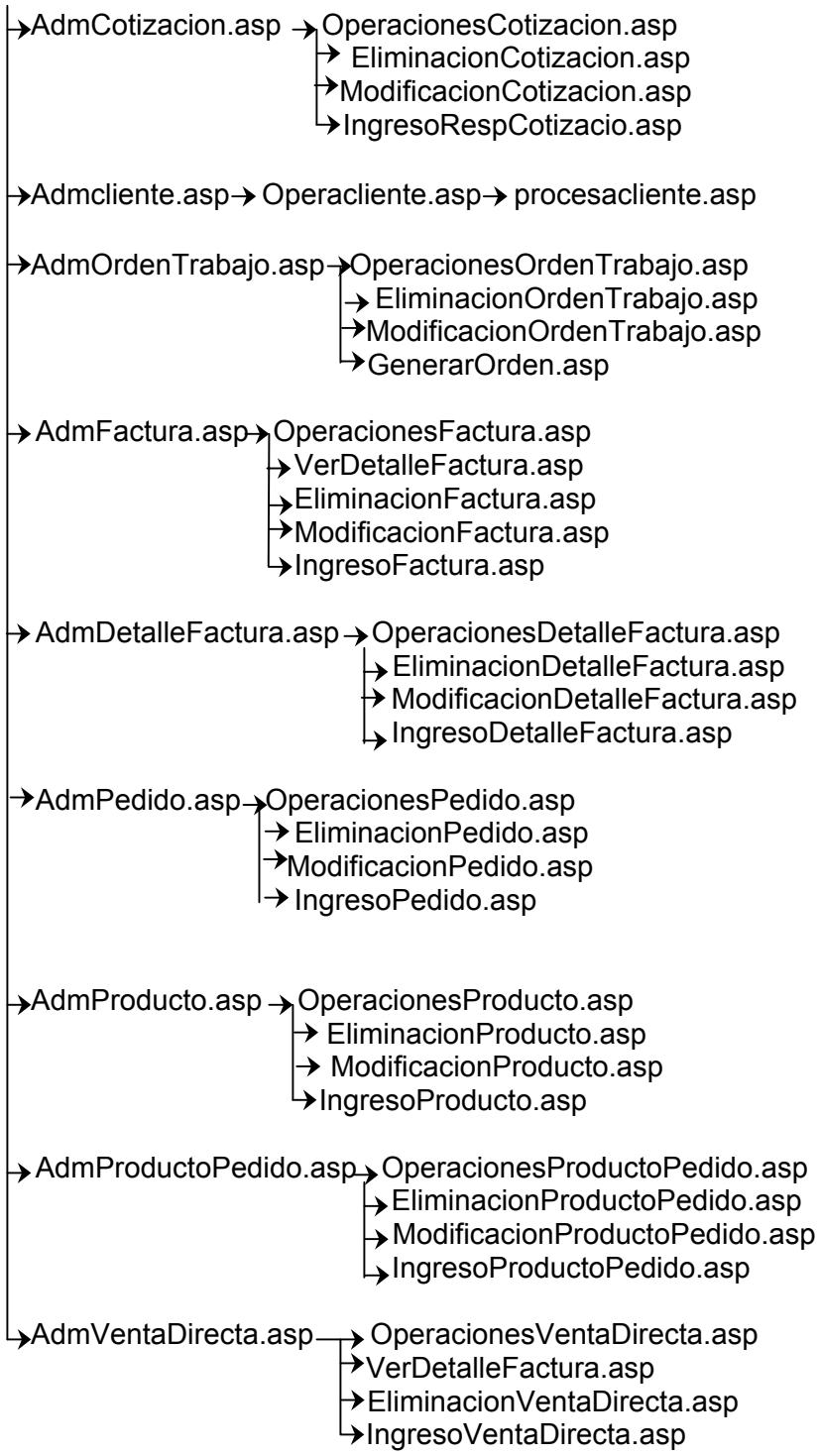
Ejemplo de estos rótulos son, figura Nº 19:

Figura Nº 19. Rótulos de encabezados

10.1.4 Inventario de las páginas web.

En este punto se dan a conocer las distintas páginas con sus respectivos nombres.




AdmCotizacion.asp
OperacionesCotizacion.asp
EliminacionCotizacion.asp
ModificacionCotizacion.asp
IngresoRespCotizacio.asp
Admcliente.asp
Operacliente.asp
procesacliente.asp
AdmOrdenTrabajo.asp
OperacionesOrdenTrabajo.asp
EliminacionOrdenTrabajo.asp
ModificacionOrdenTrabajo.asp
GenerarOrden.asp
AdmFactura.asp
OperacionesFactura.asp
VerDetalleFactura.asp
EliminacionFactura.asp
ModificacionFactura.asp
IngresoFactura.asp
AdmDetalleFactura.asp
OperacionesDetalleFactura.asp
EliminacionDetalleFactura.asp
ModificacionDetalleFactura.asp
IngresoDetalleFactura.asp
AdmPedido.asp
OperacionesPedido.asp
EliminacionPedido.asp
ModificacionPedido.asp
IngresoPedido.asp
AdmProducto.asp
OperacionesProducto.asp
EliminacionProducto.asp
ModificacionProducto.asp
IngresoProducto.asp
AdmProductoPedido.asp
OperacionesProductoPedido.asp
EliminacionProductoPedido.asp
ModificacionProductoPedido.asp
IngresoProductoPedido.asp
AdmVentaDirecta.asp
OperacionesVentaDirecta.asp
VerDetalleFactura.asp
EliminacionVentaDirecta.asp
IngresoVentaDirecta.asp

10.2 Diseño físico del sitio web.

En el diseño físico del sitio web se incluye lo siguiente:

- Generadores de aplicación web.
- Forma en que se carga las páginas.
- Levantamiento del servidor web.
- Medida de seguridad del sitio web.

10.2.1 Generadores de aplicación web.

Un generador de aplicación web permite crear un sitio de éste tipo. Los generadores de aplicación que se usan en la construcción del sitio web son:

10.2.1.1 Microsoft Frontpage98.

Este software permite crear páginas web. Por medio del código HTML se realizan las distintas aplicaciones, se determinan los colores, link, se diseñan los sistemas de navegación, rótulos y algunos botones entre otras cosas, con el objeto de formar cada página con los contenidos que se requieren para los usuarios.

10.2.1.2 EditPlus.

Este software es un editor de texto, editor HTML y editor para programación en HTML, CSS, PHP, ASP, Perl, C/C++, Java, JavaScript y VBScript. Con él se realizan las distintas páginas con código ASP, que luego son traspasadas al Frontpage98 para complementar la parte gráfica. Esto se debe a que el Editplus es más amigable al momento de la programación, pues permite tener la enumeración de las líneas de programación que son de mucha importancia al momento de detectar errores.

10.2.1.3 Macromedia Flash 5.0.

Es un software que permite crear controles de navegación, logotipos animados, animaciones de gran formato con sonido sincronizado e incluso sitios Web con gran capacidad sensorial y animación. Con este software se realiza la portada del sitio web, siendo con esto la página principal, y a la que se dedica mayor trabajo gráfico, pues es la carta de presentación de la empresa. Esta herramienta permite realizar un excelente diseño y a la vez crear distintas imágenes animadas con el objeto de captar la atención del usuario y de que este permanezca un mayor rato navegando en el sitio.

Cabe destacar que la portada del sitio es la única que está realizada con flash 5.0 y que el resto de las páginas tienen algunos gif animados creados con flash. El objeto de hacer la pura portada en flash y no otras páginas, se debe en

gran medida para no provocar una demora en el momento en que se cargan las páginas, evitando con esto que el usuario se aburra al encontrar un sitio que se torne lento al navegar.

10.2.2 Forma en que se cargan las páginas.

Las páginas web son cargadas en el servidor web, en este caso Internet Information Server (I.I.S.). Estas páginas se tratan de páginas ASP, que son ejecutadas en el servidor cuando un usuario conectado a Internet las pide; de esta manera, se pueden ejecutar dentro de ellas sentencias o scripts para devolver un código HTML a medida. El funcionamiento no es difícil. El navegador envía una pregunta (Request o Petición) al servidor, este llama al ASP, ejecuta los comandos script y envía la respuesta (Response) de vuelta al navegador en formato HTML.

La ruta de los archivos en que se almacenan las páginas, es la siguiente:

C:\Inetpub\wwwroot

Por otro lado, la ruta de todas las imágenes que se incorporan a este sistema es:

C:\Inetpub\wwwroot\Images

De esta forma, las páginas web que conforman el sistema quedan operativas.

10.2.3 Levantamiento del servidor web.

El servidor web que se utiliza es el Microsoft Internet Information Server (I.I.S). El I.I.S, servidor de Internet Microsoft es de alto rendimiento, seguro, basado en Windows NT Server. Es a la vez compatible con World Wide Web, Ftp y Gopher.

La instalación del I.I.S se debe efectuar en un equipo con sistema operativo Windows NT Server. Durante la instalación se seleccionan las opciones del I.I.S que se desean instalar como por ejemplo:

- Administrador del servicio Internet.
- Servicio World Wide Web.
- Ejemplos de servicios WWW.
- Servicios Gopher.
- Servicios FTP.
- Administración y controladores ODBC.

Luego, se configuran las carpetas de directorio para los servicios de publicación. Ejemplo:

Directorio de publicación de World Wide Web.

C:\Inetpub\wwwroot

Directorio de publicación FTP.

C:\Inetpub\ftproot

Directorio de publicación Gopher.

C:\Inetpub\gophroot

Por último, se selecciona los controladores ODBC que utilizará el I.I.S para la conexión con la base de datos a ocupar, en este caso SQL Server.

10.2.4 Medida seguridad sitio web.

La seguridad del sitio web en conjunto con la base de datos, se tratan más adelante en la sección seguridad, capitulo 13, sección 13.3.

11. Selección del DBMS.

En la selección del DBMS se debe elegir un motor de base de datos adecuado a las características y requerimientos del diseño. Esta selección se realiza entre el diseño conceptual y el diseño lógico de la base de datos.

El motor de base de datos elegido es el Microsoft SQL Server 7.0. Esta elección se debe a que el SQL Server es un motor de gran capacidad, robustez y sofisticación. A la vez, se complementa de buena forma con los sistemas orientados a web, dando rapidez y seguridad. Por último, con este motor de base de datos se pueden realizar distintas operaciones como son los triggers, procedimientos almacenados, integridad referencial entre otros, cumpliendo así las con las características deseadas al momento de la elección.

12. Prototipo.

En esta etapa se realizaron algunos prototipos con el fin de probar algunas páginas en ASP y constatar su funcionamiento, con el propósito de no tener inconvenientes en el momento de la implementación final del sistema.

Otro aspecto que se cubre al realizar prototipos, es el de confirmar los requerimientos recolectados y a la vez para ver si se incorporan otros con el fin de cubrir mayoritariamente las necesidades de la empresa.

13. Seguridad.

A continuación, se describen los distintos aspectos que se deben tomar en cuenta para que el sistema sea seguro, tanto en la base de datos, como en la web.

13.1 Acceso de datos.

La existencia de un control de acceso a los datos, permite garantizar la seguridad de éstos. El uso de una base de datos centralizada contribuye en gran medida al logro de este objetivo. Sin embargo, al tener un mecanismo de acceso y permisos, otorga un control sobre la manipulación de los datos por parte de los usuarios y de las operaciones que éstos puedan llegar a realizar.

El mecanismo de acceso que se emplea para evitar la intromisión de usuarios no deseados, consiste en tener para cada usuario una password, por lo que un cliente determinado podrá sólo ingresar dando a conocer su Rut y su respectiva contraseña, una vez dentro existen niveles de seguridad, donde se restringe para ciertos usuarios el acceso a determinados datos. Estos niveles de seguridad se crean en la tabla autorización, donde existen cuatro niveles, los cuáles serán otorgados al personal de la empresa según los trabajos que realicen éstos. Una de las personas con máximo nivel de autorización es el administrador de la base de datos, el cuál tiene nivel 1.

Un aspecto importante a mencionar, es el hecho que estos niveles se controlan a través del código asp. Primero se obtiene el nivel de seguridad del individuo que ingresa a través de la página ingreclave.asp, luego por medio de una variable Session, la que permite mantener el valor del nivel a través de varias páginas, se procede a controlar el acceso a la administración de las distintas tablas. El hecho de usar variables Session para este efecto permite obtener una medida de seguridad extra, pues si el usuario no realiza operaciones dentro de 20 minutos, el sistema envía al usuario a la página ingreclave.asp para que vuelva a ingresar. Cabe destacar que la creación de nuevos usuarios con distintos niveles de seguridad se pudo realizar por medio de objetos a través del SQL Server, pero se considera que la manipulación en forma directa de la base de datos por el administrador, puede generar riesgos innecesarios a los datos, por lo que crear usuarios con sus niveles de seguridad a través del sistema es más seguro.

Por último, para mantener en forma más específica la seguridad de los datos, se deben considerar las siguientes políticas de seguridad:

- Otorgamientos de cuentas, descrito anteriormente.
- Habilitación de usuarios.
- Actualización de software crítico. Ejemplo Antivirus.

- Instalación de software localmente.
- Respaldo y restauración de información.
- Determinar responsabilidades de usuarios.
- Consecuencias para los usuarios que divulgan información, la que podría ser propietaria.
- Administrar contraseñas seguras, o sea, que no sean fáciles de descifrar, para evitar así, que la cuenta y el sistema no sean vulnerables.

De esta manera se controla tanto el acceso de los usuarios a los datos, como las acciones que se llevan a cabo sobre ellos.

13.2 Seguridad de Microsoft Windows NT Server 4.0.

Windows NT Server 4.0 utiliza un modelo de seguridad fuerte dando a usuarios contraseñas para controlar los accesos al servidor. Esto es indispensable, pues en el servidor se concentra la base de datos y las aplicaciones web. Por lo tanto, para cualquier tipo de modificación del sistema se debe pasar primero por las medidas de seguridad que impone Windows NT. Se hace notar que las modificaciones se realizan a nivel local y no remoto.

13.3 Seguridad web.

13.3.1 Firewalls.

Un Firewalls en Internet es un sistema o grupo de sistemas que impone una política de seguridad entre la organización de red privada y el Internet. Para que un Firewall sea efectivo, todo tráfico de información a través de Internet debe pasar a través del mismo, donde podrá ser inspeccionada la información. El Firewall podrá únicamente autorizar el paso del tráfico, y el mismo podrá ser inmune a la penetración.

13.3.1.1 Beneficios de un Firewall.

- Administra los accesos posibles de Internet a la red privada.
- Ofrece un punto donde la seguridad puede ser monitoreada, y si aparece alguna actividad sospechosa, este genera una alarma ante la posibilidad de que ocurre un ataque o suceda algún problema en el tránsito de los datos.
- Concentra la seguridad.
- Centraliza los accesos.
- Genera alarmas de seguridad. Traduce direcciones NAT.
- Monitorea y registra el uso de servicios de WWW y FTP.

Así como el Firewall tiene beneficios, también tiene sus limitaciones, las cuales son:

13.3.1.2 Limitaciones de un Firewall.

- Un Firewall no puede proteger contra aquellos ataques que se efectúen fuera de su punto de operación.
- No puede protegerse contra los ataques posibles de la red interna, por ejemplo por virus informativos a través de archivos o software.
- No puede proteger contra los ataques de la ingeniería social, ejemplo hacker que pretendan ser supervisor.

El firewall utilizado para este propósito es el ZoneAlarm, el cual cumple con el o los propósito(s) ya descritos anteriormente.

14. Implementación.

14.1 Implementación de la base de datos.

En la implementación de la base de datos se crean las tablas, procedimientos almacenados y los triggers o desencadenadores.

