



**UNIVERSIDAD AUTÓNOMA DEL ESTADO
DE HIDALGO**

**INSTITUTO DE CIENCIAS BÁSICAS E
INGENIERÍA**

**“ACTUALIZACIÓN DEL SISTEMA DE CORRESPONDENCIA EN EL
DEPARTAMENTO DE INFRAESTRUCTURA EDUCATIVA DE LA
SECRETARÍA DE EDUCACIÓN PÚBLICA DEL ESTADO DE HIDALGO”**

M O N O G R A F Í A

**QUE PARA OBTENER EL TÍTULO DE
LICENCIADO EN SISTEMAS COMPUTACIONALES**

**P R E S E N T A
P.L.S.C. SILVIA JAZMÍN CRESPO HERNÁNDEZ**

**A S E S O R
L.S.C. J. JONATHAN MONROY CEDILLO**

**COASESOR
L.S.C. ALBERTO SUÁREZ NAVARRETE**

PACHUCA, HGO. ENERO DE 2009

Quiero agradecer a todos aquellos que hicieron que esto fuera posible:

A Papá y Mamá porque cada día con su ejemplo, apoyo, esfuerzo y gran cariño pude llegar a realizar cada una de mis metas y alcanzar este sueño.

A mis hermanos, porque siempre han estado conmigo en todo momento.

A ti Raúl por acompañarme, entenderme y apoyarme para que juntos logremos superarnos.

A ti hermoso bebé quiero que sepas que este esfuerzo también es tuyo por ser parte de mi vida y un motivo de impulso.

Los amo mucho.

A mis profesores, a quienes no solo les dedico esta página sino el eterno agradecimiento porque por ustedes soy mejor no solo en el ámbito profesional sino también en el sentido humano.

De nuevo, muchas gracias por ayudarme en este logro más.

Quiero agradecer a todos aquellos que esto fuera posible, empezando por mis padres, a quienes les agradezco por su infinito apoyo quiero que sepan que el objetivo logrado también es de ustedes

A mis padres por su apoyo incondicional y su infinito cariño

A mis hermanos

A ti Raùl por tu confianza

A ti hermoso bebè por ser fuente de inspiración

A todos muchas gracias por ayudarme en este logro más, el cual hubiera sido imposible sin su incondicional ayuda y comprensión

ÍNDICE

Justificación.....	I
Objetivo general.....	II
Objetivos específicos.....	II
Introducción.....	III

Capítulo 1 Marco Teórico y Antecedentes del Sistema

1.1 Tecnologías de Información y Comunicación (TIC).....	3
1.1.1 Elementos de las Tecnologías de Información.....	3
1.2 Sistemas de Información.....	4
1.2.1 Concepto de Sistema.....	4
1.2.2 Concepto de Información.....	5
1.2.3 Sistemas de Información.....	6
1.2.4 Actividades básicas de un Sistema de Información.....	6
1.2.5 Objetivos generales de los Sistemas de Información.....	7
1.2.6 Clasificación de los Sistemas de Información.....	7
1.2.7 Ciclo de vida de un Sistema de Información.....	10
1.2.8 Ciclo de desarrollo de los Sisemas de Información.....	11
1.2.9 Beneficios de los Sistemas de Información.....	16
1.3.Topología de Red.....	16
1.3.1 Concepto de Topología de Red.....	16
1.3.2 Clasificación de las Topologías de Red.....	17
1.3.2.1 Topología en Estrella.....	17
1.3.2.2 Topología en Bus.....	17
1.3.2.3 Topología en Anillo.....	18
1.3.2.4 Topología en Árbol.....	18
1.3.2.5 Topología en Malla.....	19
1.3.3 Diseño de la Red en la SEP.....	19
1.3.4 Estructura de la red en la SEP.....	20
1.4 Sistema de Correspondencia Anterior.....	23
1.4.1 Pantalla de Entrada.....	23
1.4.2 Pantalla de Salida.....	25

Capítulo 2 Analisis

2.1 Control de Oficios en el Departamento de Infraestructura Educativa de la SEP.....	29
2.2 Descripción del Control del Sistema de Correspondencia.....	31

2.3 Descripción de Pantallas	30
2.3.1 Pantalla de Entradas	32
2.3.2 Pantalla de Salidas.....	32
2.3.3 Pantalla de Consultas de Información Actual	32
2.3.4 Pantalla de Consultas de Información Anterior	32
2.3.5 Pantalla de Consultas de Informes x CCT.....	32

Capítulo 3 Diseño y Construcción

3.1 Programación	35
3.1.1 Concepto de Programación.....	35
3.1.2 Visual Studio .NET 2003 (VB NET)	36
3.1.2.1 Creación de un Proyecto en Visual Studio .NET 2003	37
3.2 Microsoft SQL Server 2000	40
3.2.1 Creación de una Base de Datos.....	41
3.2.2 Diseño de la Base de Datos "Corresp_Plan".....	43
3.2.2.1 Creación de Tablas	43
3.2.2.2 Diagrama de la BD del Sistema de Correspondencia	50
3.2.2.3 Creación de Vistas	51
3.2.3 Diseño de Pantallas	59
3.2.3.1 Pantalla de Acceso al Sistema de Correspondencia.....	59
3.2.3.2 Pantalla principal	59
3.2.3.3 Pantalla de Entradas	60
3.2.3.4 Pantalla de Salidas.....	61
3.2.3.5 Pantalla de Consultas de Información Actual	62
3.2.3.6 Pantalla de Consultas de Información Anterior	63
3.2.3.7 Pantalla de Consultas de Informes x CCT.....	64

Capítulo 4 Sistema de Correspondencia

4.1 Sistema de Correspondencia Actual	69
4.1.1 Pantalla de Entradas	74
4.1.2 Pantalla de Salidas.....	81
4.1.3 Pantalla de Información Actual.....	86
4.1.4 Pantalla de Información Anterior	94
4.1.5 Pantalla de Consultas de Informes x CCT.....	100