14.1.1 Creación de tablas de la base de datos.

```
CREATE TABLE  CLIENTE(
rut_cliente          numeric(8)    NOT NULL Primary Key,
dig_cliente          char(1)       NOT NULL,
nombre_cliente       char(30)      NOT NULL,
direccion_cliente    char(30)      NOT NULL,
ciudad_cliente       char(20)      NOT NULL,
contacto_cliente     char(30)      NOT NULL,
telefono_cliente     numeric(10)   NOT NULL,
email_cliente        char(30)      NULL,
password_cliente     char(10)      NOT NULL
)
```

```
CREATE TABLE CALENDARIO(
correl          numeric(8)    NOT NULL Primary Key,
fecha           datetime(8)   NOT NULL,
tipo_trabajo    char(35)      NOT NULL,
equipo          char(35)      NOT NULL,
rut_cliente     numeric(8)    Foreign Key  References CLIENTE
)

CREATE TABLE COTIZACION(
numero_doc            numeric(10)   NOT NULL Primary Key,
fecha                 datetime(8)   NOT NULL,
detalle_cotizacion    text(16)      NOT NULL,
estado_cotizacion     char(9)       NOT NULL  Default  'Pendiente' ,
respuesta_cotización  text(16)      NULL,
rut_cliente           numeric(8)    Foreign Key  References CLIENTE
)

CREATE TABLE FACTURA(
num_doc       numeric(10)   NOT NULL Primary Key,
origen_doc    char(13)      NOT NULL,
fecha         datetime(8)   NOT NULL,
neto          real(24)      NOT NULL,
```

```
iva              real(24)        NOT NULL,
total            int(10)         NOT NULL,
estado           char(9)         NOT NULL,
forma_pago       char(10)        NULL,
rut_cliente      numeric(8)      Foreign Key References CLIENTE
)

CREATE TABLE DETALLE_FACTURA(
detalle_producto   char(110)        NOT NULL,
cantidad           numeric(4)       NOT NULL,
valor_procom_ci    int (10)         NOT NULL,
especificacion     text(16)         NULL,
num_doc            numeric(10) Foreign Key  References FACTURA,
cod_prod           numeric(10) Foreign Key  References PRODUCTO,
Primary Key (num_doc, cod_prod)
)

CREATE TABLE ORDEN_DE_TRABAJO(
num_orden          numeric(6)      NOT NULL Primary Key,
fecha_emision      datetime(8)     NOT NULL,
tipo_trabajo       char(30)        NOT NULL,
detalle_orden      text(16)        NOT NULL,
```

```
solicitado_por          char(30)        NOT NULL,
nombre_técnico          char(30)        NULL,
trabajo_realizado       text(16)        NULL,
fecha_trabajoi          datetime(8)     NULL,
fecha_trabajot          datetime(8)     NULL,
aprobado_por            char( 30)       NULL,
validez_orden           char(1)         NOT NULL  Default  '0',
situacion_trabajo       char(9)         NOT NULL  Default  'Pendiente',
situación_pago          char(9)         NOT NULL  Default  'Pendiente' ,
rut_cliente             numeric(8)      Foreign Key  References CLIENTE,
num_doc                 numeric(10)     NOT  NULL  Default  '0'  Foreign  Key
References FACTURA
)

CREATE TABLE PRODUCTO(
cod_prod                numeric(10)  NOT NULL Primary Key,
tipo_producto           char(30)     NOT NULL,
detalle_producto        char(110)    NOT NULL,
valor_original_si       real(24)     NOT NULL,
valor_original_ci       int(10)      NOT NULL,
valor_procom_ci         int(10)      NOT NULL,
)
```

```
CREATE TABLE PRODUCTO_PEDIDO(
cantidad                numeric(4)   NOT NULL,
num_pedido              numeric(10) Foreign Key  References PEDIDO,
cod_prod                numeric(10) Foreign key  References  PRODUCTO,
Primary Key  (num_pedido, cod_prod)
)

CREATE TABLE PEDIDO(
num_pedido              numeric(10)   NOT NULL   Primary Key,
situacion_pedido        char(9)       NOT NULL  Default  'Pendiente',
fecha_pedido            datetime(8)   NOT NULL,
proveedor               char(30)      NOT NULL
)

CREATE TABLE VENTA_DIRECTA(
num_venta               numeric(6)     NOT NULL  Primary Key,
rut_cliente             numeric(8)     Foreign Key  References CLIENTE,
num_doc                 numeric(10)    Foreign Key  References FACTURA
)

CREATE TABLE autorización(
usuario                 char(10)      NOT NULL   Primary Key
password                char(10)      NOT NULL,
```

```
nivel                    numeric(1)     NOT NULL
)
```

14.1.2 Creación de procedimientos almacenados.

Uno de los objetivos de crear procedimientos almacenados es aumentar el rendimiento del sistema, como también aislar a los usuarios de los detalles de las tablas de la base de datos, evitando en gran medida que éstos tengan acceso directo a la base de datos.

A continuación, se dan a conocer algunos procedimientos almacenados:

Despliegues:

```
CREATE PROCEDURE Obt_Cliente
@rut numeric(8)
AS
SELECT
rut_cliente,nombre_cliente,direccion_cliente,ciudad_cliente,contacto_cliente,tele
fono_cliente,email_cliente,password_cliente
FROM CLIENTE WHERE rut_cliente=@rut


CREATE PROCEDURE Obt_ClienteTd
@rut numeric(8)
AS
SELECT * FROM CLIENTE WHERE rut_cliente=@rut


CREATE PROCEDURE Obt_Autori
@login char(13),
@pass char(10)
AS
SELECT * FROM autorizacion
WHERE login=@login AND password=@pass
```

```
CREATE PROCEDURE Obt_Calendario
 @p numeric(8)
AS
SELECT correl,fecha,tipo_trabajo,equipo
FROM CALENDARIO
WHERE rut_cliente=@p ORDER BY fecha


CREATE PROCEDURE Obt_CalendarioCli
@p numeric(8)
AS
SELECT fecha,tipo_trabajo,equipo
FROM CALENDARIO
WHERE rut_cliente=@p ORDER BY fecha


CREATE PROCEDURE Obt_Cotizacion
@p numeric(8)
AS
SELECT
numero_doc,fecha,detalle_cotizacion,estado_cotizacion,respuesta_cotizacion
 FROM COTIZACION WHERE rut_cliente=@p


CREATE PROCEDURE Obt_CotizacionCli
@p numeric(8)
AS
SELECT nombre_cliente,telefono_cliente
FROM CLIENTE WHERE rut_cliente=@p


CREATE PROCEDURE Obt_CotizacionI
@p numeric(8),
@PD char(9)
AS
SELECT
numero_doc,fecha,detalle_cotizacion,estado_cotizacion,respuesta_cotizacion
FROM COTIZACION
 WHERE rut_cliente=@p AND estado_cotizacion=@PD
```

```
CREATE PROCEDURE Obt_DetalleFactura
@numdoctxt numeric(10)
 AS
SELECT
num_doc,cod_prod,detalle_producto,cantidad,valor_procom_ci,especificacion
 FROM DETALLE_FACTURA
WHERE num_doc=@numdoctxt


CREATE PROCEDURE Obt_EstadoTrabajoCli
@p numeric(8)
AS
SELECT num_orden,fecha_trabajot,tipo_trabajo,situacion_trabajo
FROM ORDEN_DE_TRABAJO WHERE rut_cliente=@p ORDER BY
num_orden


CREATE PROCEDURE Obt_Factura
@p numeric(8)
AS
SELECT
num_doc,origen_doc,fecha,rut_cliente,neto,iva,total,estado,forma_pago
FROM FACTURA WHERE rut_cliente=@p

CREATE PROCEDURE Obt_Factventas
@origen char(13)
AS
SELECT
nombre_cliente,num_doc,origen_doc,fecha,FACTURA.rut_cliente,neto,iva,total,
estado
FROM FACTURA,CLIENTE
WHERE origen_doc=@origen AND CLIENTE.rut_cliente=FACTURA.rut_cliente


CREATE PROCEDURE Obt_InformeTecnico
@p numeric(8)
AS
SELECT
num_orden,fecha_trabajoi,tipo_trabajo,nombre_tecnico,trabajo_realizado,fecha
_trabajot,aprobado_por
FROM ORDEN_DE_TRABAJO
WHERE rut_cliente=@p ORDER BY num_orden
```

```
CREATE PROCEDURE Obt_Nomtelcli
@p numeric(8)
AS
SELECT nombre_cliente,telefono_cliente
FROM CLIENTE WHERE rut_cliente=@p


CREATE PROCEDURE Obt_Ordentrabajo
 @p numeric(8)
AS
SELECT
num_orden,fecha_emision,tipo_trabajo,detalle_orden,solicitado_por,situacion_tr
abajo,situacion_pago,num_doc
 FROM ORDEN_DE_TRABAJO
WHERE rut_cliente=@p


CREATE PROCEDURE Obt_OrdentrabajoM
@p numeric(8)
AS
SELECT
num_orden,fecha_emision,fecha_trabajoi,tipo_trabajo,nombre_tecnico,
trabajo_realizado,fecha_trabajot,aprobado_por,validez_orden,situacion_trabajo,
situacion_pago,num_doc
FROM ORDEN_DE_TRABAJO WHERE rut_cliente=@p


CREATE PROCEDURE Obt_Pedido
@pedido char(9)
 AS
SELECT num_pedido,situacion_pedido,fecha_pedido,proveedor
FROM PEDIDO
WHERE situacion_pedido=@pedido ORDER BY num_pedido


CREATE PROCEDURE Obt_Ppendientes
@Emi char(9)
AS
SELECT
nombre_cliente,num_doc,origen_doc,fecha,FACTURA.rut_cliente,neto,iva,total,
estado
FROM FACTURA,CLIENTE
WHERE estado=@Emi AND CLIENTE.rut_cliente=FACTURA.rut_cliente
```

```
CREATE PROCEDURE Obt_Producto
@producto char(30)
AS
SELECT
cod_prod,tipo_producto,detalle_producto,valor_original_si,valor_original_ci,
valor_procom_ci
FROM PRODUCTO
WHERE tipo_producto LIKE '%@producto%'


CREATE PROCEDURE Obt_ProductoF
@codtxt numeric(10)
AS
SELECT cod_prod,detalle_producto,valor_procom_ci
 FROM PRODUCTO WHERE cod_prod=@codtxt


CREATE PROCEDURE Obt_ProductoPedido
@producto char(30)
AS
SELECT
PRODUCTO.cod_prod,tipo_producto,detalle_producto,num_pedido,cantidad,
stock
FROM PRODUCTO,PRODUCTO_PEDIDO
WHERE tipo_producto LIKE '%@producto%' AND
PRODUCTO.cod_prod=PRODUCTO_PEDIDO.cod_prod


CREATE PROCEDURE Obt_ProductoPedidoTd
AS
SELECT
PRODUCTO.cod_prod,tipo_producto,detalle_producto,num_pedido,cantidad,
stock
FROM PRODUCTO,PRODUCTO_PEDIDO
WHERE PRODUCTO.cod_prod=PRODUCTO_PEDIDO.cod_prod


CREATE PROCEDURE Obt_ProductoTd
AS
SELECT * FROM PRODUCTO
```

```
CREATE PROCEDURE Obt_RespCotizacion
@p numeric(8)
AS
SELECT numero_doc,fecha,estado_cotizacion,respuesta_cotizacion
FROM COTIZACION WHERE COTIZACION.rut_cliente=@p ORDER BY fecha


CREATE PROCEDURE Obt_rutnomcli
AS
SELECT rut_cliente,nombre_cliente FROM CLIENTE


CREATE PROCEDURE Obt_rutpasnomcli
@rut numeric(8),
@password char(10)
AS
SELECT rut_cliente,password_cliente,nombre_cliente
FROM CLIENTE
WHERE rut_cliente=@rut AND password_cliente=@password


CREATE PROCEDURE Obt_SituacionPagos
@p numeric(8),
@R char(9)
AS
SELECT num_orden,fecha_trabajot,tipo_trabajo,situacion_pago,num_doc
FROM ORDEN_DE_TRABAJO
WHERE rut_cliente=@p AND situacion_trabajo=@R ORDER BY num_orden


CREATE PROCEDURE Obt_Stock
@stoc numeric(4)
 AS
SELECT
tipo_producto,num_pedido,PRODUCTO_PEDIDO.cod_prod,stock,cantidad
FROM PRODUCTO,PRODUCTO_PEDIDO
WHERE stock=@stoc AND
PRODUCTO.cod_prod=PRODUCTO_PEDIDO.cod_prod
CREATE PROCEDURE Obt_VentaDirecta
@p numeric(8)
AS
SELECT num_venta,VENTA_DIRECTA.rut_cliente,VENTA_DIRECTA.num_doc
FROM VENTA_DIRECTA
```

```
WHERE VENTA_DIRECTA.rut_cliente=@p
```

Eliminación:

```
CREATE PROCEDURE Eli_Calendario
@codigo numeric(8)
AS
DELETE FROM CALENDARIO
WHERE correl=@codigo


CREATE PROCEDURE Eli_Cliente
@codigo numeric(8)
AS
DELETE FROM CLIENTE
WHERE rut_cliente=@codigo


CREATE PROCEDURE Eli_Cotizacion
@codigo numeric(10)
AS
DELETE FROM COTIZACION
WHERE numero_doc=@codigo


CREATE PROCEDURE Eli_Factura
@codigo numeric(10)
AS
DELETE FROM FACTURA
WHERE num_doc=@codigo


CREATE PROCEDURE Eli_Ordentrabajo
@codigo numeric(6)
AS
DELETE FROM ORDEN_DE_TRABAJO
WHERE num_orden=@codigo
```

```
CREATE PROCEDURE Eli_Pedido
@codigo numeric(10)
AS
DELETE FROM PEDIDO WHERE num_pedido=@codigo


CREATE PROCEDURE Eli_ProductoPedido
@codigo numeric(10)
 AS
DELETE FROM PRODUCTO_PEDIDO
WHERE cod_prod=@codigo


CREATE PROCEDURE Eli_VentaDirecta
@codigo numeric(6)
AS
DELETE FROM VENTA_DIRECTA
WHERE num_venta=@codigo
```

Modificación:

```
CREATE PROCEDURE Mo_Calendario
@vFecha datetime,
@vTipo char(35),
@vEquipo char(35),
@vCorrelativo numeric(8)
AS
UPDATE CALENDARIO SET
fecha=@vFecha,tipo_trabajo=@vTipo,equipo=@vEquipo
WHERE(correl=@vCorrelativo)


CREATE PROCEDURE Mo_Cliente
@nom char(30),
@dir char(30),
@ciu char(20),
@con char(30),
@tel numeric(10),
@mai char(30),
@pass char(10),
```

```
@rut numeric(8)
 AS
UPDATE CLIENTE SET
nombre_cliente=@nom,direccion_cliente=@dir,ciudad_cliente=@ciu,
contacto_cliente=@con,telefono_cliente=@tel,email_cliente=@mai,password_cl
iente=@pass
WHERE rut_cliente=@rut


CREATE PROCEDURE Mo_Cotizacion
@fecha datetime,
@estado char(9),
@respuestatxt text,
@r numeric(8),
@corr numeric(10)
 AS
UPDATE COTIZACION SET
fecha=@fecha,estado_cotizacion=@estado,respuesta_cotizacion=@respuestat
xt
WHERE rut_cliente=@r AND numero_doc=@corr


CREATE PROCEDURE Mo_Factura
@vOrigen char(13),
@vFecha datetime,
@vNeto real,
@vIva real,
@vTotal int,
@vEstado char(9),
@vFpago char(10),
@vNumdoc numeric(10)
 AS
UPDATE FACTURA SET
origen_doc=@vOrigen,fecha=@vFecha,neto=@vNeto,iva=@vIva,
total=@vTotal,estado=@vEstado,forma_pago=@vFpago
WHERE num_doc=@vNumdoc


CREATE PROCEDURE Mo_Ordentrabajo
@vName char(20),
@vTrabajo text,
@vFechati datetime,
@vFechatt datetime,
```

```
@vAprobado char(30),
@vValidez char(1),
@vSituaciont char(9),
@vSituacionp char(9),
@vNumdoc numeric(10),
@vCorrelativo numeric(6)
 AS
UPDATE ORDEN_DE_TRABAJO SET
nombre_tecnico=@vName,trabajo_realizado=@vTrabajo,fecha_trabajoi=@vFe
chati,
fecha_trabajot=@vFechatt,aprobado_por=@vAprobado,validez_orden=@vValid
ez,
situacion_trabajo=@vSituaciont,situacion_pago=@vSituacionp,num_doc=@vNu
mdoc
WHERE num_orden=@vCorrelativo


CREATE PROCEDURE Mo_Pedido
@vsituacion char(9),
@vFecha datetime,
@vproveedor char(30),
@vCorrelativo numeric(10)
AS
UPDATE PEDIDO SET
situacion_pedido=@vsituacion,fecha_pedido=@vFecha,
proveedor=@vproveedor
WHERE num_pedido=@vCorrelativo


CREATE PROCEDURE Mo_Producto
@vsil real,
@Voriginalcil int,
@Vprocoml int,
@vCorrelativo numeric(10)
AS
UPDATE PRODUCTO SET
valor_original_si=@vsil,valor_original_ci=@Voriginalcil,
valor_procom_ci=@Vprocoml
WHERE cod_prod=@vCorrelativo
```

```
CREATE PROCEDURE Mo_Respcoti
@num_cot numeric(10),
@respcotiza text,
@p numeric(8)
AS
UPDATE COTIZACION SET respuesta_cotizacion=@respcotiza
WHERE rut_cliente=@p AND numero_doc=@num_cot
```

Ingreso:

```
CREATE PROCEDURE Ing_Calendario
@fecha datetime,
@tipotrabajo char(35),
@equipo char(35),
@p numeric(8)
AS
INSERT INTO CALENDARIO(fecha,tipo_trabajo,equipo,rut_cliente)
VALUES(@fecha,@tipotrabajo,@equipo,@p)
```

```
CREATE PROCEDURE Ing_Cliente
@rut numeric(8),
@dig char(1),
@nom char(30),
@dir char(30),
@ciu char(20),
@con char(30),
@tel numeric(10),
@mai char(30),
@pas char(10)
AS
INSERT INTO
CLIENTE(rut_cliente,dig_cliente,nombre_cliente,direccion_cliente,
ciudad_cliente,contacto_cliente,telefono_cliente,email_cliente,
password_cliente)
VALUES(@rut,@dig,@nom,@dir,@ciu,@con,@tel,@mai,@pas)
```

```
CREATE PROCEDURE Ing_CotizacionCli
@fechaemision datetime,
@detalle text,
@rut numeric(8)
AS
INSERT INTO COTIZACION(fecha,detalle_cotizacion,rut_cliente)
VALUES(@fechaemision,@detalle,@rut)


CREATE PROCEDURE Ing_DetalleFactura
@numdoc numeric(10),
@codtxt numeric(10),
@detalle char(110),
@cantidadtxt  numeric(4),
@valorpro int,
@especiftxt text
 AS
INSERT INTO
DETALLE_FACTURA(num_doc,cod_prod,detalle_producto,cantidad,valor_proc
om_ci,especificacion)
VALUES (@numdoc,@codtxt,@detalle,@cantidadtxt,@valorpro,@especiftxt)


CREATE PROCEDURE Ing_Factura
@numdoc numeric(10),
@origendoc char(13),
@fecha datetime,
@p numeric(8),
@neto real,
@iva real,
@total int,
@estado char(9),
@fpago char(10)
 AS
INSERT INTO
FACTURA(num_doc,origen_doc,fecha,rut_cliente,neto,iva,total,estado,forma_p
ago)
VALUES
(@numdoc,@origendoc,@fecha,@p,@neto,@iva,@total,@estado,@fpago)
```

```
CREATE PROCEDURE Ing_Ordentrabajo
@fechaemision datetime,
@tipotrabajo char(30),
@detalle text,
@solicitadopor char(30),
@rut numeric(8)
 AS
INSERT INTO ORDEN_DE_TRABAJO(
fecha_emision,tipo_trabajo,detalle_orden,solicitado_por,rut_cliente)
VALUES (@fechaemision,@tipotrabajo,@detalle,@solicitadopor,@rut)

CREATE PROCEDURE Ing_Pedido
@fecha datetime,
@Proveedor char(30)
 AS
INSERT INTO PEDIDO(fecha_pedido,proveedor)
VALUES(@fecha,@Proveedor)


CREATE PROCEDURE Ing_Producto
@Tproducto char(30),
@Dproducto char(110),
@vsi1 real,
@Voriginalci1 int,
@Vprocom1 int
 AS
INSERT INTO
PRODUCTO(tipo_producto,detalle_producto,valor_original_si,valor_original_ci,v
alor_procom_ci)
VALUES(@Tproducto,@Dproducto,@vsi1,@Voriginalci1,@Vprocom1)


CREATE PROCEDURE Ing_ProductoPedido
@num2 numeric(10),
@cod2 numeric(10),
@cant2 numeric(4),
@stock3 numeric(4)
AS
INSERT INTO PRODUCTO_PEDIDO(num_pedido,cod_prod,cantidad,stock)
VALUES (@num2,@cod2,@cant2,@stock3)
```

```
CREATE PROCEDURE Ing_VentaDirecta
@p numeric(8),
@num numeric(10)
AS
INSERT INTO VENTA_DIRECTA(rut_cliente,num_doc)
VALUES(@p,@num)
```

14.1.3 Triggers o Desencadenadores.

BORRAR:

```
CREATE TRIGGER Cliente_del ON CLIENTE
FOR DELETE
AS
begin
declare
@numrows int
SELECT @numrows=@@rowcount
   IF @numrows=0
       return
DELETE ORDEN_DE_TRABAJO
FROM ORDEN_DE_TRABAJO,deleted
WHERE ORDEN_DE_TRABAJO.rut_cliente=deleted.rut_cliente

DELETE FACTURA
FROM FACTURA,deleted
WHERE FACTURA.rut_cliente=deleted.rut_cliente

DELETE CALENDARIO
FROM CALENDARIO,deleted
WHERE CALENDARIO.rut_cliente=deleted.rut_cliente

DELETE COTIZACION
FROM COTIZACION,deleted
WHERE COTIZACION.rut_cliente=deleted.rut_cliente

DELETE VENTA_DIRECTA
FROM VENTA_DIRECTA,deleted
WHERE VENTA_DIRECTA.rut_cliente=deleted.rut_cliente
return
END
```

```
CREATE TRIGGER Factura_del ON FACTURA
FOR DELETE
AS
Begin
Declare
@numrows int

SELECT @numrows=@@rowcount
  IF @numrows=0
     return

DELETE ORDEN_DE_TRABAJO
FROM ORDEN_DE_TRABAJO,deleted
WHERE ORDEN_DE_TRABAJO.num_doc=deleted.num_doc

DELETE DETALLE_FACTURA
FROM DETALLE_FACTURA,deleted
WHERE DETALLE_FACTURA.num_doc=deleted.num_doc

DELETE VENTA_DIRECTA
FROM VENTA_DIRECTA,deleted
WHERE VENTA_DIRECTA.num_doc=deleted.num_doc

return
END

CREATE TRIGGER Pedido_del ON PEDIDO
FOR  DELETE
AS
Begin
Declare
  @errno int,
  @errmsg varchar(255),
  @numrows int

SELECT @numrows=@@rowcount
IF @numrows=0
    return
IF (EXISTS(SELECT PRODUCTO_PEDIDO.num_pedido
   FROM PRODUCTO_PEDIDO,deleted
   WHERE PRODUCTO_PEDIDO.num_pedido=deleted.num_pedido))
   Begin
      SELECT @errno=30006,
```

```
      @errmsg='Existen registros asociados en tabla "producto_pedido",
                    No se puede ejecutar operación Eliminar Pedido'
            goto error
    END

return
 error:
       Raiserror @errno @errmsg
      Rollback transaction
END

CREATE TRIGGER Producto_del ON PRODUCTO
FOR  DELETE
AS
Begin
Declare
@errno int,
@errmsg varchar(255),
@numrows int

Select @numrows= @@rowcount
  IF @numrows= 0
      return
  IF (Exists(Select DETALLE_FACTURA.cod_prod
    FROM DETALLE_FACTURA,deleted
    WHERE DETALLE_FACTURA.cod_prod=deleted.cod_prod))
    begin
      select @errno=30006, @errmsg='Registros Asociados a tabla detalle factura, No
puede ejecutar operación'
      goto error
    end
  IF (Exists (Select PRODUCTO_PEDIDO.cod_prod
      FROM PRODUCTO_PEDIDO,deleted
      WHERE PRODUCTO_PEDIDO.cod_prod=deleted.cod_prod))
     Begin
      Select @errno=3006, @errmsg='Registros Asociados a tabla Producto Pedido, No se
puede ejecutar operación'
      goto error
     End
  return
error:
   rollback tran
   return
```

```
end
```

INSERT:

```
CREATE TRIGGER Producto_In ON PRODUCTO
FOR INSERT
AS
Begin
Declare
    @errmsg varchar(255),
    @errno int
IF (Select count(*)
   From PRODUCTO,inserted
   Where  PRODUCTO.detalle_producto=inserted.detalle_producto)!=@@rowcount
   Begin
     Select @errno=30002,
     @errmsg='El producto ya ha sido Ingresado'
     goto error
   end
   Else
      Print 'Inserción Válida'
  Return

error:
     Raiserror @errno @errmsg
     rollback transaction
end


CREATE TRIGGER Cliente_In ON CLIENTE
FOR INSERT
AS
Begin
declare
    @errmsg varchar(255),
    @errno int
  IF (Select count(*)
      From CLIENTE,inserted
      Where CLIENTE.rut_cliente=inserted.rut_cliente)!=@@rowcount
      Begin
        Select @errno=30002,
        @errmsg='El Cliente ya ha sido Ingresado'
```

```
          goto error
        end
      else
         Print 'Inserción Válida'
      return
error:
      Raiserror @errno @errmsg
      rollback transaction
end

CREATE TRIGGER Factura_In ON  FACTURA
FOR INSERT
AS
Begin
Declare
   @errmsg varchar(255),
   @errno int

If (select count(*)
   from FACTURA,inserted
   Where FACTURA.num_doc=inserted.num_doc)!=@@rowcount
    Begin
         Select @errno=30002,
        @errmsg='La Factura ya ha sido Ingresado'
         goto error

    End
  Else
    Print 'Inserción Válida'
RETURN

error:
     Raiserror @errno @errmsg
     rollback transaction
end


CREATE TRIGGER ProductoPedido_In ON  PRODUCTO_PEDIDO
FOR INSERT
AS
Begin
Declare
   @errmsg varchar(255),
```

```
   @errno int

If (select count(*)
   from PRODUCTO_PEDIDO,inserted
   Where PRODUCTO_PEDIDO.num_pedido=inserted.num_pedido and
PRODUCTO_PEDIDO.cod_prod=inserted.cod_prod)!=@@rowcount
    Begin
         Select @errno=30002,
        @errmsg='El producto pedido ya ha sido Ingresado'
         goto error

    End
  Else
    Print 'Inserción Válida'
RETURN

error:
     Raiserror @errno @errmsg
     rollback transaction
end

CREATE TRIGGER VentaDirecta_In ON  VENTA_DIRECTA
FOR INSERT
AS
Begin
Declare
   @errmsg varchar(255),
   @errno int

If (select count(*)
   from VENTA_DIRECTA,inserted
   Where VENTA_DIRECTA.num_doc=inserted.num_doc)!=@@rowcount
    Begin
         Select @errno=30002,
        @errmsg='La Venta ya ha sido Ingresada'
         goto error
      End
  Else
    Print 'Inserción Válida'
RETURN

error:
     Raiserror @errno @errmsg
```

```
    rollback transaction
end

CREATE TRIGGER DetalleProducto_In ON DETALLE_FACTURA
FOR INSERT
AS
Begin
Declare
    @errmsg varchar(255),
    @errno int
IF (Select count(*)
   From DETALLE_FACTURA,inserted
   Where  DETALLE_FACTURA.cod_prod=inserted.cod_prod)!=@@rowcount
   Begin
     Select @errno=30002,
     @errmsg='El código del producto ya ha sido Ingresado en la Factura'
     goto error
   end
   Else
      Print 'Inserción Válida'
  Return

error:
     Raiserror @errno @errmsg
     rollback transaction
end
```

14.2 Implementación del sitio web.

La implementación del sitio web, comprende la realización del sitio por medio de páginas creadas con código en ASP y HTML.

14.2.1 Creación de código ASP y HTML para las páginas web.

Administración clientes:

```
<HTML>
<HEAD>
```

```
<TITLE>Admcliente.asp</TITLE>
</HEAD>
<body bgcolor="#5F6399" text="#CC0033" alink="#CCFF99" vlink="#33FFFF">
<TABLE BORDER=1 CELLPADDING="1" BGCOLOR="#000080"
ALIGN=CENTER>
<TR><TH ALIGN=RIGHT VALIGN=CENTER><H1><FONT FACE="MS Sans
Serif" SIZE="5" COLOR="#FFFFFF">Administración
Clientes</FONT></H1></TH>
</TR>
</TABLE><BR><BR>
<%
Set Conn = Server.CreateObject("ADODB.Connection")
Conn.Open "soporte","sa",""
%>

<FORM METHOD="post" ACTION="operacliente.asp" >
<TABLE BORDER="0" BGCOLOR="#CECECE" ALIGN=CENTER>
<TR>
<FONT FACE="arial,verdana" color="#CACEB3"size="2">Administrar
Clientes</FONT>
</TR>
<TR><TD>
<FONT FACE="arial, Arial"><FONT FACE="Arial" SIZE="2">Seleccione Cliente
         
          
</FONT></FONT></TD>
<%
'SQL="SELECT rut_cliente,nombre_cliente FROM CLIENTE"
Set RS=Conn.Execute("Obt_rutnomcli")
%>

<TD>
<CENTER><SELECT NAME="rut_cliente">
<OPTION VALUE="" >[Seleccione Cliente]

<%
Do While Not RS.Eof
%>

<OPTION VALUE=<%=RS("rut_cliente")%>><%=RS("nombre_cliente")%>

<%
RS.MoveNext
```

```
Loop
%>

</TD>
</TR>
</SELECT>
</CENTER>
<TR>
<TD COLSPAN="2" ALIGN="center">
<FONT FACE="Arial, Arial">
<INPUT TYPE="radio" NAME="Accion" VALUE="Ver" CHECKED>
<FONT FACE="Arial" SIZE="2"> Ver</FONT>
<INPUT TYPE="radio" NAME="Accion" VALUE="Eliminar">
<FONT FACE="Arial" SIZE="2"> Eliminar</FONT>
<INPUT TYPE="radio" NAME="Accion" VALUE="Modificar">
<FONT FACE="Arial" SIZE="2"> Modificar</FONT>
<INPUT TYPE="radio" NAME="Accion" VALUE="Ingresar">
<FONT FACE="Arial" SIZE="2"> Ingresar</FONT>
<INPUT TYPE="submit" NAME="Buscar" VALUE="Buscar">
</FONT></TD>
</TR>
</TABLE>
 <BR>
<CENTER><INPUT TYPE="SUBMIT" VALUE="Volver"
ONCLICK=history.back(-1)></CENTER>

</FORM>
<%Conn.Close
Set Conn = Nothing
Set RS   = Nothing
%>
</BODY>
</HTML>
```