Conclusiones	109
---------------------------	-----

Referencias Bibliográficas	111
---	-----

Referencias Electrónicas	111
---------------------------------------	-----

Glosario de Términos	113
-----------------------------------	-----

ÍNDICE DE FIGURAS

Capítulo 1 Marco Teórico y Antecedentes del Sistema

1.1	Actividades de un Sistema de Información.....	7
1.2	Ciclo de vida de un Sistema de Información	11
1.3	Modelo Lineal Secuencial.....	12
1.4	Modelo de Construcción de Prototipos.....	12
1.5	Modelo de Desarrollo Rápido de Aplicaciones.....	13
1.6	Modelo Incremental.....	14
1.7	Modelo Espiral.....	14
1.8	Modelo de Ensamblaje de Componentes.....	15
1.9	Topología de Estrella	17
1.10	Topología de Bus	17
1.11	Topología de Anillo.....	18
1.12	Topología en Árbol.....	19
1.13	Topología en Malla.....	19
1.14	Estructura de la Red en la SEP	21
1.15	Ubicación del Departamento de Infraestructura de la SEP	21
1.16	Ubicación del Equipo de Desarrollo del Sistema de Correspondencia.....	22
1.17	Pantalla de Entrada.....	23
1.18	Pantalla de Salida	25
1.19	Opciones de Búsqueda	26

Capítulo 3 Diseño y Construcción

3.1	Pantalla de Bienvenida de Visual Studio .NET.....	37
3.2	Creación de una Solución	37
3.3	Pantalla de Solución.....	38
3.4	Explorador de Soluciones.....	38
3.5	Creación de un Proyecto.....	38
3.6	Ubicación y Nombre del Proyecto	39
3.7	Pantalla para empezar a programar.....	39
3.8	Administrador Cooperativo.....	41
3.9	Propiedades de la Base de Datos	41
3.10	Archivo de Datos	42
3.11	Registro de Transacciones.....	42
3.12	Catálogo "cat_clasificador"	43
3.13	Catálogo "cat_escuelas".....	43
3.14	Catálogo "cat_edos"	44
3.15	Catálogo "cat_identificador"	44
3.16	Catálogo "cat_localidades".....	44

3.17 Catálogo "cat_mun"	45
3.18 Catálogo "cat_stat"	45
3.19 Catálogo "cat_tproc"	45
3.20 Catálogo "cat_tsolic"	46
3.21 Catálogo "cat_turno"	46
3.22 Catálogo "cat_usuarios "	46
3.23 Catálogo "det_entradas"	47
3.24 Catálogo "Entradas"	47
3.25 Catálogo "Entradas_historico"	48
3.26 Catálogo "Salidas"	48
3.27 Catálogo "Salidas_historico"	49
3.28 Diagrama del Sistema de Correspondencia	50
3.29 Vista "vusqueda"	52
3.30 Vista "vdet_entradas"	53
3.31 Vista "ventradas"	54
3.32 Vista "ventradas_historico"	55
3.33 Vista "ventradas_sindet"	56
3.34 Vista "vescuelas"	57
3.35 Vista "vmpios_cat_esc"	58
3.36 Vista "vsalidas"	58
3.37 Pantala de Acceso al Sistema de Correspondencia.....	59
3.38 Pantalla Principal del Sistema de Correspondencia.....	60
3.39 Pantalla de Entradas	61
3.40 Pantalla de Salidas.....	62
3.41 Pantalla de Consultas de Información Actual	63
3.42 Pantalla de Consultas de Información Anterior.....	64
3.43 Pantalla de Consultas de Informes x CCT.....	65

Capítulo 4 Sistema de Correspondencia

4.1 Pantala de Acceso al Sistema de Correspondencia.....	69
4.2 Pantalla Principal del Sistema de Correspondencia.....	71
4.3 Menú de Mantenimiento	72
4.4 Menú Consultas	73
4.5 Pantalla de Entradas	74
4.6 Botón Guardar	75
4.7 Selección de la fecha a través de un calendario	76
4.8 Botón Leer Folios	77
4.9 Datos del oficio desde un grid	77
4.10 Anexado de Escuelas.....	78
4.11 Grid con la información del CCT	80
4.12 Búsqueda de CCT	80
4.13 Pantalla Principal de Salidas	81
4.14 Resultado de la búsqueda.....	83
4.15 Aparición del número de oficio	84
4.16 Pantalla de Información Actual.....	86

4.17 Grid con la información de los criterios de búsqueda.....	90
4.18 Pantalla de Consulta de Entradas con información seleccionada de un grid .	90
4.19 Informe de Consultas de Salidas.....	91
4.20 Exportación del Oficio.....	92
4.21 Cuadro de diálogos para exportar	92
4.22 Oficio exportado en formato PDF	93
4.23 Pantalla de Información Anterior	94
4.24 Grid en la Pantalla de Información Anterior.....	97
4.25 Despliegue de la información	97
4.26 Diseño del oficio de la Consulta de la Información Actual.....	98
4.27 Exportación de Oficios de la pantalla de Información Actual.....	99
4.28 Cuadro de diálogos para exportar un oficio	99
4.29 Exportación del Oficio de Consulta Anterior en PDF	100
4.30 Pantalla de Consultas de Informes por CCT	100
4.31 Información del CCT y la aparición del botón imprimir	101
4.32 Formato del Reporte para las Consultas por CCT	102
4.33 Como exportar el documento de Consultas por CCT.....	102
4.34 Cuadro de descargas de archivos de Consultas por CCT.....	103
4.35 Exportación del Oficio por Consultas del CCT en PDF	103
4.36 Combos de Selección para obtener una clave de CCT.....	104
4.37 Como buscar un centro de trabajo	105
4.38 Resultado de la búsqueda de los CCT	106
4.39 Menú Salir	107

JUSTIFICACIÓN

El desarrollo del presente trabajo, se encuentra respaldado por la necesidad que vive el Departamento de Infraestructura Educativa de la Secretaría de Educación Pública del Estado de Hidalgo para satisfacer el control administrativo de la información manejada en dicho departamento. Donde se encargan de dar solución a las distintas solicitudes de construcción, equipamiento, mobiliario, etcétera, que envían las diferentes dependencias de la SEP.

En 1998 se crearon unos formularios en dBase para llevar un mejor control de la información, los cuales se entregaron al departamento de Infraestructura Educativa, sin código y con la finalidad de que no se pudiesen realizar cambios sin consultar y dar una paga al propietario. El sistema de correspondencia, constaba de 2 simples formularios, el de Entradas y Salidas.

En el formulario de Entradas se capturaba la solicitud con todos los datos que traía el oficio, y en el de Salidas, solo se daba captura de las respuestas a dichas solicitudes.

Dichos formularios no eran precisos, exactos, íntegros, significativos, ni oportunos. Los cuales no eran de gran ayuda, ya que mucha de la información capturada no se guardaba completa y a la hora de hacer consultas dicha *información* no se visualizaba, lo que les creó problemas, ya que mucha de esa *información* era relevante para realizar los oficios requeridos.

Por lo mencionado anteriormente, y para que el personal administrativo cuente con la información necesaria para darle solución a las diferentes solicitudes enviadas al Departamento de Infraestructura Educativa de la SEP, es necesario centralizar la *información*, y se hará mediante tablas en *SQL Server*. Además tendrá una interfaz realizada en *Visual .NET 2003*, la cuál ayudará a que la *información* sea precisa, exacta, íntegra, oportuna y significativa.

La interfaz del sistema de correspondencia actual será amigable y le dará al usuario la posibilidad de capturar el oficio de forma rápida y sencilla, al igual que la captura de respuesta mediante el servicio de páginas web. Además ayudará a visualizar la información necesaria, requerida en el momento que el usuario la desee, esto a través de consultas programadas, teniendo la como opción exportarlas a *Word*, *Excel*, o *PDF*. Permitiendo agilización en las operaciones de las solicitudes que se manejan dentro del departamento.

Para ingresar al sistema de correspondencia, existirá una pantalla de acceso, la cuál pedirá un usuario y contraseña, lo que permitirá que en el manejo de la información sea el debido, ya que si se le envía un oficio a algún usuario, este tendrá acceso solo a los oficios que le son enviados para darles la debida respuesta, sin ver ni tener acceso a la información de los demás usuarios. Mas sin embargo, el emisor (persona que envía la información a los demás usuarios) podrá ver y tener acceso a toda la información manejada en la base de datos a través del sistema.

OBJETIVOS

OBJETIVO GENERAL

Desarrollar un Sistema de Información que optimice el proceso de control de registro de las solicitudes que se envían al Departamento de Infraestructura Educativa de la SEP.

OBJETIVOS ESPECÍFICOS

- Implementar una interfaz amigable realizada en *visual .NET 2003*
- Diseñar y desarrollar una base de datos en *SQL Server* para el almacenamiento de la información de la aplicación.
- Implementar en la red actual el sistema de correspondencia.

INTRODUCCIÓN

Las *Tecnologías de Información (TI)* son herramientas y métodos aplicados para crear, almacenar, procesar y distribuir *información* generada, requerida automáticamente. Dentro de los elementos pertenecientes a las *TI* se encuentran la Informática, el hardware, software, las bases de datos, telecomunicaciones, redes, intranets, extranets y los sistemas de información.

Un *Sistema de Información* es un conjunto de elementos (hardware, software, recurso humano, telecomunicaciones y reglas de operación) que interactúan entre sí con el fin de apoyar las actividades de una empresa, organización o negocio.

Dichos sistemas han logrado importantes cambios logrando ventajas competitivas, pues automatizan los procesos operativos, suministran la información necesaria por lo desarrollan una ventaja estratégica para los usuarios.

Para el funcionamiento de los sistemas de información, se necesitan las redes, que son una manera de conectar equipos (computadoras y/o *dispositivos*) entre sí, compartiendo recursos, *información*, y servicios a través de cables, señales o cualquier transporte de datos.

Un *SI* se genera a través de la programación, que es un conjunto de instrucciones escrito en determinado lenguaje de programación esperando obtener el resultado deseado.

Para el desarrollo del sistema que se describirá mas adelante, se utilizará Microsoft Visual Studio .NET y la BD se desarrollará con SQL Server 2000.

El desarrollo del presente trabajo consta de 4 capítulos, los que se describen a continuación:

En el capítulo 1 se hablará a grandes rasgos de lo que son las *Tecnologías de Información* y el cómo están construidas; de los *Sistemas de Información*, sus actividades básicas, objetivos y su clasificación, así como se hablará de su ciclo de vida y metodologías, así como los beneficios que otorgan.

Se hablará también de las *Topologías de Redes*, y el cómo están construidas, asimismo una breve descripción de la estructura de red dentro de la Secretaría de Educación Pública de Hidalgo.

En el capítulo 2 se llevará a cabo el análisis de requerimientos del Departamento de Infraestructura Educativa de la SEP, comentando cómo utilizaba la información el Sistema de Correspondencia anterior, y cómo se evaluarán funciones, rendimientos y restricciones del Sistema de Correspondencia Actual.

En el capítulo 3 se describirá lo que es la programación y las aplicaciones en las que se desarrolló el sistema y la BD (Microsoft Visual Studio .NET 2003 y Microsoft SQL Server 2000). Además se hablará acerca del diseño de cómo se realizó el sistema de Correspondencia Actual y el diseño para la Base de Datos, con sus respectivas tablas y vistas.

En el capítulo 4 se muestra lo que era el Sistema de Correspondencia anterior en dBase y se hablará también de cómo fue desarrollado el Sistema de Correspondencia Actual. Mostrando las pantallas que forman el sistema, así como partes del código que lo generan.

Además se presentan las conclusiones, referencias bibliográficas y electrónicas, así como el glosario.



Capítulo 1. Marco Teórico y Antecedentes del Sistema

Este capítulo contiene las referencias de lo que son las *Tecnologías de Información, Sistemas de Información* y cómo están construidas, asimismo una breve descripción de lo que son las *Topologías de Redes*, su clasificación, además de la estructura de red dentro de la Secretaría de Educación Pública de Hidalgo. La *Información* servirá de base para el desarrollo del sistema del presente trabajo.

1.1 TECNOLOGÍAS DE INFORMACIÓN Y COMUNICACIÓN (TIC)

En la actualidad, las *Tecnologías de Información (TI)* están cambiando la forma tradicional de hacer las cosas para las personas que trabajan en empresas privadas, de gobierno o como profesional en cualquier campo; ya que éstas hacen posible capturar y utilizar la información en el momento que se genera.