Operaciones Cliente:

```
<HTML>
<HEAD>
<TITLE>Operacliente.asp</TITLE>

<!--#include file="../wwwroot/validarut.asp"-->

</HEAD>
```

```
<body bgcolor="#5F6399" text="#CC0033" alink="#CCFF99" vlink="#33FFFF">
<TABLE BORDER=1 CELLPADDING=1 BGCOLOR="#000080"
ALIGN=CENTER>
<TR><TH ALIGN=RIGHT VALIGN=CENTER><H1><FONT FACE="MS Sans
Serif" SIZE=5 COLOR="#FFFFFF">Operaciones sobre
Clientes</FONT></H1></TH>
</TR>
</TABLE><BR><BR>
<DIV ALIGN=CENTER>

<%
Const adOpenkeyset = 1

rut     = Request.Form("rut_cliente")
accion  = Request.Form("accion")
If accion="" Then
   accion=Request.QueryString("accion")
End If

Set Conexion = Server.CreateObject("ADODB.Connection")
Conexion.Open "soporte","sa",""

Select Case accion
Case "Ingresar"  'INGRESAR CLIENTE
  IF Session(autorizacion)=1 THEN
%>
<FONT FACE="Arial" COLOR="#CACEB3"SIZE=+1>Ingresar Nuevo
Cliente</FONT>
<BR>
<FORM METHOD="POST" ACTION="procesacliente.asp" NAME="miform"
ONSUBMIT="if((valido()) && (validorut())) return true; else return false;">
<TABLE border="1" cellpadding="1" cellspacing="2"
bgcolor="#CECECE"width="48%">
<TR>
<TD ALIGN=RIGHT VALIGN=CENTER>
<P> <FONT SIZE="2" FACE="Arial">Rut</FONT> </TD><TD ALIGN=LEFT
VALIGN=CENTER><INPUT TYPE=TEXT NAME="vlr" SIZE="8"
MAXLENGTH="8">
         -
<INPUT TYPE=TEXT NAME="dgt" SIZE="1" MAXLENGTH="1"></TD></TR>
<TR><TD ALIGN=RIGHT VALIGN=CENTER> <FONT SIZE="2"
FACE="Arial">Nombre </FONT></TD><TD ALIGN=LEFT VALIGN=CENTER>
```

```
<INPUT TYPE=TEXT NAME="nombre" SIZE="30"
MAXLENGTH="30"></TD></TR>
<TR><TD ALIGN=RIGHT VALIGN=CENTER> <FONT SIZE="2"
FACE="Arial">Dirección</FONT> </TD><TD ALIGN=LEFT VALIGN=CENTER>
<INPUT TYPE=TEXT NAME="direccion" SIZE="30"
MAXLENGTH="30"></TD></TR>
<TR><TD ALIGN=RIGHT VALIGN=CENTER><FONT SIZE="2" FACE="Arial">
Ciudad</FONT> </TD>
<TD ALIGN=LEFT VALIGN=CENTER>
<SELECT NAME=ciudad>
<OPTION VALUE="" >[Seleccione una ciudad]
<OPTION VALUE="Ancud" >Ancud
<OPTION VALUE="Castro" >Castro
<OPTION VALUE="Calbuco" >Calbuco
<OPTION VALUE="Osorno" >Osorno
<OPTION VALUE="Puerto Montt">Puerto Montt
<OPTION VALUE="Puerto Varas" >Puerto Varas
<OPTION VALUE="Valdivia">Valdivia
<OPTION VALUE="Otra">Otra
</SELECT></TD></TR>
<TR><TD ALIGN=RIGHT VALIGN=CENTER><FONT SIZE="2" FACE="Arial">
Contacto</FONT> </TD> <TD ALIGN=LEFT VALIGN=CENTER>
<INPUT TYPE=TEXT NAME="contacto" SIZE="30"
MAXLENGTH="30"></TD></TR>
<TR><TD ALIGN=RIGHT VALIGN=CENTER><FONT SIZE="2" FACE="Arial">
Teléfono</FONT> </TD> <TD ALIGN=LEFT VALIGN=CENTER>
<INPUT TYPE=TEXT NAME="telefono" SIZE="10"
MAXLENGTH="10"></TD></TR>
<TR><TD ALIGN=RIGHT VALIGN=CENTER><FONT SIZE="2" FACE="Arial">
E-Mail</FONT> </TD> <TD ALIGN=LEFT VALIGN=CENTER>
<INPUT TYPE=TEXT NAME="mail" SIZE="30"
MAXLENGTH="30"></TD></TR>
<TR><TD ALIGN=RIGHT VALIGN=CENTER><FONT SIZE="2" FACE="Arial">
Password Cliente</FONT> </TD><TD ALIGN=LEFT VALIGN=CENTER>
<INPUT TYPE=TEXT NAME="password" SIZE="10"
MAXLENGTH="10"></TD></TR>
<TR><TD></TD><TD ALIGN=RIGHT VALIGN=MIDDLE>
<INPUT TYPE=HIDDEN NAME="accion" VALUE="Ingresar">
<INPUT TYPE=SUBMIT VALUE="Insertar">
</TD><TR>
</TABLE><BR>
<CENTER><INPUT type="SUBMIT" value="Cancelar" onclick=history.back(-
1)></CENTER>
```

```
</P>
</FORM>

<% ELSE%>

      <CENTER><FONT SIZE="4" COLOR="#CACEB3">"Permiso Denegado al
Ejecutar esta Acción"</font>
      <CENTER><INPUT type="SUBMIT" value="Volver" onclick=history.back(-
1)></CENTER>

   <% END IF %>

<%
Case "Ver"  'VER CLIENTE
  IF Session(autorizacion)=1 OR Session(autorizacion)=2 OR
Session(autorizacion)=3 OR Session(autorizacion)=4 THEN
%>

<FONT FACE="Arial" COLOR="#CACEB3"SIZE=+1>Listado de
clientes</FONT>
<BR><BR>

<%
Set RS = Server.CreateObject("ADODB.Recordset")
RS.PageSize = 10

SQL=" SELECT rut_cliente,nombre_cliente FROM CLIENTE"

RS.Open SQL, Conexion, adOpenKeyset

If RS.EOF And RS.BOF then
%>

<CENTER><h3><FONT SIZE="4" COLOR="#CACEB3"> No hay Clientes
disponibles</FONT></h3></center>

<%
Else
   If Request.QueryString("pagina") = "" Then
      RS.AbsolutePage = 1
   Else
      RS.AbsolutePage = Request.QueryString("pagina")
   End If%>
```

```
<CENTER>
<FONT SIZE=-2 COLOR="#CACEB3"FACE="Arial">
Página

<%
If Request.QueryString("pagina") = "" Then
  Response.Write(RS.AbsolutePage)
Else
  Response.Write(Request.QueryString("pagina"))
End If
%>

de

<% = RS.PageCount %>

</FONT>
</CENTER>
<CENTER>
<TABLE BORDER=0>
<TR>

<%
For p = 1 To RS.PageCount
%>

<TD ALIGN=CENTER>
<FONT SIZE=4 COLOR="red" FACE="Arial">
<A HREF="operacliente.asp?accion=Ver&pagina=<% = p %>"><% = p %></A>
</FONT>
</TD>

<%
Next
%>

<TABLE BORDER="1" CELLPADDING="0" CELLSPACING="2"
BGCOLOR="#CECECE">
<TR>
<TH WIDTH="60%"><FONT FACE="Arial">Cliente </FONT></TH>
<TH> </TH>
</TR>
```

```
<%
For i = 1 To RS.PageSize
  If RS.EOF Then
    Exit For
  End If
%>

<FORM METHOD=POST ACTION="procesacliente.asp">
<TR>
<TD><%=RS("nombre_cliente")%></TD>
<TD>
<INPUT TYPE=HIDDEN NAME="vlr" VALUE=<%=RS("rut_cliente")%>>
<INPUT TYPE=HIDDEN NAME="accion" VALUE="Ver">
<INPUT TYPE=SUBMIT VALUE="Ver Cliente">
</TD>
</TR>
</FORM>

<%
RS.MoveNext
Next
%>

</TABLE><BR>
<INPUT type="SUBMIT" value="Cancelar" onclick=history.back(-1)>
  <%
End If%>
<% ELSE%>

    <CENTER><FONT SIZE="4" COLOR="#CACEB3">"Permiso Denegado al
Ejecutar esta Acción"</font>
    <CENTER><INPUT type="SUBMIT" value="Volver" onclick=history.back(-
1)></CENTER>

<%
End If
Case "Modificar"  'MODIFICAR CLIENTE
IF Session(autorizacion)=1 THEN
If rut<>"" And Not Isnull(rut) Then
  Set RS=conexion.execute("Obt_Cliente('&rut&')")
%>
<FONT FACE="Arial"COLOR="#CACEB3" SIZE=+1>Modificar Cliente</FONT>
<BR><BR>
```

```
<FORM METHOD="POST" ACTION="procesacliente.asp" NAME="miform">
<TABLE border="1" cellpadding="1" cellspacing="2" bgcolor="#CECECE"
width="60%">
<TR>
<TR><TD ALIGN=RIGHT VALIGN=CENTER ><FONT SIZE="2" FACE="Arial">
Nombre</FONT> </TD><TD ALIGN=LEFT VALIGN=CENTER>
<INPUT TYPE=TEXT NAME="nombre" SIZE="30" MAXLENGTH="30"
VALUE="<%=RS("nombre_cliente")%>"></TD></TR>
<TR><TD ALIGN=RIGHT VALIGN=CENTER> <FONT SIZE="2"
FACE="Arial">Dirección</FONT> </TD><TD ALIGN=LEFT VALIGN=CENTER>
<INPUT TYPE=TEXT NAME="direccion" SIZE="30" MAXLENGTH="30"
VALUE="<%=RS("direccion_cliente")%>"></TD></TR>
<TR><TD ALIGN=RIGHT VALIGN=CENTER><FONT SIZE="2" FACE="Arial">
Ciudad</FONT> </TD>
<TD ALIGN=LEFT VALIGN=CENTER>
<SELECT NAME=ciudad>
<OPTION VALUE="<%=RS("ciudad_cliente")%>"><%=RS("ciudad_cliente")%>
<OPTION VALUE="" >[Seleccione una ciudad]
<OPTION VALUE="Ancud" >Ancud
<OPTION VALUE="Castro" >Castro
<OPTION VALUE="Calbuco" >Calbuco
<OPTION VALUE="Osorno" >Osorno
<OPTION VALUE="Puerto Montt">Puerto Montt
<OPTION VALUE="Puerto Varas" >Puerto Varas
<OPTION VALUE="Valdivia">Valdivia
<OPTION VALUE="Otra">Otra
</SELECT></TD></TR>
<TR><TD ALIGN=RIGHT VALIGN=CENTER><FONT SIZE="2" FACE="Arial">
Contacto</FONT> </TD> <TD ALIGN=LEFT VALIGN=CENTER>
<INPUT TYPE=TEXT NAME="contacto" SIZE="30" MAXLENGTH="30"
VALUE="<%=RS("contacto_cliente")%>"></TD></TR>
<TR><TD ALIGN=RIGHT VALIGN=CENTER> <FONT SIZE="2"
FACE="Arial">Teléfono</FONT> </TD> <TD ALIGN=LEFT VALIGN=CENTER>
<INPUT TYPE=TEXT NAME="telefono" SIZE="10" MAXLENGTH="10"
VALUE="<%=RS("telefono_cliente")%>"></TD></TR>
<TR><TD ALIGN=RIGHT VALIGN=CENTER><FONT SIZE="2" FACE="Arial">
E-Mail</FONT> </TD> <TD ALIGN=LEFT VALIGN=CENTER>
<INPUT TYPE=TEXT NAME="mail" SIZE="30" MAXLENGTH="30"
VALUE="<%=RS("email_cliente")%>"></TD></TR>
<TR><TD ALIGN=RIGHT VALIGN=CENTER><FONT SIZE="2" FACE="Arial">
Password Cliente</FONT> </TD><TD ALIGN=LEFT VALIGN=CENTER>
<INPUT TYPE=TEXT NAME="password" SIZE="10" MAXLENGTH="10"
VALUE="<%=RS("password_cliente")%>"></TD></TR>
```

```
<TR><TD></TD><TD ALIGN=RIGHT VALIGN=MIDDLE>
<INPUT TYPE=HIDDEN NAME="vlr" VALUE=<%=rut%>>
<INPUT TYPE=HIDDEN NAME="accion" VALUE="Modificar">
<INPUT TYPE=SUBMIT VALUE="Actualizar">
</TD></TR>
</TABLE>
</P>
<INPUT type="SUBMIT" value="Cancelar" onclick=history.back(-1)>
</FORM>

<%
Else
   Response.write "<div align='center'>"
   Response.write "<font face='Verdana' size='2'>"
   Response.write "Se ha provocado un error, debe seleccionar un cliente..."
   Response.write "<br>"
   Response.write "<a href='admcliente.asp'>Regresar</a>"
   Response.write "</font>"
   Response.write "</div>"

End If %>
<% ELSE%>

    <CENTER><FONT SIZE="4" COLOR="#CACEB3">"Permiso Denegado al
Ejecutar esta Acción"</font>
    <CENTER><INPUT type="SUBMIT" value="Volver" onclick=history.back(-1)>
</CENTER>

<%End If %>
<%Case "Eliminar"  'ELIMINAR CLIENTE
   IF Session(autorizacion)=1 THEN
   SQL=" SELECT rut_cliente,nombre_cliente FROM CLIENTE"
   Set RS = Conexion.Execute(SQL)
  ' Set RS = Conexion.Execute("Obt_rutnomcli")
%>

<FONT FACE="Arial" COLOR="#CACEB3"SIZE=+1>Eliminar Cliente</FONT>
<BR>
<FORM ACTION="procesacliente.asp" METHOD="post">
<TABLE BORDER="1" CELLPADDING="0"
CELLSPACING="2"bgcolor="#CECECE" >
<TR>
```

```
<TD WIDTH="40%"><STRONG><FONT FACE="Arial">Cliente
</FONT></STRONG></TD>
<TD WIDTH="20%"></TD>
</TR>

<%
Do While Not RS.Eof
%>

<TR>
<TD><FONT SIZE="2" FACE="Arial"COLOR="#400040">
<%=RS("nombre_cliente")%></FONT></TD>
<TD ><INPUT TYPE="checkbox" NAME="chequeo"
VALUE=<%=RS("rut_cliente")%> ></FONT>
</TD>
</TR>

<%
RS.Movenext
Loop
%>

</TABLE><BR>
<INPUT TYPE=HIDDEN NAME="accion" VALUE="Eliminar">
<INPUT TYPE=SUBMIT VALUE="Eliminar">  
<INPUT type="SUBMIT" value="Cancelar" onclick=history.back(-1)>
</FORM>
 <% ELSE%>

      <CENTER><FONT SIZE="4" COLOR="#CACEB3">"Permiso Denegado al
Ejecutar esta Acción"</font>
<CENTER><INPUT type="SUBMIT" value="Volver" onclick=history.back(-1)>
</CENTER>

   <% END IF %>
<%
End Select

Conexion.Close
Set Conexion = Nothing
Set RS       = Nothing
%>
</DIV></BODY></HTML>
```