Las *TI* son aquellas herramientas y métodos aplicados para crear, almacenar, procesar y distribuir *información* generada, requerida automáticamente; y se encuentra asociada con las computadoras y las tecnologías. [1]

Dentro de esta definición, encontramos los siguientes elementos que son parte de las *TI*:

- Informática.
- Hardware.
- Software.
- Bases de datos.
- Telecomunicaciones.
- Redes.
- Intranets.
- Extranets.
- Sistemas de información.

1.1.1 ELEMENTOS DE LAS TECNOLOGÍAS DE INFORMACIÓN

La *informática* es una disciplina que estudia el procesamiento de la *información* automática mediante dispositivos y sistemas. El concepto de informática se forma de las palabras Información y Automática. [5]

Un elemento muy importante es el *Hardware*, ya que es el equipo utilizado para el funcionamiento de una computadora, y se encargan de llevar a cabo las actividades de entrada, procesamiento y salida. Los dispositivos de entrada son los encargados de enviar la información al CPU en código binario, tales como: Teclado, Mouse, Escáner, Micrófono, Cámara digital, etcétera. Los dispositivos de procesamiento son los encargados de realizar operaciones lógicas, matemáticas y de control; los principales dispositivos de procesamiento son: La Unidad Central de Procesamiento "CPU" y la memoria principal. Los dispositivos de salida son

aquellos que reciben información que es procesada por la CPU para hacerla perceptible para el usuario, entre ellos están: Monitor, Impresoras, Fax, etcétera.

El *Software* es el conjunto de programas, documentos, procedimientos y rutinas que el ordenador emplea para dirigir las operaciones de una computadora: comúnmente llamados “programas de computación”, y estos se encargan que el programa o sistema cumpla con sus objetivos y opere con eficiencia. El *software* se clasifica en 2; de sistema y de aplicación. El *software* de *sistema*, es el sistema operativo, es decir, el conjunto de programas que facilita el uso del computador, administra y asigna los recursos del sistema. El *software* de aplicación son programas que controlan y optimizan la operación de la máquina, estableciendo de forma amigable la relación básica y fundamental entre usuario y computador.

En la actualidad, los sistemas informáticos manejan grandes cantidades de *información*, y para ello, utilizan las *Bases de Datos* que son un conjunto de datos pertenecientes a un mismo contexto y almacenados posteriormente para su uso [1], a la que se podrá realizar operaciones de consulta, modificación, inserción y eliminación.

Para la transmisión de esa *información* de un punto a otro se utilizan las *Telecomunicaciones* (del prefijo griego tele, “distancia” o “lejos”, “comunicación a distancia”) el cuál cubre todas las formas de comunicación a distancia de manera electrónica, y para que esto funcione se utilizan las *Redes*, que son una manera de conectar equipos (computadoras y/o *dispositivos*) entre sí, compartiendo recursos, *información*, y servicios ya sea de *Internet*, Chat, juegos, etc. a través de cables, señales o cualquier transporte de datos. El *Internet* consiste en una interconexión mundial de las *redes*, ya sean gubernamentales, académicas, públicas y privadas. Con esta tecnología (protocolos TCP/IP para *Internet*) se basan las *Intranets* que son una *red* privada empresarial o educativa donde se puede intercambiar *información* y trabajar en proyectos comunes. Y para que las organizaciones puedan enlazar recursos seleccionados de su *Intranet* con sus clientes, proveedores o socios se utilizan las *Extranets*.

1.2. SISTEMAS DE INFORMACIÓN

1.2.1. CONCEPTO DE SISTEMA

Conjunto de partes o elementos dinámicamente organizados y relacionados entre sí, que se afectan mutuamente formando una actividad para alcanzar un objetivo operando sobre datos para proveer *información*. Los sistemas se clasifican en: [6]

- *Simple*: Integrado con pocos elementos y su relación es sencilla.
- *Complejo*: Integrado con muchos elementos estrechamente relacionados.
- *Abiertos*: Interactúan con el ambiente continuamente intercambiando materia y energía.
- *Cerrados*: Son herméticos a cualquier influencia ambiental.
- *Temporales*: Aquellos q desaparecen en corto tiempo.
- *Permanentes*: Diseñados para funcionar en un periodo largo.
- *Estables*: Sus propiedades y operaciones que realizan no cambian.
- *No estables*: Sus propiedades y operaciones cambian constantemente ajustándose al tiempo y recursos.
- *Adaptativos*: Aquellos que cambian de acuerdo al entorno para mejorar su funcionamiento, logro y supervivencia.
- *No adaptativos*: Aquellos que no son capaces de modificarse y tienen problemas con su entorno, pueden ser eliminados o fracasar.
- *Determinísticos*: Interactúan en forma predecible.
- *Probabilísticos*: interactúan de forma incierta.

1.2.2. CONCEPTO DE INFORMACIÓN

Es un conjunto de datos organizados, significativos y pertinentes que describen sucesos o entidades. La *información* para que sea una fuente segura debe cumplir con las siguientes características:

- *Exacta*: aquella con resultados precisos
- *Objetiva*: aquella que identifica la finalidad hacia la cual se deben dirigir para llegar a la misión.
- *Completa*: contiene todos aquellos datos importantes.
- *Simple*: extraer la *información* importante, evitando la que no sea útil.
- *Flexible*: la *información* obtenida debe de ser útil para varios propósitos.
- *Oportuna*: la *información* se debe generar cuando se requiera y notificar por la persona adecuada a la par con los acontecimientos, de tal manera, que permita la toma de decisión y la actuación inmediata.
- *Confiable*: aquella que garantiza la calidad de los datos, así como de las fuentes de *información*.
- *Segura*: la *información* debe de estar restringida y permitir el acceso solo a usuarios autorizados.
- *Pertinente*: aquella *información* que sea realmente importante para la toma de decisión adecuada.
- *Verificable*: que se pueda comprobar comprobando con *información* similar que es la correcta.

1.2.3. SISTEMAS DE INFORMACIÓN

Los *Sistemas de Información (SI)* son un conjunto de elementos que interactúan entre sí con el fin de apoyar a una organización o empresa. Dichos *sistemas* han cambiado la forma en que operan las organizaciones actuales, ya que logran importantes mejoras, automatizando procesos y suministran la *información* para la toma de decisiones logrando ventajas competitivas.