Operaciones calendario:

```
<HTML>
<HEAD>
<TITLE>OperacionesCalendario.asp</TITLE>
</HEAD>
 <BODY bgcolor="#5F6399" text="#CC0033" alink="#CCFF99"
vlink="#33FFFF">
 <BODY>
 <CENTER><TABLE border=1 cellpadding="1" bgCOLOR="#000080" >
 <TR><TH align=right valign=center><h1><FONT face="MS Sans Serif"
SIZE="5" COLOR="#FFFFFF">Operaciones  Calendario</FONT></h1></TH>
 </TR>
 </TABLE></CENTER><BR><BR>

<%
 p=request.form("Cliente")
 SESSION("p")=REQUEST.FORM("Cliente")

 Set Conn = Server.CreateObject("ADODB.Connection")
 Conn.Open "DSN=soporte","sa",""
%>
<% SESSION("R")=p %>
<%IF  REQUEST.FORM("Cliente")="" THEN %>
   <CENTER><FONT SIZE="4" COLOR="#CACEB3">NO SELECCIONÓ
CLIENTE A BUSCAR</font>
   <FORM METHOD=POST ACTION="AdmCalendario.asp" >
                <INPUT TYPE="submit" VALUE="Volver">
        </FORM>
        </CENTER>
 <%ELSE%>
 <% '*********  LISTADO  ***********  %>

<%
IF REQUEST("Buscar")<>"" AND REQUEST("Accion")= "Listar" THEN
  IF Session(autorizacion)=1 OR Session(autorizacion)=2 OR
Session(autorizacion)=4 THEN %>

<%
  set RS = Conn.Execute("Obt_Calendario('&p&')")
 %>
```

```
<CENTER><H1><FONT SIZE="5"
COLOR="#CECECE">LISTADO</FONT></H1></CENTER>
<CENTER>
<TABLE border="1" cellpadding="1" cellspacing="2" bgcolor="#CECECE"
width="80%">
<TR>
<TH width="18%"><strong><FONT
FACE="Arial"SIZE="2">Código</FONT></strong></TH>
<TH width="15%"><strong><FONT
FACE="Arial"SIZE="2">Fecha</FONT></strong></TH>
<TH width="33%"><strong><FONT FACE="Arial"SIZE="2">Tipo de
Trabajo</FONT></strong></TH>
<TH width="33%"><strong><FONT
FACE="Arial"SIZE="2">Equipo</FONT></strong></TH>
</TR>

<% WHILE NOT RS.EOF %>

<TR>
<TD width="18%"><FONT SIZE="2px" COLOR="#400040" FACE="Arial"
>  <%=RS.Fields("correl")%></FONT></TD>
 <TD width="15%"><FONT SIZE="2px" COLOR="#400040" FACE="Arial"
>  <%=RS.Fields("fecha")%></FONT></TD>
<TD width="33%"><FONT SIZE="2px" COLOR="#400040"
FACE="Arial" >  <%=RS.Fields("tipo_trabajo")%></FONT></TD>
<TD width="33%"><FONT SIZE="2px" COLOR="#400040" FACE="Arial">
  <%=RS.Fields("equipo")%></FONT></TD>
</TR>
<%
            RS.MOVENEXT
           WEND
%>
</TABLE></CENTER><BR>
<CENTER>
<FONT face="Arial">
<INPUT type="SUBMIT" value="Imprimir" onclick="Imprimir()">
<INPUT type="SUBMIT" value="Cancelar" onclick=history.back(-1)>
</FONT>
</CENTER>
<% ELSE%>

<CENTER><FONT SIZE="4" COLOR="#CACEB3">"Permiso Denegado al
Ejecutar esta Acción"</font>
```

```
<CENTER><INPUT    type="SUBMIT"    value="Volver"    onclick=history.back(-
1)></CENTER>

  <% END IF %>

 <% '***** ELIMINACION ******  %>
 <% ELSE
   IF REQUEST("Buscar")<>"" AND REQUEST("Accion")= "Eliminar" THEN
     IF Session(autorizacion)=1 THEN

     set RS = Conn.Execute("Obt_Calendario('&p&')")
  %>

<CENTER><H1><FONT SIZE="5"
COLOR="#CECECE">ELIMINAR</FONT></H1></CENTER>

<FORM ACTION="EliminacionCalendario.asp" METHOD="post">
<CENTER>
<TABLE border="1" cellpadding="1" cellspacing="2" bgcolor="#CECECE"
width="80%">
<TR>
<TH width="18%"><strong><FONT
FACE="Arial"SIZE="2">Código</FONT></strong></TH>
<TH width="15%"><strong><FONT
FACE="Arial"SIZE="2">Fecha</FONT></strong></TH>
<TH width="33%"><strong><FONT FACE="Arial"SIZE="2">Tipo de
Trabajo</FONT></strong></TH>
<TH width="33%"><strong><FONT
FACE="Arial"SIZE="2">Equipo</FONT></strong></TH>
<TH width="22%"><strong><FONT
FACE="Arial"SIZE="2">Selección</FONT></strong></TH>
</TR>

<% WHILE NOT RS.EOF %>

<TR>
<TD width="18%"><FONT SIZE="2px" COLOR="#400040" FACE="Arial">
  <%=RS.Fields("correl")%></FONT></TD>
<TD width="15%"><FONT SIZE="2px" COLOR="#400040" FACE="Arial">
  <%=RS.Fields("fecha")%></FONT></TD>
<TD width="33%"><FONT SIZE="2px" COLOR="#400040" FACE="Arial">
  <%=RS.Fields("tipo_trabajo")%></FONT></TD>
```

```
<TD width="33%"><FONT SIZE="2px" COLOR="#400040" FACE="Arial">
  <%=RS.Fields("equipo")%></FONT></TD>
<TD width="22%"><FONT SIZE="2px" COLOR="#400040" FACE="Arial">
<INPUT TYPE="CHECKBOX" NAME="CHEQUEO"
value=<%=RS.fields(0)%>></FONT>
</TD>
</TR>

<% RS.MOVENEXT
WEND
%>
</TABLE></CENTER><BR>
<CENTER>
<FONT FACE="Arial">
<INPUT type="SUBMIT" value="Eliminar">  
<INPUT type="SUBMIT" value="Cancelar" onclick=history.back(-1)>
</FONT>
</CENTER>
</FORM>

<% ELSE%>

<CENTER><FONT SIZE="4" COLOR="#CACEB3">"Permiso Denegado al
Ejecutar esta Acción"</font><BR><BR>
<CENTER><INPUT type="SUBMIT" value="Volver" onclick=history.back(-
1)></CENTER>

  <% END IF %>

<% '************************* MODIFICAR ******************************** %>

<% ELSE
    IF REQUEST("Buscar")<>"" AND REQUEST("Accion")= "Modificar" THEN
     IF Session(autorizacion)=1 THEN

       set RS = Conn.Execute("Obt_Calendario('&p&')")
  %>

<CENTER><H1><FONT SIZE="5"
COLOR="#CECECE">MODIFICAR</FONT></H1></CENTER>

<FORM ACTION="ModificacionCalendario.asp" METHOD="Post">
<CENTER>
```

```
<TABLE border="1" cellpadding="1" cellspacing="2" bgcolor="#CECECE"
width="80%">
<TR>
<TH width="18%"><strong><FONT
FACE="Arial"SIZE="2">Código</FONT></strong></TH>
<TH width="15%"><strong><FONT
FACE="Arial"SIZE="2">Fecha</FONT></strong></TH>
<TH width="33%"><strong><FONT FACE="Arial"SIZE="2">Tipo de
Trabajo</FONT></strong></TH>
<TH width="33%"><strong><FONT
FACE="Arial"SIZE="2">Equipo</FONT></strong></TH>
<TH width="22%"><strong><FONT
FACE="Arial"SIZE="2">Selección</FONT></strong></TH>
</TR>
<% WHILE NOT RS.EOF %>
<TR>
<TD width="18%"><FONT SIZE="2px" COLOR="#400040" FACE="Arial">
  <%=RS.Fields("correl")%></FONT></TD>
<TD width="15%">
<INPUT name="<%Response.Write("F" + cstr(RS.fields(0)))%>"
type="TEXT" value="<%=RS.Fields("fecha")%>">
</TD>
<TD width="33%">
<FONT SIZE="2px" COLOR="#400040" FACE="Arial">
<SELECT name=<%Response.Write("T" + cstr(RS.fields(0)) )%>>
<option selected value="<%=RS.Fields("tipo_trabajo")%>">
<%=RS.Fields("tipo_trabajo")%>
</option>
<option value="Instal.Software">Instal.Software</option>
<option value="Instal.Hardware">Instal.Hardware</option>
<option value="Mantención Computadores">Mantención
Computadores</option>
<option value="Mantención Impresoras">Mantención Impresoras</option>
<option value="Recuperar equipos infectados">Recuperar equipos
infectados</option>
<option value="Instal.y configuración Red">Instal.y configuración Red</option>
<option value="Otros">Otros</option>
</SELECT>
</FONT>
</TD>
<TD width="33%">
<INPUT type="TEXT" name="<%Response.Write("E" + cstr(RS.fields(0)))%>"
value="<%=RS.Fields("equipo")%>">
```

```
</TD>
<TD width="22%">
<INPUT type="CHECKBOX" name="CHEQUEO" id="CHEQUEO"
value=<%=RS.fields(0)%>>
</TD>
</TR>

<%      RS.MOVENEXT
         WEND
%>
</TABLE></CENTER><BR>
<CENTER>
<INPUT type="SUBMIT" value="Modificar" onClick="CargarCambios()">
  
 <INPUT type="SUBMIT" value="Cancelar" onclick=history.back(-1)>
 </CENTER>
 </FORM>

<% ELSE%>

 <CENTER><FONT SIZE="4" COLOR="#CACEB3">"Permiso Denegado al
Ejecutar esta Acción"</font><BR><BR>
<CENTER><INPUT type="SUBMIT" value="Volver" onclick=history.back(-
1)></CENTER>

<% END IF %>

<%   '***************AGREGAR UN REGISTRO*************%>
<%
  ELSE
    IF REQUEST("Buscar")<>"" AND REQUEST("Accion")="Ingresar" THEN %>
  <% IF Session(autorizacion)=1 THEN %>

<FORM ACTION="IngresoCalendario.asp" METHOD="Post">
<CENTER>
<TABLE border="1" cellpadding="1" cellspacing="2" bgcolor="#CECECE">
<TR>
<TD align=right valign=center>
<FONT SIZE="2" FACE="Arial">
 Fecha :<BR><BR>
</FONT>
</TD>
<TD align=left valign=center>
```

```
<input type="text" name="Fecha" size="10" maxlength="10">
<FONT SIZE="2" FACE="Arial">
00-00-0000
</FONT><BR><BR>
</TD>
</TR>
<TR>
<TD align=right valign=center>
<FONT SIZE="2" FACE="Arial">
Tipo de Trabajo :<BR><BR>
</FONT>
</TD>
<TD align=left valign=center>
<SELECT name="TipoTrabajo">
<option value="Instal.Software">Instal.Software</option>
<option selected value="Instal.Hardware">Instal.Hardware</option>
<option value="Mantención Computadores">Mantención
Computadores</option>
<option value="Mantención Impresoras">Mantención Impresoras</option>
<option value="Recuperar equipos infectados">Recuperar equipos
infectados</option>
<option value="Instal.y configuración Red">Instal.y configuración Red</option>
<option value="Otros">Otros</option>
</SELECT><BR><BR>
</TD>
</TR>
<TR>
<TD align=right valign=center>
<FONT SIZE="2" FACE="Arial">
Equipo (o) Ubicación de Trabajo :<BR><BR>
</FONT>
</TD>
<TD align=left valign=center>
<input type="text" name="Equipo" size="35" maxlength="35"><BR><BR>
</TD>
</TR>
<TR>
<TD align=right valign=center>
<INPUT TYPE="SUBMIT" value="Enviar">
</TD>
<TD align=left valign=center>
<INPUT TYPE="RESET" VALUE="Limpiar">
</TD>
```

```
</TR>
</TABLE></CENTER><BR>
<CENTER>
<INPUT type="SUBMIT" value="Cancelar" onclick=history.back(-1)>
</CENTER>
</FORM>
<% ELSE%>
<CENTER><FONT SIZE="4" COLOR="#CACEB3">"Permiso Denegado al
Ejecutar esta Acción"</font><BR><BR>
<CENTER><INPUT type="SUBMIT" value="Volver" onclick=history.back(-
1)></CENTER>

<% END IF
     END IF
    END IF
   END IF
  END IF
END IF%>

<%Conn.Close %>
</BODY>
</HTML>

<script>

function Imprimir()
{
        window.print();
};

function CargarCambios()
{
        var checks;
        var i,largo;
        var fecha,tipo,equipo;

        checks = document.all('CHEQUEO');

        if (typeof(checks.length)=='undefined')
        {
                largo=1;
        }
        else
```

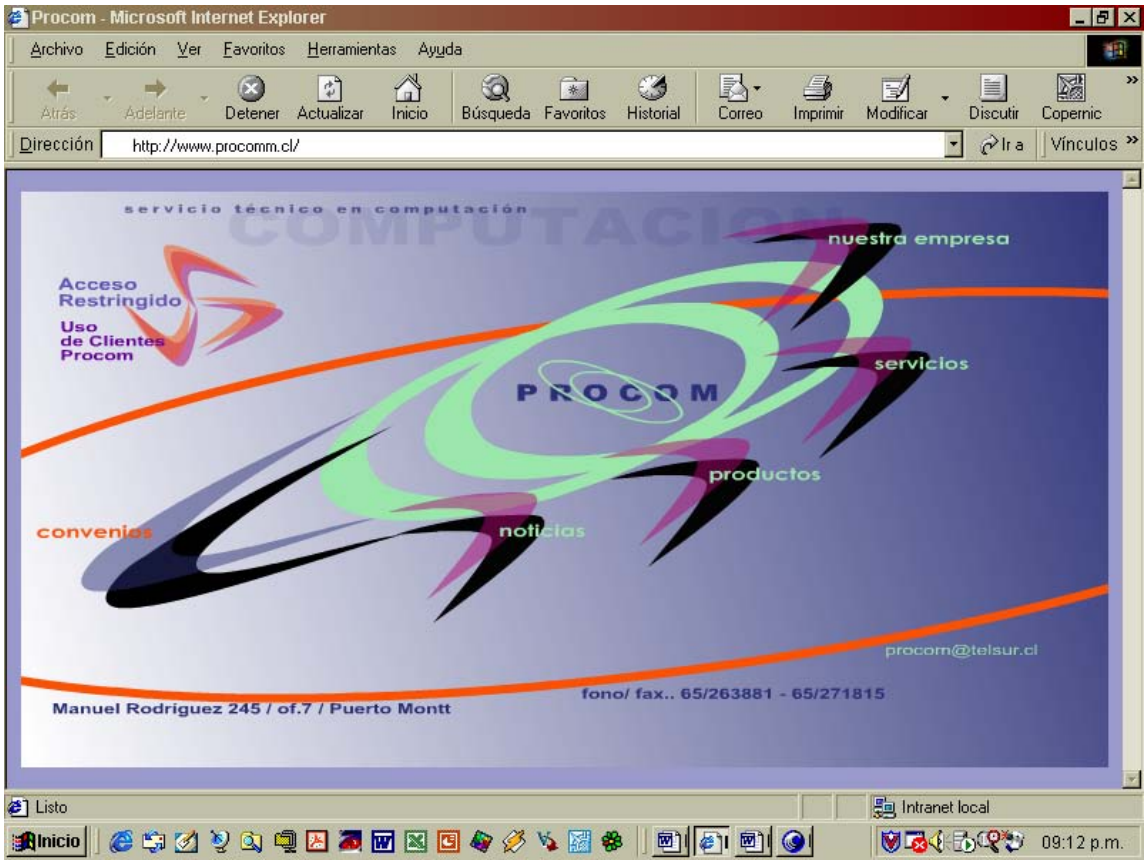
```
        {
                largo=checks.length;
        };

        for (i=0;i<largo;i++)
        {
                if (checks(i).checked)
                {

                        fecha = document.all('F'+checks(i).value).value
                        tipo = document.all('T'+checks(i).value).value
                        equipo = document.all('E'+checks(i).value).value
                        checks(i).value = checks(i).value + "&" + fecha + "&" + tipo +
"&" + equipo
                                        };
        };

};

</script>
```

14.2.2 Vistas de distintas páginas web que forman parte del sistema.

A continuación, se dan a conocer algunas pantallas con las que interactúan clientes y personal de empresa, con este sistema.