Los elementos que intervienen en un *Sistema de Información* son: el equipo computacional (*hardware* necesario para que el *SI* opere), el recurso humano (aquella persona que interactúa con el *SI*), la *información* (los datos fuente), los programas (*software*), *telecomunicaciones*, los procedimientos y reglas de operación.

1.2.4. ACTIVIDADES BÁSICAS DE UN SISTEMA DE INFORMACIÓN

Los *SI* realizan 4 actividades básicas:

- *Entrada de información:* en esta actividad solo se recolectan los datos exactos para obtener la salida deseada; ya sea manual o automáticamente. Cuando el usuario proporciona de forma directa la información es manual, y los que provienen de otros sistemas o módulos se les denomina automáticos.
Las terminales o unidades de entrada de datos a la computadora son los diskettes, teclado, mouse, escáner, códigos de barra, etcétera.
- *Almacenamiento de información:* a través de esta actividad el sistema puede recordar información guardada (archivos) en la sesión o proceso anterior, ya sea en el disco duro, diskettes, o discos compactos (CD-ROM).
- *Procesamiento de información:* en esta actividad se realiza la ejecución de operaciones de cálculo, comparación, filtrado, etc. a través de una secuencia de operaciones establecida. La cuál permite la transformación de los datos fuente a *información* útil para la toma de decisiones.
- *Salida de información:* es la capacidad de un *SI* para obtener la *información* procesada al exterior, mediante salidas típicas como los la impresora, diskettes, cintas magnéticas, graficadores, plotters, etc.
- *Retroalimentación de la información:* se utiliza para realizar cambios o ajustes, y detectar los errores en la captura de datos.

En la figura 1 se esquematizan las actividades que realiza un *Sistema de Información*.

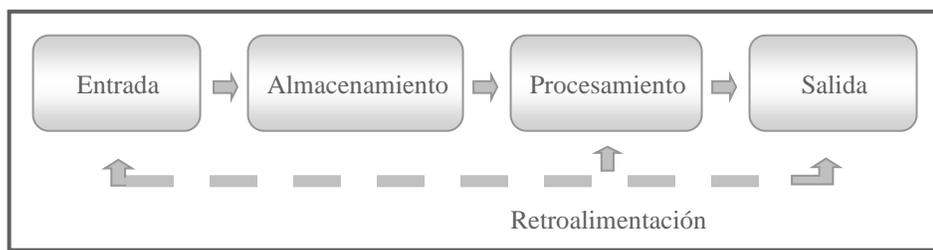


Figura 1.1. Actividades de un sistema de información.

1.2.5. OBJETIVOS GENERALES DE LOS SISTEMAS DE INFORMACIÓN

Los *SI* son creados para facilitar las actividades en una organización. Algunos de los objetivos que cumplen, son:

- Automatizar los procesos operativos.
- Proporcionar datos oportunos, exactos y confiables a los encargados de la toma de decisiones.
- Asegurar que la *información* este disponible cuando se necesite.
- Proporcionar la *información* de una manera fácil de entender.
- Incrementar la productividad operacional a través de su uso.

1.2.6. CLASIFICACIÓN DE LOS SISTEMAS DE INFORMACIÓN

Según las necesidades del negocio, los *Sistemas de Información* pueden clasificarse en:

a. *Sistema de Procesamiento de Transacciones (TPS).*

Son *sistemas* desarrollados para procesar transacciones rutinarias en las empresas o negocios de manera eficiente; reducen tiempos y/o errores, eliminando el tedio de realizar las operaciones manuales rutinarias, tales como: la facturación de ventas, entrega de mercancía, comprobación de las cuentas (a cobrar y/o a pagar), etcétera.

Dichos procesos se llevan a cabo mediante distintas actividades: cálculos analíticos, clasificación y ordenamiento (según importancia), almacenamiento y recuperación de *información*, generación de resúmenes (para comprobar rápidamente el estado de la empresa). Estas actividades tienen características en común: gran volumen de transacciones repetitivas, similitud entre dichas transacciones, los procedimientos al procesar las transacciones son bien definidos y detallados.

b. *Sistema de Automatización de Oficina (OAS)*

Sistemas destinados a ayudar al trabajo diario del administrativo (secretaria, archivistas, etc.) de una organización, quienes utilizan la *información* para analizarla, transformarla o manejarla de alguna forma y compartirla o diseminarla por toda la organización. [4]

Los procesadores de textos, hojas de cálculo, correo electrónico, etcétera, forman parte de estos *sistemas*. El *software* más conocido que se puede anexar a este tipo de sistemas es el *Microsoft Office* en cualquier versión.

c. *Sistema de Trabajo de Conocimiento (KWS)*

Sistemas de manejo de conocimientos ayudan a crear nuevos conocimientos que contribuyan con la organización o sociedad a trabajadores profesionales, tales como científicos, doctores, ingenieros, etcétera.

d. *Sistemas de Soporte para la toma de Decisiones (SSD)*

Estos sistemas trabajan a nivel gerencial. Proporciona información a los directivos que deben tomar dediciones particulares (semiestructuradas, únicas o que cambian rápidamente), que combinan datos y modelos analíticos sofisticados; tienden a ser mas interactivos que otros sistemas.

Ofrecen a los usuarios flexibilidad, adaptabilidad, y rápida respuesta, dando soporte a decisiones y problemas cuyas soluciones no se pueden especificar de antemano. [4]

e. *Sistemas de Información Administrativa (MIS)*

Proporcionan a los directivos y gerentes informes y acceso a los registros ordinarios e históricos de la institución proporcionando *información* necesaria, la cuál ayuda a tomar decisiones y resolver problemas estructurados. La *información*

sirve principalmente a las funciones de planeación, control y toma de decisiones a nivel de administración gerencial.

Las decisiones se denominan estructuradas, ya que los administradores conocen de antemano los factores a tener en cuenta (previamente establecida con la *información* proveniente de los TPS de forma rápida, precisa y fiable.

f. *Sistema de Apoyo a Ejecutivos (ESS)*

Es un *Sistema de Información* computarizado utilizado en la parte más alta de la empresa, proporciona *información* interna y externa que sea de relevancia para el ejecutivo, la cual se utiliza como apoyo; ya que es capaz de acceder y gestionar *información* en distintas fuentes y formatos en forma significativa y manejable para el usuario, maneja *información* estructurada y no estructurada, tanto cualitativa como cuantitativa. La cuál ayuda a la toma de decisiones a nivel estratégico. [7]

Los ejecutivos pueden utilizar los *ESS* de 2 formas:

- *Solo lectura*: cuando los ejecutivos tienen acceso a los últimos datos o informes para examinar la *información*, pero no pueden procesar datos.
- *Análisis Personalizado*: cuando los ejecutivos tienen acceso a la *información* y los permisos que permitan considerar, cambiar, extender y operar datos según los procedimientos que el ejecutivo necesite llevar a cabo con dicha *información*.

g. *Sistemas Expertos (SE, Expert System)*

Un Sistema Experto o también llamado un Sistema Basado en Conocimiento, es un *software* que imita la capacidad de razonamiento del ser humano para la solución de un problema particular experimentado, arrojando la mejor solución de una forma rápida; ya que almacenan amplia *información* específica de un dominio concreto, cuyo uso es gracias a la amplia disponibilidad de *hardware* y *software* de hoy en día.

Se podría decir que la Inteligencia Artificial (AL) puede ser considerada la meta de los *sistemas expertos*.

Arquitectura de los Sistemas Expertos:

- *Base de conocimiento*: parte del sistema que contiene los conocimientos y experiencias del experto en la materia.
- *Base de hechos (Memoria de Trabajo)*: al hacer una *consulta* de un problema actual, se descubren hechos, el SE comparan dichos hechos con el conocimiento de la base de conocimientos para deducir nuevos hechos.
- *Motor de inferencia*: modela el proceso de razonamiento humano con el motor de inferencia, el cuál trabaja con la *información* de la base de conocimientos y la base de hechos para deducir las conclusiones del problema.
- *Subsistema de explicación*: en este modulo se percibe la habilidad del sistema para explicar su razonamiento, ya que puede proporcionar al usuario una explicación del por qué hace una pregunta y como es que ha llegado a una conclusión.
- *Interfaz de usuario*: la interacción entre el sistema y el usuario es de forma natural (siguiendo el patrón de una conversación), para ello es necesario saber formular las preguntas, y/o diseñar la interfaz utilizando gráficas o menús para obtener *información* más fiable.

1.2.7. CICLO DE VIDA DE UN SISTEMA DE INFORMACIÓN

El ciclo de vida de un *Sistema de Información* comprende 5 fases: nacimiento, desarrollo, operación, mantenimiento y muerte (Figura 1.1).

1. *Nacimiento*: esta fase se inicia cuando al usuario le surge una necesidad. Se realiza un estudio de factibilidad para decidir si en verdad se requiere el desarrollo del sistema.
2. *Desarrollo*: se analizan los requerimientos, creando un diseño que servirá de base para el desarrollo del sistema, programando lo necesario para que el sistema funcione. Aquí se diseña, construye el sistema y/o adecua.
3. *Operación*: en este paso, el sistema ya esta terminado, y es donde el usuario empieza a introducir datos, obteniendo *reportes* de la *información* necesaria.
4. *Mantenimiento*: aquí se corrigen los errores o funciones detectados en los programas y/o agregar nuevos requerimientos.
5. *Muerte*: el sistema llega a esta fase cuando deja de ser útil al *usuario* o empresa y se necesita reemplazar por uno mejor (haciendo mejoras o

cambios que sobrepasen el mantenimiento), iniciando nuevamente con el ciclo de vida de un sistema ya que se desarrollara uno nuevo.

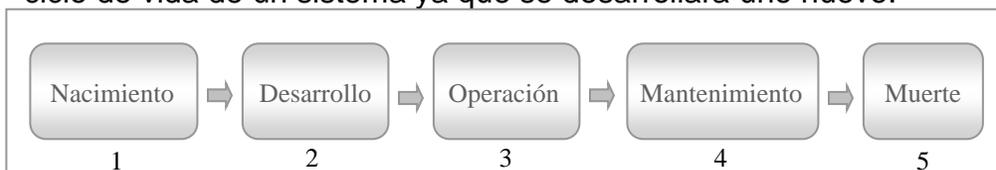


Figura 1.2. Ciclo de vida de un sistema de información.

1.2.8. CICLO DE DESARROLLO DE LOS SISTEMAS DE INFORMACIÓN.

El ciclo de vida de los sistemas (SDLC, Systems Development Life Cycle) utilizado en la gestión de proyectos, es un enfoque formado por etapas de análisis y diseño que mejora el desarrollo de los sistemas implementando el uso de un ciclo específico de actividades del analista y *usuarios*. [8]

Se han desarrollado varias metodologías o ciclos de vida (SDLC) para desarrollar sistemas, los cuales se describen a continuación:

a. Modelo Lineal Secuencial

Llamado también “Ciclo de vida básico o Modelo de cascada”, este modelo de desarrollo de *software* es el más antiguo y está basado en un enfoque secuencial de actividades, donde la estrategia es seguir el progreso del desarrollo del *software* hacia puntos de revisión definida mediante entregas calendarizadas (Figura 1.3.). Las fases de dicho modelo son:

1. *Análisis*: aquí se reúnen los requisitos que debe cumplir el *software*, extrayendo una clara idea del problema a resolver mediante cuestionarios, entrevistas, observaciones, etcétera).
2. *Diseño*: esta fase está construida por la arquitectura del *software*, las representaciones de la interfaz y el detalle procedimental (algoritmo).
3. *Generación de Código*: llamado también Implementación, en esta parte se genera *código* entendible por la máquina, dependiendo de lo detallado del diseño.
4. *Pruebas*: después de generar el *código*, en esta etapa se lleva a cabo el proceso de depuración, donde se comprueba que el sistema funciona correctamente antes de ser puesto en marcha. Asegurando que a partir de los datos de entrada se genere la salida deseada.
5. *Mantenimiento*: el *software* obtenido se pone en producción, en esta fase se corrigen errores no detectados en la etapa de pruebas, o introducen mejoras debido a nuevos requerimientos del cliente. Aplicando nuevamente las fases anteriores.



Figura 1.3. Modelo Lineal Secuencial

b. *Modelo de Construcción de Prototipos*

También llamado modelo con prototipación, se inicia con los objetivos globales para el *software*, identificando los requisitos ya conocidos y las áreas que requieran ser analizadas. Se plantea con rapidez la iteración de prototipos y se realiza el modelado (Figura 1.4.).