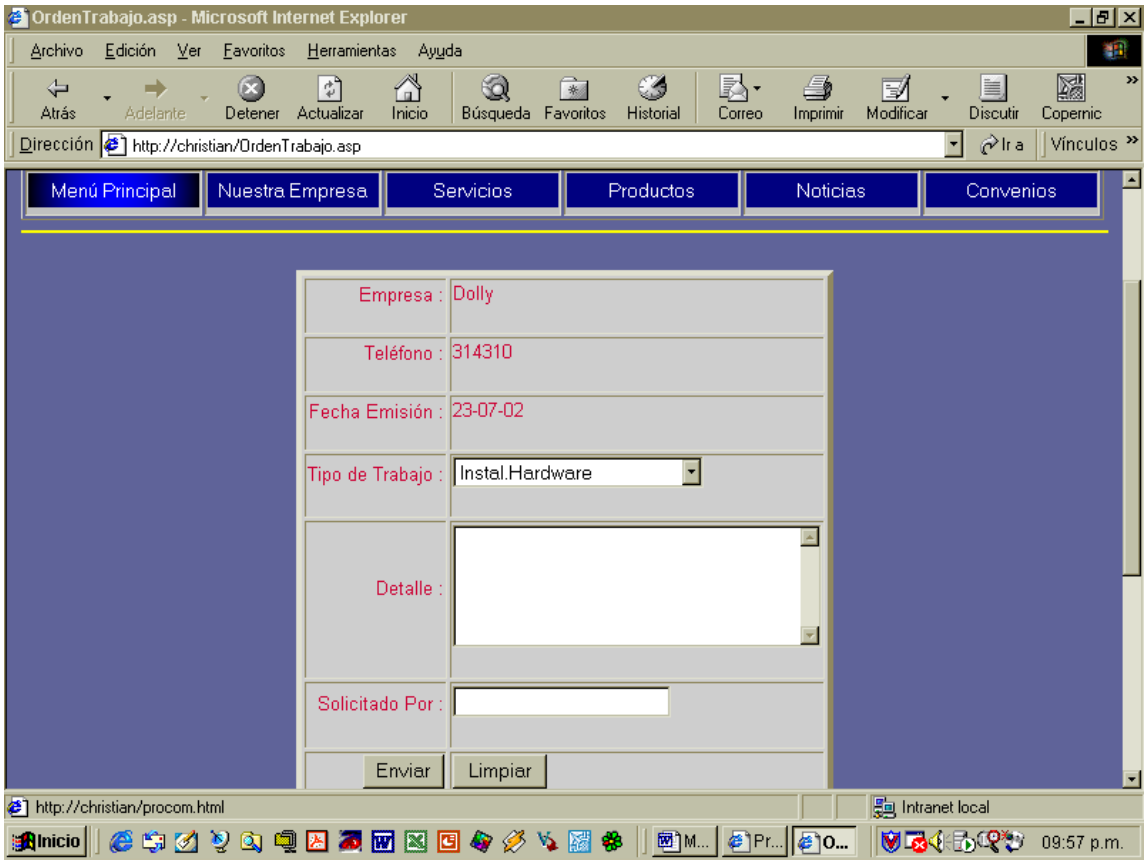
14.2.2.1 Pantalla “portada principal”:

Figura Nº 20. Portada Principal.



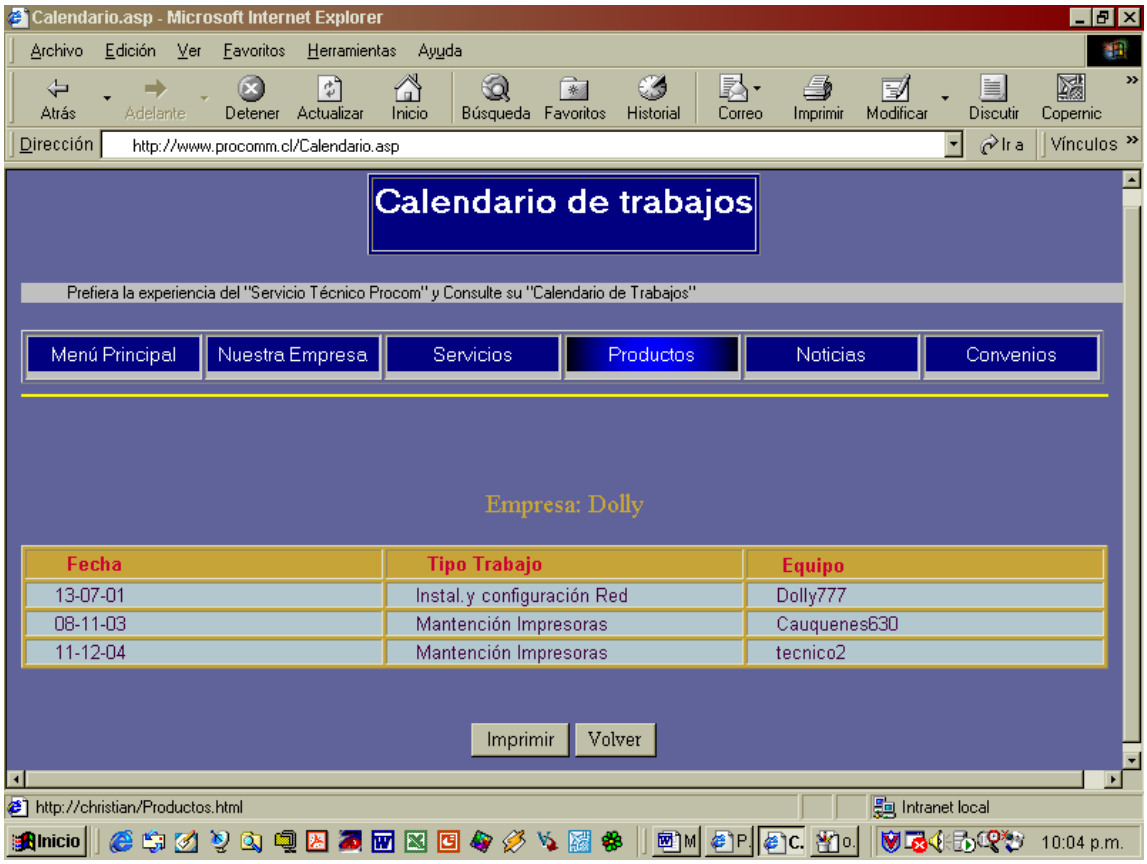
14.2.2.2 Formulario Orden de trabajo para clientes:

Figura Nº 21. Orden de trabajo para Clientes.



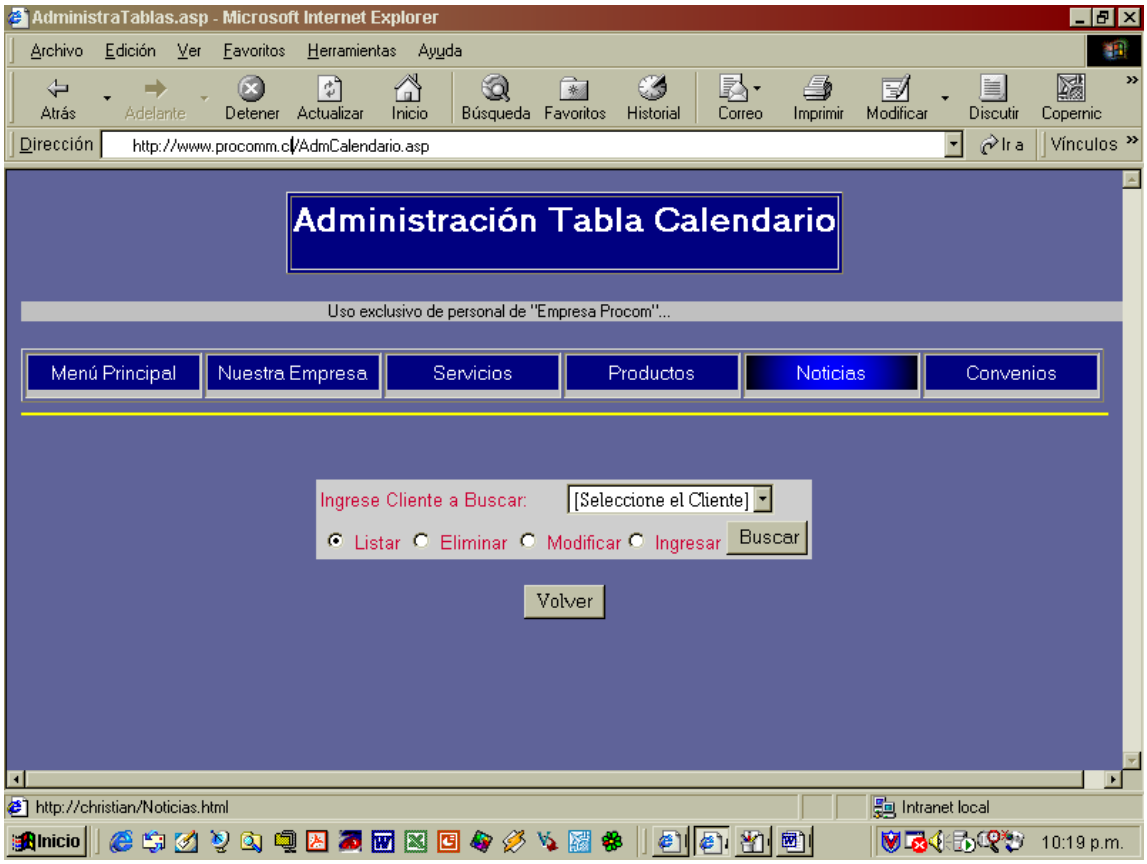
14.2.2.3 Pantalla Calendario de trabajos:

Figura Nº 22. Calendario de trabajos.



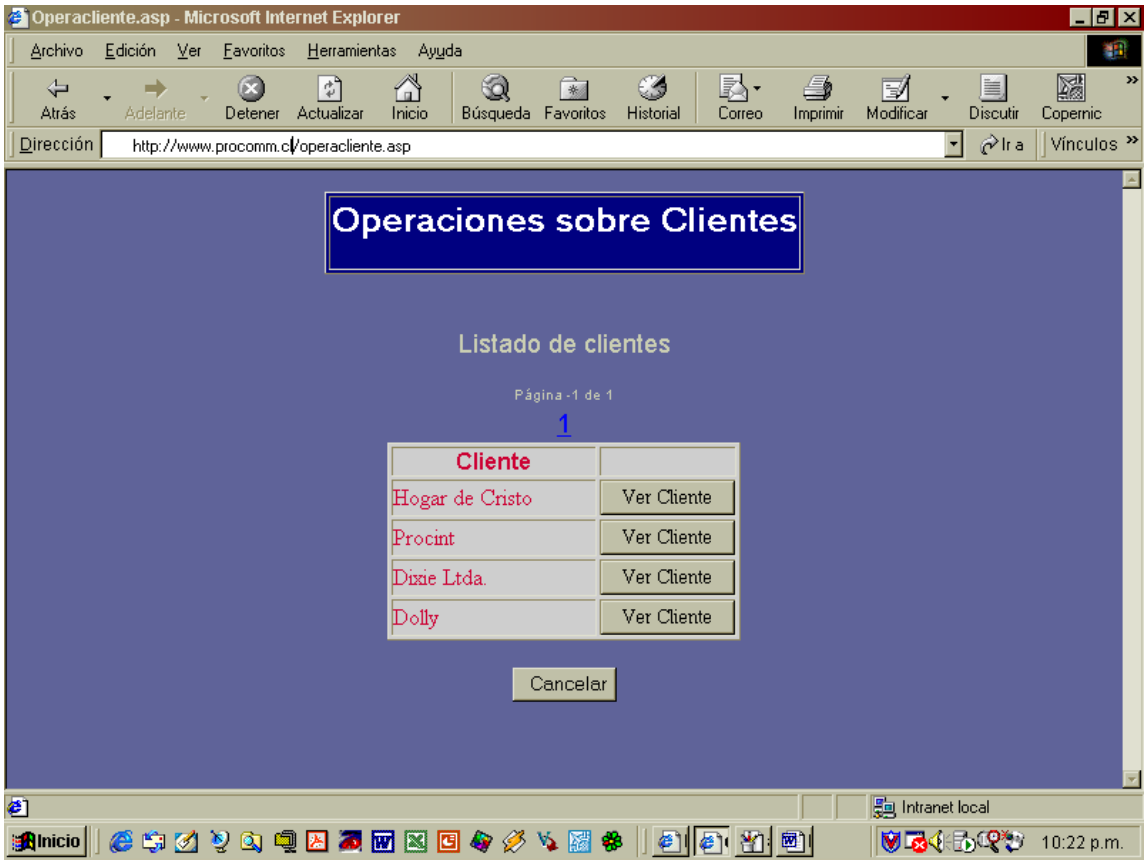
14.2.2.4 Pantalla Administración calendario de trabajos:

Figura Nº 23. Administración calendario de trabajos



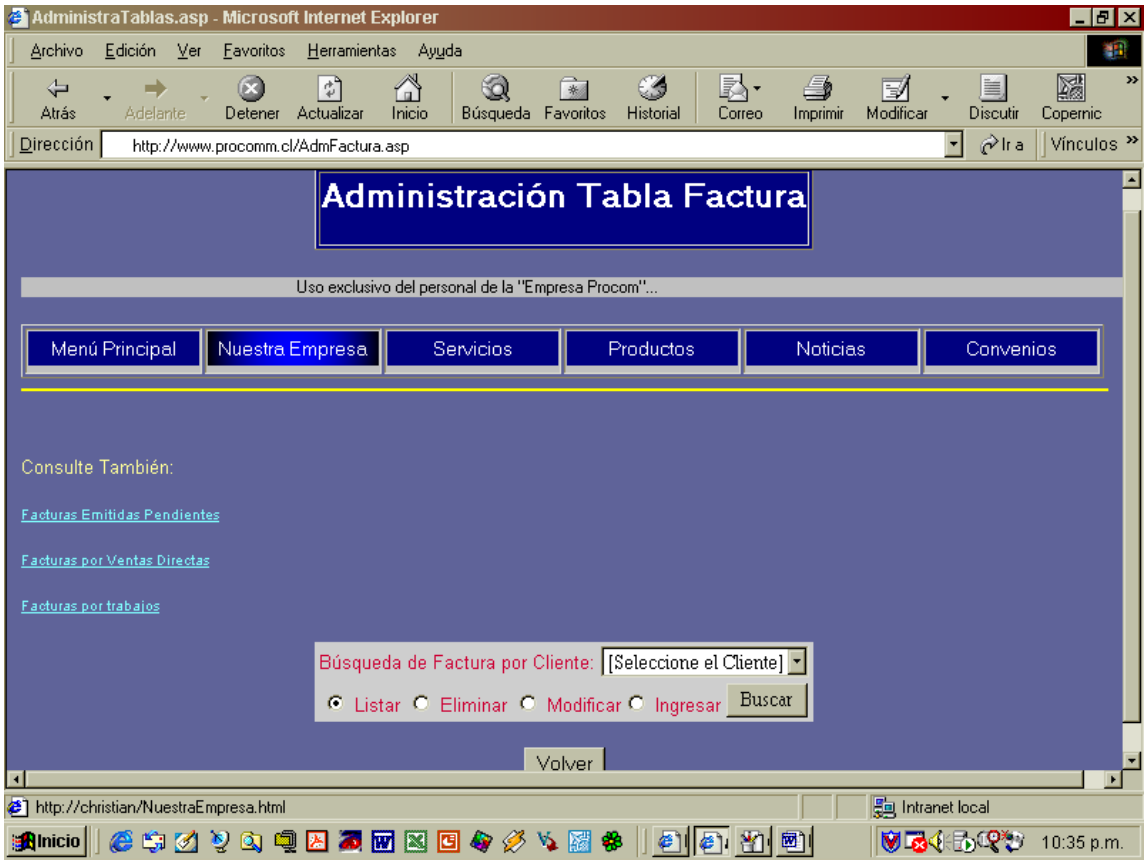
14.2.2.5 Pantalla operaciones con clientes:

Figura Nº 24. Operaciones con Clientes.



14.2.2.6 Pantalla administración factura:

Figura Nº 25. Administración Factura.