Figura 1.4. Modelo de Construcción de Prototipos

c. *Modelo de Desarrollo Rápido de Aplicaciones (RAD)*

Consiste en desarrollar un modelo rápido a corto plazo, utilizando un enfoque de construcción de componentes, permitiendo crear un sistema completamente funcional (Figura 1.5.).

1. Modelo de gestión.- se modela el flujo de *información*, respondiendo a preguntas como ¿Quién la genera?, ¿A dónde va la *información*?, ¿Quién la procesa?
2. Modelado de datos.- se definen los atributos de los objetos creando las relaciones entre sí.
3. Modelado del proceso.- se crean descripciones del proceso para añadir, modificar o eliminar un objeto de datos.
4. Generación de aplicaciones.- Reutilización de componentes de programas existentes o se crean componentes reutilizables.
5. Prueba y entrega.- los componentes ya han sido examinados y probados anteriormente, por lo que las pruebas se realizan en menos tiempo; los componentes nuevos, se prueban uno por uno.

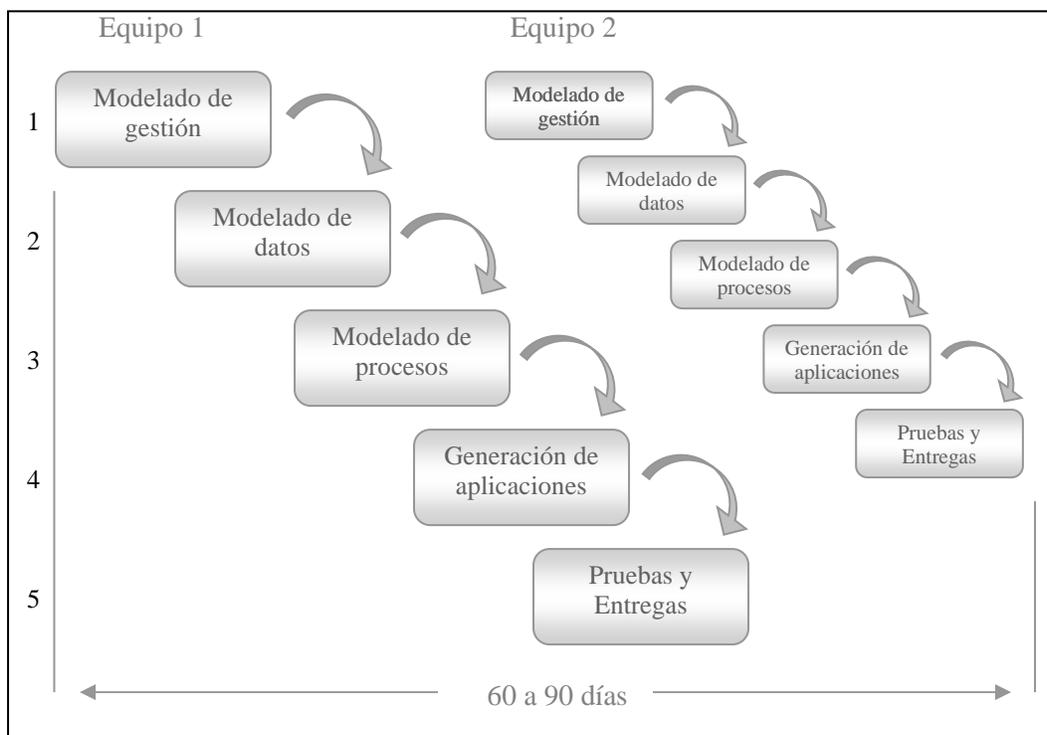


Figura 1.5. Modelo de Desarrollo Rápido de Aplicaciones (RAD)

d. *Modelo Incremental*

Para la producción del *software* con este modelo, se usa el principio de “trabajo en cadena o Pipeline”, con ello para mantener al cliente en contacto con los resultados obtenidos en cada incremento a fin que el *software* se adapte a sus necesidades. El proceso se repite hasta que el *software* este completo (Figura 1.6.).

El modelo se lleva a cabo cuando particularmente no se cuenta con el personal suficiente, y en cada incremento añadir personal si es necesario. Este método se puede planear para cubrir riesgos técnicos.

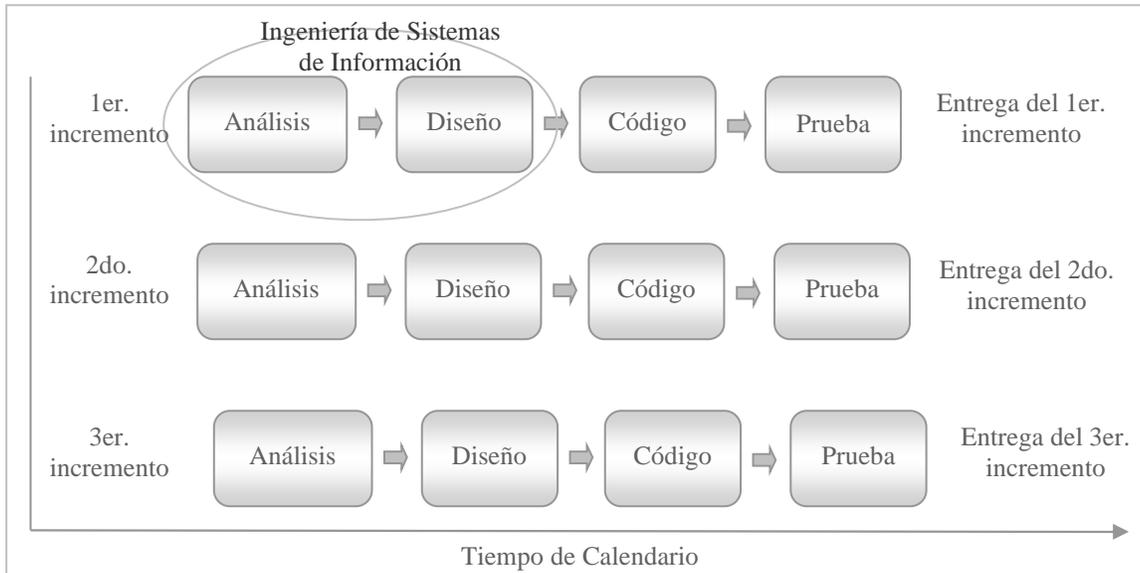


Figura 1.6. Modelo Incremental

e. *Modelo Espiral*

Este modelo se divide en *bucles*, donde cada bloque es una actividad, las cuales no están fijadas a prioridad, sino que se eligen en función del análisis, comenzando desde el centro del *bucle*. En cuanto se termina una actividad comienza otra hasta terminar con el *software* (Figura 1.7).