14.2.2.7 Pantalla lista de facturas pendientes:

Figura Nº 26. Listado de facturas pendientes.

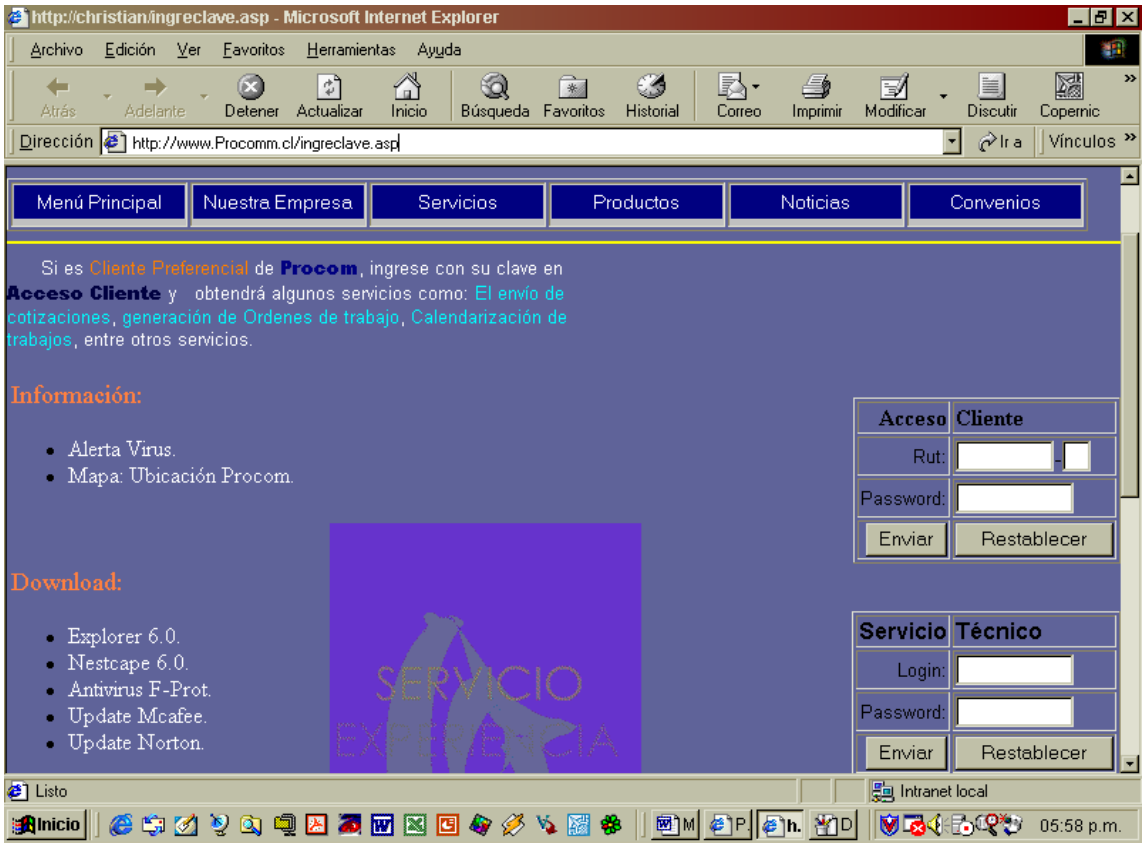
Nombre Cliente	Número Documento	Origen Documento	Fecha	Neto	Iva	Total	Estado
Hogar de Cristo	13121351	Ventas	13-11-02	15000	17000	17000	Emitida
Dolly	999555444	Ventas	14-06-02	50000	65000	65000	Emitida
Dolly	7777777	Ventas	29-06-02	55000	65000	65000	Emitida
Procint	12345678	Ventas	13-11-02	45000	55000	55000	Emitida
Dolly	15155475	Ventas	13-02-02	15000	17000	17000	Emitida

14.2.2.8 Pantalla modifica calendario de un determinado cliente:

Figura Nº 27. Modifica calendario.

14.2.2.9 Pantalla ingreso de claves.

Figura Nº 28. Ingreso de claves.



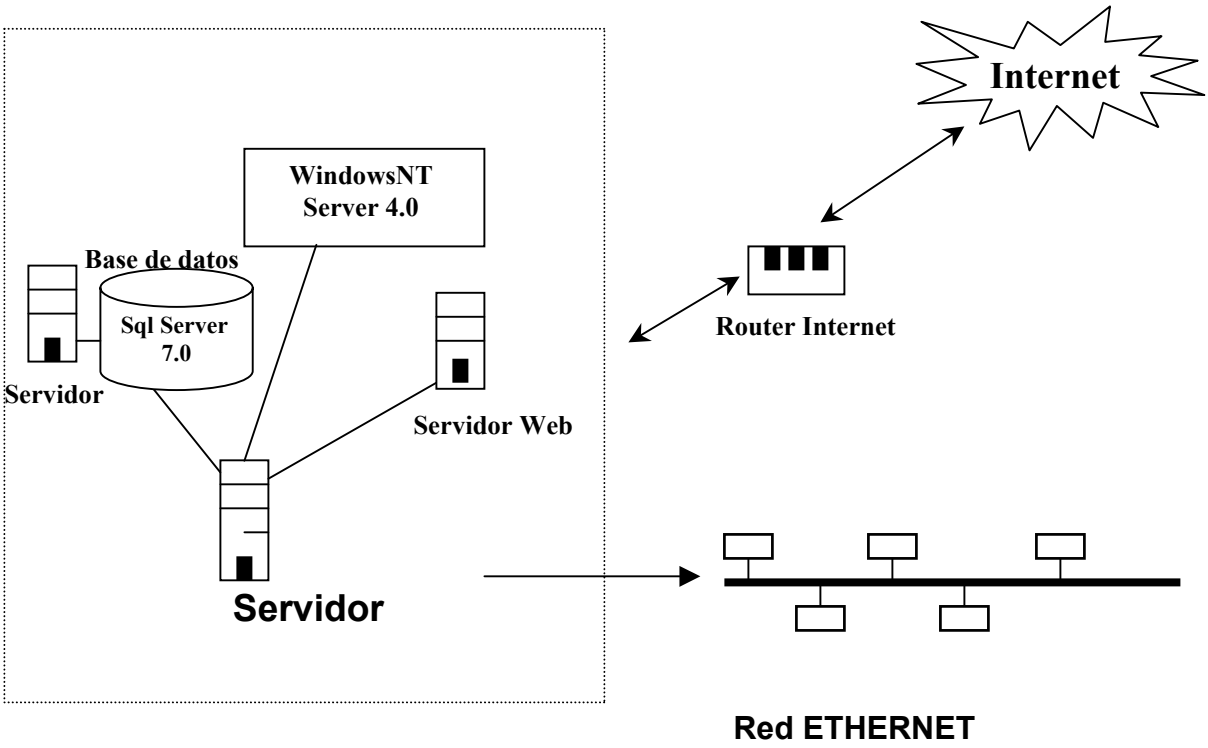
14.3 Implementación de la arquitectura final para el funcionamiento del sistema.

Con la arquitectura final, figura Nº 29, se puede dar funcionamiento al sistema. Dentro de este esquema se puede ver el Servidor de la red, el que está formado por los siguientes componentes:

1. Sistema operativo Windows NT Server 4.0.
2. Motor de base de datos SQL Server 7.0. Plataforma en la cual se puede crear una aplicación de base de datos para la web, a la vez se complementa de forma excelente con Windows NT.
3. Servidor web, Internet Information Server (IIS). Este tipo de servidor soporta comunicaciones en Internet.
4. Firewall. Software que otorga seguridad a datos entre la red privada y Internet.

Por otro lado, se puede apreciar un router el que sirve de enlace entre el servidor y Internet. Un router es un dispositivo capaz de conectar redes de distinto tipos. El router utilizado es de marca Cisco. Por último, se encuentra la red privada, la que está compuesta por un hub, tarjetas de red, el servidor descrito anteriormente, cable UTP nivel 5 y los distintos equipos que conforman la red, con sus respectivos conectores (RJ45 Y puntos de red).

Figura Nº 29. Arquitectura final que permite el funcionamiento del sistema.



.

15. Conversión y carga de los datos.

En esta etapa se procede a traspasar los datos que residen, en gran parte, sobre planillas Excel y los restantes en documentos archivados.

Este traspaso de datos se realiza incorporando uno a uno la información de estas planillas al nuevo sistema, al igual que cierta información relevante para la empresa que se encuentra almacenada en documentos físicos. Este tipo de conversión o carga de datos se realiza en forma programada, pues requiere de algún tiempo el tener absolutamente toda la información que se necesita para el nuevo sistema.

16. Pruebas.

En esta fase se procede a realizar algunas pruebas con el fin de detectar posibles errores en distintos ámbitos, con el propósito claro de corregirlos y mantener así un funcionamiento óptimo del sistema.

16.1 Prueba a procedimientos almacenados.

Las pruebas a procedimientos almacenados se realizan para ver el comportamiento que éstos tienen al momento en que se realizan llamadas a través del sistema.

Con el objeto de detectar errores se procedió a desplegar, insertar, eliminar y modificar datos, para así constatar que los procedimientos realicen las transacciones que se solicitan.

Por último, estas pruebas permiten asegurar que lo que se realiza a través de estos procedimientos sean los correctos.

16.2 Prueba a Triggers.

Los triggers o desencadenadores son de vital importancia para mantener la consistencia de la base de datos. Es por este motivo, que las pruebas que se realizan deben abarcar en su totalidad este tipo de procedimientos.

Las pruebas que se realizaron consistieron en desencadenar, a propósito, errores para que de esta forma se gatillarán los triggers, y así poder ver a ciencia cierta que estos realicen su función en forma correcta.

Algunos ejemplos de estas pruebas fueron:

1. Realizar eliminación en tablas padres: Cuando ocurre eliminación de registros en estas tablas, se debe eliminar toda aquella información que esté vinculada en forma directa a los registros que se borraron con el fin de mantener la consistencia de la base de datos.
2. Realizar inserción con datos ya existentes: En esta parte se procede a insertar datos repetidos con el objeto de provocar un error y hacer funcionar el trigger, para así constatar que estos realicen su función.
3. Realizar eliminaciones no posibles: Se procede aquí a eliminar aquellos registros que por su importancia y función no deben ser eliminados. Con esto se consigue tener certeza del cumplimiento del trigger, al no permitir dicha operación.

16.3 Prueba de seguridad.

Las pruebas de seguridad que se realizaron involucran lo siguiente:

- Verificar que el acceso a través de contraseñas funcionen.
- Verificar los niveles de seguridad para cada usuario.

- Constatar la seguridad de la base de datos.

16.3.1 Verificación del acceso a través de contraseñas.

Lo que se persigue en este punto es ver si efectivamente el acceso a través de contraseñas funciona en forma correcta. Para este fin, se probó el acceso, a través de las distintas contraseñas de los usuarios constatando su correcto funcionamiento.

Otro punto a considerar en esta parte, fue el haber tratado de entrar con contraseñas falsas y probar así, en forma real, la seguridad en el acceso al sistema.

16.3.2 Verificación de niveles de seguridad.

La verificación de niveles de seguridad es un paso importante, pues todos los usuarios no pueden tener acceso completo a la información de la empresa. En este proceso se comienza a revisar el acceso de cada usuario (empleados de la empresa). Se ingresa al sistema, ocupando las cuentas de los distintos empleados con el objeto de verificar que cada usuario, según su nivel de autorización, tenga la información que necesita dependiendo de su función dentro de la empresa.

16.3.3 Verificación seguridad base de datos.

En este punto, se debe señalar que la base de datos está bajo el sistema de seguridad que proporciona Windows NT, por lo que el ingreso a este lugar se realiza a través de sus políticas, como por ejemplo el ingreso de contraseña.

16.4 Prueba de aceptación visual.

Esta prueba se efectuó en conjunto con algunos empleados de la empresa para constatar sus opiniones y a la vez para comenzar a familiarizarlos con el sistema. Esta prueba finalmente, resulta exitosa, pues tubo una muy buena acogida entre los empleados y gerente de la empresa.

16.5 Prueba de rendimiento.

La prueba rendimiento se efectuó con el objeto de ver el comportamiento en cuanto a la respuesta del servidor web tras peticiones, capacidad de accesos y descarga de las páginas.

Esta prueba se realizó con la aplicación en demo del WebStress 4.4. Esta aplicación permite mostrar en forma gráfica y a través de resultados, el tiempo en que se demora en descargar páginas. Para este caso se efectuó la prueba través de la página principal del sitio web, figura Nº 30. Por otro lado, esta aplicación muestra la demora del servidor en crear y transmitir el código HTML, figura Nº 31.

Figura Nº 30. Descarga página principal sitio web.

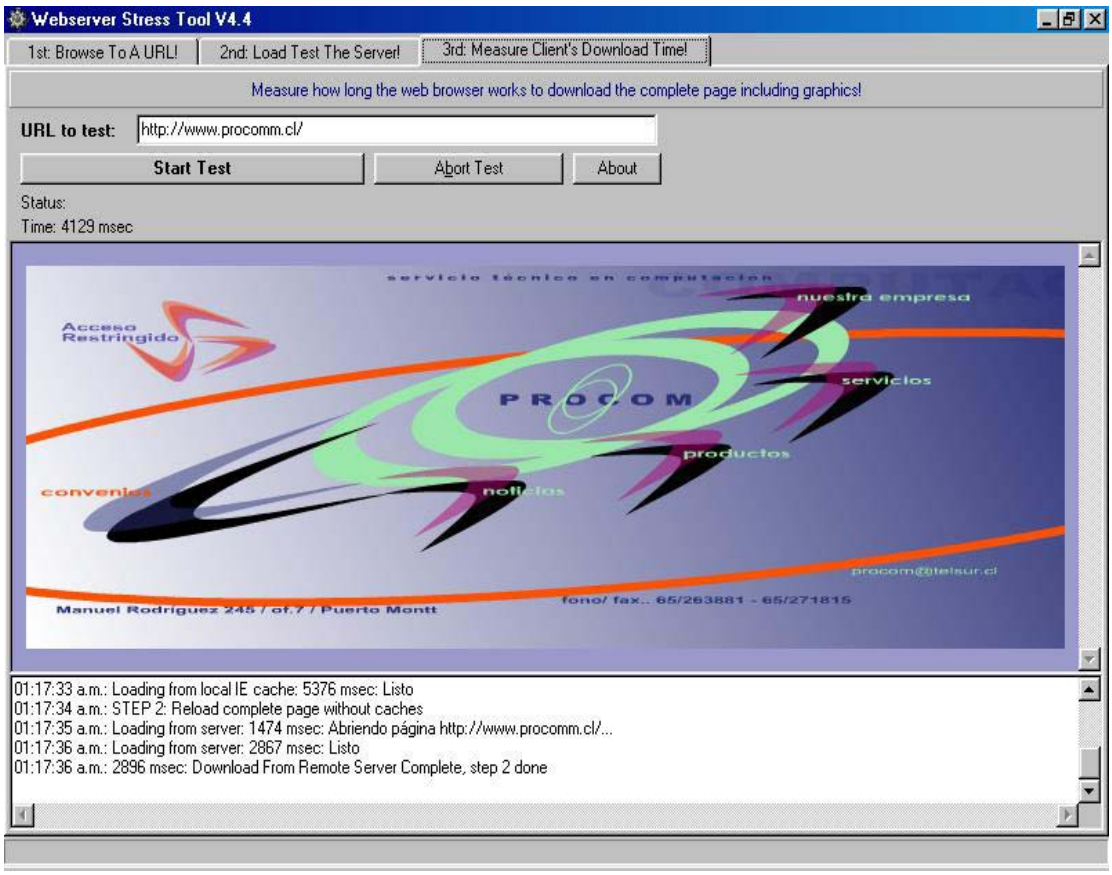
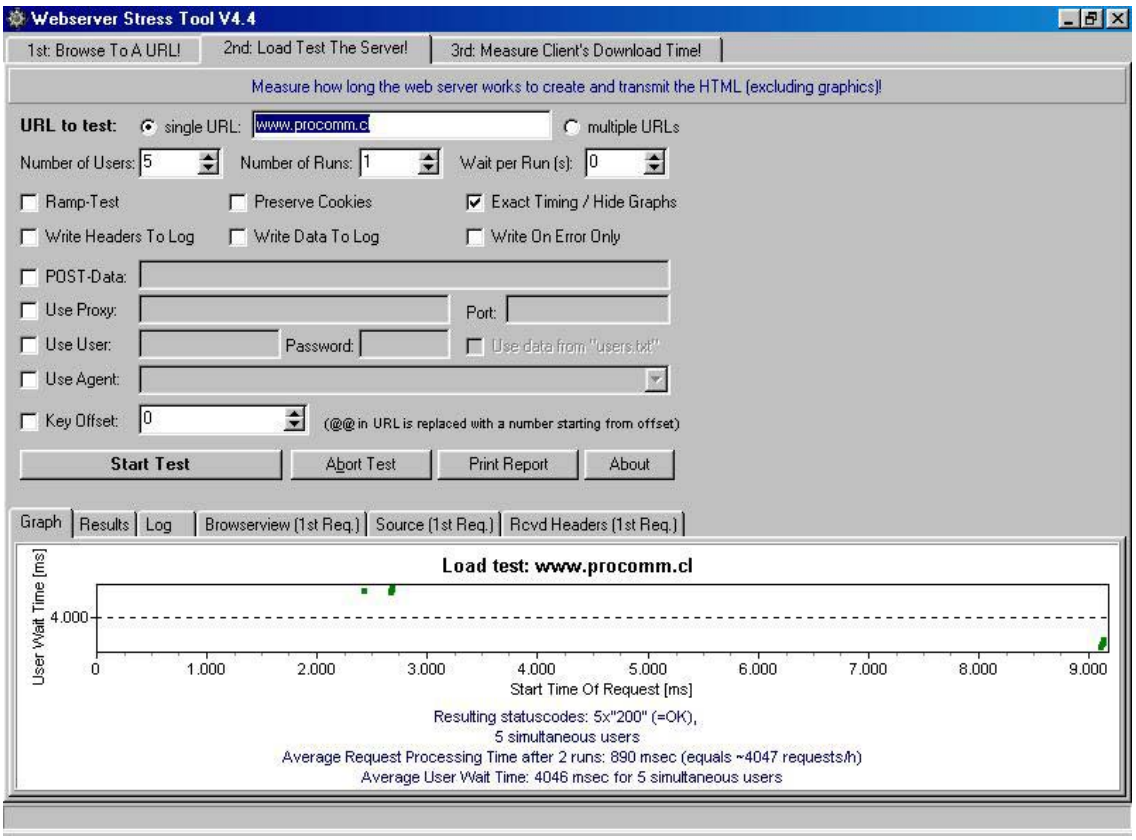


Figura Nº 31. Tiempo del servidor en crear y transmitir código HTML.



Los resultados de estas pruebas se muestran a continuación en forma detallada.

Test 1: www.procomm.cl
Server Load Test Results (by Webserver Stress Tool Version 4.4, http://www.paessler.com)
Tested URL: http://www.procomm.cl/
Settings: use GET requests
Settings: 5 simultaneous users
Client system: Windows 98 V4.10 (Build 67766222) , CPU Proc. Lev. 586 (Rev. 2060) at 0 MHz,

Client system: 9 MB available RAM of 125 MB total physical RAM, 445 MB available pagefile, 443 MB free disk space on C:
Test run on 28-08-02 10:07:12 p.m.

Results of run 1
Created 5 requests in 95,558ms (equals ~52 requests per second)
Time to first byte: ~2906 msec
Number of Bytes in Result: 3535 Bytes (1st request)
Resulting statuscodes: 5x"200" (=OK),
Average Request Processing Time after 1 runs: 1214 msec (equals ~2966 requests/h)
Average User Wait Time: 3638 msec for 5 simultaneous users

Test 2: www.procomm.cl

Server Load Test Results (by Webserver Stress Tool Version 4.4, http://www.paessler.com)
Tested URL: www.procomm.cl
Settings: use GET requests
Settings: 5 simultaneous users
Client system: Windows 98 V4.10 (Build 67766222) , CPU Proc. Lev. 586 (Rev. 2060) at 0 MHz,
Client system: 0 MB available RAM of 125 MB total physical RAM, 438 MB available pagefile, 437 MB free disk space on C:
Test run on 28-08-02 10:16:54 p.m.

Results of run 1
Created 5 requests in 263,68ms (equals ~19 requests per second)
Time to first byte: ~5249 msec
Number of Bytes in Result: 3535 Bytes (1st request)
Resulting statuscodes: 5x"200" (=OK),
Average Request Processing Time after 1 runs: 1199 msec (equals ~3004 requests/h)
Average User Wait Time: 5493 msec for 5 simultaneous users

Results of run 2
Created 5 requests in 22,624ms (equals ~221 requests per second)
Time to first byte: ~2183 msec
Resulting statuscodes: 5x"200" (=OK),
Average Request Processing Time after 2 runs: 890 msec (equals ~4047 requests/h)
Average User Wait Time: 4046 msec for 5 simultaneous users

Test 3 : www.telsur.cl

Server Load Test Results (by Webserver Stress Tool Version 4.4, http://www.paessler.com)
Tested URL: www.telsur.cl
Settings: use GET requests
Settings: 5 simultaneous users
Client system: Windows 98 V4.10 (Build 67766222) , CPU Proc. Lev. 586 (Rev. 2060) at 0 MHz,
Client system: 0 MB available RAM of 125 MB total physical RAM, 439 MB available pagefile, 434 MB free disk space on C:
Test run on 28-08-02 10:32:51 p.m.

Results of run 1
Created 5 requests in 97,274ms (equals ~51 requests per second)
Time to first byte: ~1857 msec
Number of Bytes in Result: 47373 Bytes (1st request)
Resulting statuscodes: 5x"200" (=OK),
Average Request Processing Time after 1 runs: 4703 msec (equals ~765 requests/h)
Average User Wait Time: 20102 msec for 5 simultaneous users

Traducción:

1. Average Request Processing Time: Tiempo promedio en que el servidor completa una petición.
2. Average User Wait Time: Tiempo promedio que el usuario debe esperar hasta que su petición es completada cuando hay X usuarios enviando peticiones simultáneamente.
3. Time to first byte: Tiempo medido entre el envío de la primera petición y la recepción del primer byte desde el servidor

17. Conclusión.

Dada la masificación del uso de tecnologías de información, es que cobra gran relevancia el haber realizado un sistema que cumpla tanto con satisfacer necesidades de la empresa, como la de los clientes con que ésta trabaja.

Es importante destacar que durante el proceso de elaboración de éste sistema, se tuvieron que considerar diversos elementos, como el estudio previo de una metodología para poder hacer frente en forma más, acabada y ordenada la construcción de un sitio web. También se considera valioso el hecho de haber aprendido a usar nuevas herramientas como Flash 5.0 y EditPlus para ejecutar así la realización de este sistema, pues en este ambiente web es importante el hecho de realizar páginas que contengan algunas animaciones, como también una gráfica adecuada a las necesidades de los clientes.

De esta manera, con el uso de tecnologías en el desarrollo web y después de haber estudiado los distintos requerimientos, se llegó a una solución informática, la que consiste en un sistema que sirva por un lado como una alternativa de comunicación entre clientes y empresa y a la vez de apoyo en la gestión que la empresa realiza día a día.

La realización de este sistema cumple con los diversos objetivos que se especificaron en esta tesis, quedando el cliente satisfecho con la realización del proyecto, el cuál fue desarrollado con una metodología eficiente.

Todo sistema que se desarrolla e implementa en sus inicios suelen contener errores típicos de programación, los que son fáciles de solucionar, por lo cuál es de suma importancia las pruebas, antes de que este sistema quede operativo en su totalidad, lo que ocurrirá dentro de un par de semanas.

Finalmente, no es menos importante mencionar como el poner a prueba la capacidad de responder ante distintas problemáticas que se manifiestan ante el desarrollo y implementación de un sistema, pues el manejo adecuado y también las soluciones adecuadas permiten en definitiva el tener o no éxito en la realización de un producto final, como es la entrega a la empresa de un sistema que satisfaga tanto a ellos como a sus clientes.

18. Bibliografía.

[Connolly1999] Connolly Thomas – Begg Carolyn – Strachan Anne. Data Base System. Addison – Wesley longman Limited. Segunda edición. 1999.

[Microsoft1998] Microsoft Corporation. Microsoft SQL Server Introduction. Microsoft Corporation. Primera edición.1998.

[Pressman1997] Pressman Roger. Ingeniería del software: Un enfoque práctico. Mcgraw – Hill. Tercera edición.1993.

[Mercovich2000] Mercovich, Eduardo. Estrategias para el desarrollo de un sitio en la web.
Disponible en
http://planeta.gaiasur.com.ar/seminarios/metodo-websites.html

[Rosenfeld1999] Arquitectura de la información para WWW. O'Ryan

[Senn1995] Senn James A. Análisis y Diseño de sistemas de información. McGraw-Hill Interamérica Ltda.. Segunda edición.1995.

[Byrne2000] Byrne, Jeffry.
Creación de sitios Web con SQL Server 7.
Prentice Hall, 2000

19. Anexos.

A. Documentos recolectados en la etapa de análisis y colección de requerimientos.

Estos documentos son:

- Cotización.
- Informe Técnico de trabajos.
- Estado de cuentas (ventas y trabajos).
- Lista de precios.
- Calendario de trabajos.

P R O C O M
Proyectos en Computación
M. Rodríguez 245 Of. 7 Pto. Montt

E-mail procom@telsur.cl

Fono/fax 56-65-263881

Puerto Montt, Agosto 02 del 2001

Señores
Eco Travels
Presente

At. Srta. Maria Eliana Tejeda.

Por medio de la presente y según lo conversado, me es muy grato someter a su consideración lo que ha continuación se detalla:

Impresora HP Deskjet 840 Color.....Valor $ 92.273 + IVA

Impresora Epson Stylus 777..... Valor $ 85.915 + IVA

Impresora Canon Bjc-2100..... Valor $ 71.910 + IVA

Los valores son más IVA
La forma de pago es Contado, documentado.
La cotización tiene validez por 5 días.

Sin otro particular le saluda atte.

Rodrigo Pinto Ortiz
Gerente Comercial

COT020801.doc

P R O C O M
Proyectos en Computación
M. Rodríguez 245 Of. 7 Pto. Montt.
E-mail procom@telsur.cl
Fono/fax 56-65-263881

Puerto Montt, Septiembre 08 del 2001

Señores
Procint Ltda.
Presente
At: Sr. Pablo Zúñiga

Remito a UD. el detalle de la revisión hecha a computador Celeron 800 Mhz :

- Problema: Cliente reporta problema en el encendido.
- Detalle: Se realizan las pruebas correspondientes y se determina mal funcionamiento en la fuente de poder, ante lo cual se procede a reemplazar la original, por una nueva. Luego de realizado esto se verifica que el equipo arranca correctamente.

Guía de despacho N° 5912.

Sin otro particular se despide atentamente,

Julio Reinoso Villarroel
Servicio Técnico
ITEC080901.doc

Estado de cuentas.

FACT.	FECHA	RAZON SOCIAL	DETALLE	NETO	I.V.A.	TOTAL	ESTADO
899	01-09-00	TELECTRONIC	ATENCIONES AGOSTO				CANCELADA
900	01-09-00	MORGAN Y FUENZALIDA	VARIOS				PENDIENTE
901	08-09-00	NOCEDAL	C.SERV - REC. SIST.				CANCELADA
902	11-09-00	CONST. SAN JUAN LTDA	CELERON 533 + IMPR.				CANCELADA
908	11-09-00	SOC.CLAUDIO SMITH	MANT. P´REV. EQ. (21)				CANCELADA
909	12-09-00	VICTOR WEITZLER	REINS. S.OPERATIVO				CANCELADA
903	14-09-00	CONST. SAN JUAN LTDA	DIF. DE PRECIO				CANCELADA
1000	21-09-00	INVERSIONES ASPEN S	MANT. IMPRESORA				PENDIENTE
904	23-09-00	COM. AQUA VIVA CHILE	M.MODEM INST.+CONF.				CANCELADA
905	23-09-00	NOCEDAL	ATENCION TERRENO				CANCELADA
906	27-09-00	BIOSUB LTDA.	CABLE SERIAL				CANCELADA
				$ 0	$ 0	$ 0	

Lista de precios

Código	IMPRESORAS CANON / EPSON / H.P.	PRECIO
1054	CANON BJC-1000, 720x360 DPI, 4PPM NEGRO/0,6PPM COLOR, PUERTO PARALELO	
1093	CANON BJC-2100, 720x360 DPI, 5PPM NEGRO/2PPM COLOR, PUERTOS USB	
1097	IMPRESORA EPSON STYLUS COLOR 480	
1098	IMPRESORA EPSON STYLUS COLOR 777	
1096	IMPRESORA H.P. DESKJET 840 COLOR	
1090	IMPRESORA H.P. LASERJET 1100	
Código	UPS, FILTROS DE LINEA, TRANSFORMADORES "TS SHARA"	PRECIO
1000	UPS 600VA 1 BATERIA PARA RESPALDO SOHO TS SHARA 3ENCHUFES,12VDA,C/PROT.	
1001	UPS 600VA 2 BATERIAS PARA RESPALDO TS SHARA 4ENCHUFES, 24VDA, C/PROT.	
1002	UPS 1000VA 2 BATERIAS PARA RESPALDO TS SHARA 4ENCHUFES, 24VDA, C/PROT.	
1003	UPS 1200VA 2 BATERIAS PARA RESPALDO TS SHARA 4ENCHUFES, 24VDA, C/PROT.	
1004	UPS 1500VA 2 BATERIAS PARA RESPALDO TS SHARA 4ENCHUFES, 24VDA, C/PROT.	
Código	NOTEBOOK	PRECIO
1006	NOTEBOOK SOYO PW-930S GRIS,Celeron 466/32MB/HDD 6GB/CD24x/FDD 1,44(EXT)/10"	
	Bolso y Drv (Win 98 Ing pre-inst, sin lic.)	
Código	SCANNER	PRECIO
1015	Scanner OPTIC PRO P12 3D 19200 DPI (600x1200), 48bit color, Paralelo, 220V.,	
1016	Scanner ACER S2W 3300u 19200 DPI (600x1200), 48bit color, USB	
1014	Scanner CANON 340P PARALELO	
1009	Scanner AGFA SNAPSCAN 1212 PARALELO	
Código	FUENTES DE PODER	PRECIO
0	FUENTE DE PODER AT 250W 110/220V OEM	
1	FUENTE DE PODER ATX 250W 110/220V OEM	
3	FUENTE DE PODER ATX 300W 110/220V 1.5A OEM	
Código	GABINETES	PRECIO
1002	AT Minitower Genérico 230W sin puerta (2 Bahias 5¼ y 2 Bahias 3½)	
1028	Gabinete ATX2303 (2x5 1/4" y 2X3 1/2"), Cable Nacional 250W/2,5A	
1016	ATX MidTower 250W (3 Bahias 5¼ y 2 Bahias 3½)	
Código	ACCESORIOS P/GABINETES, ADAPTADORES Y OTROS	PRECIO

Calendario de Trabajos

Enero	Febrero	Marzo	Abril	Mayo	Junio	Julio	Agosto	Septiembre	Octubre	Noviembre	Diciembre
								Mantencion Impresoras Dolly 17-sep-01		Mantencion Impresoras Intac 25-nov-01	
											Mantención Computads Weitzler 4-dic-01
								Mantencion Computads Intac 28-sep-01			