Figura 1.7. Modelo Espiral

Cada actividad comprende:

1. *Comunicación con el cliente:* el desarrollador establece la comunicación con el cliente.
2. *Planificación:* se fijan requerimientos, especificaciones, restricciones y tiempos.
3. *Análisis de riesgos:* se estudian todos los posibles riesgos y se seleccionan una o varias alternativas para reducirlos o evitarlos
4. *Ingeniería:* se construye un o varios esquemas para la aplicación.
5. *Construcción y Adaptación:* se desarrollan las actividades (construir, instalar, probar) del software, proporcionando soporte al usuario.
6. *Evaluación del cliente:* se evalúa el *software* con el cliente para su aprobación por el mismo.

f. *Modelo de Ensamblaje de Componentes*

El modelo trabaja orientado a objetos, utilizando muchas de las características del modelo en espiral. Es evolutivo y exige un enfoque interactivo para la creación del *software* (Figura 1.8.).

Existen clases candidatas, las cuales podrán utilizarse nuevamente cuando se necesiten, y en caso de no existir, se podrán crear e incluir en la biblioteca de clases. La principal ventaja de este modelo, es la reutilización del *software*.

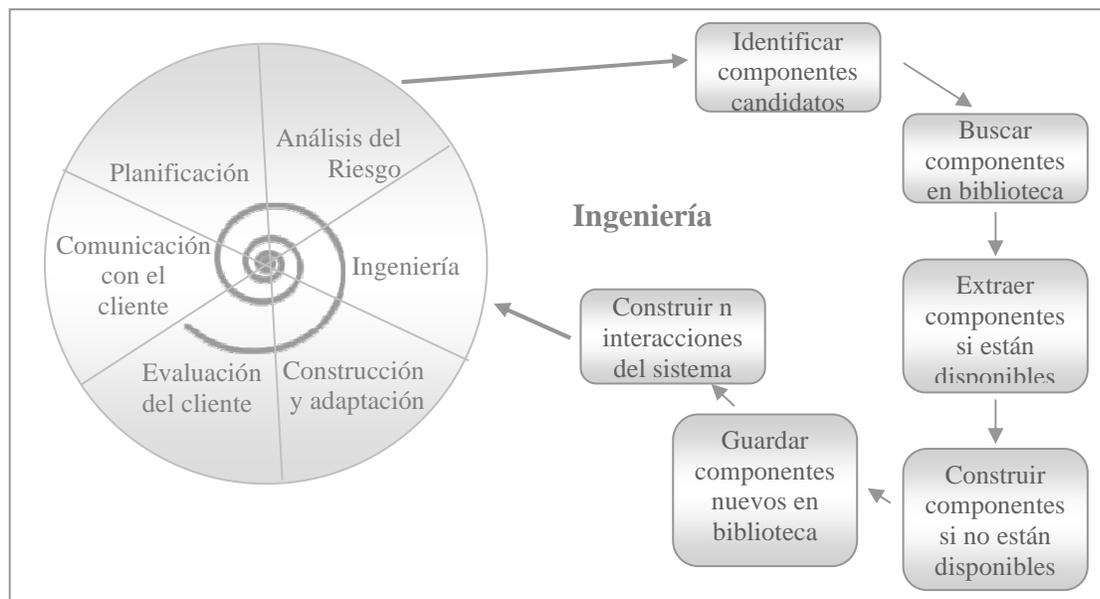


Figura 1.8. Modelo de Ensamblaje de Componentes

1.2.9. BENEFICIOS DE LOS SISTEMAS DE INFORMACIÓN.

Los *sistemas de información* se especializan en el desarrollo, uso y administración de las *TI (Tecnología de Información)* en una empresa. La cuál nos aporta beneficios como:

- La *información* es clasificada por temas de interés general y particular, lo cual facilita su manejo.
- Tomar eficientes tomas de decisiones, debido al rápido acceso a la información necesaria.
- Ahorra tiempo recopilando *información* almacenada en las BD compartida.
- Impulso a la creación de grupos de trabajo, debido a la facilidad para encontrar *información* y manipulación de la misma.
- Mejoran la integración y comunicación entre las áreas de la organización.
- Permiten corregir, detectar y controlar fallas al generar *informes*.
- Reducen la mano de obra y reducen costos.
- Ayudan a la mejora de calidad de los productos y/o servicios.
- Contribuyen a lograr las metas y objetivos de la organización.
- Garantizan la supervivencia de la organización en un entorno competitivo y exigente.

1.3. TOPOLOGÍA DE RED

1.3.1. CONCEPTO DE TOPOLOGÍA DE RED

La *topología* es la forma en como están conectados entre sí los diferentes *nodos*; la forma en como se comunican es la *topología de red*. [2]

1.3.2. CLASIFICACIÓN DE LAS TOPOLOGÍAS DE RED

1.3.2.1. TOPOLOGÍA EN ESTRELLA

Todos los *dispositivos* están conectados a un *hub* central. Los *nodos* se comunican en la *red* a través del *hub* (Figura 1.9).

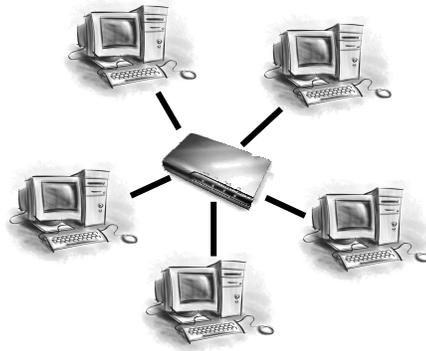


Figura 1.9. Topología de estrella.

1.3.2.2. TOPOLOGÍA EN BUS

En esta topología todos los *dispositivos* son conectados a un único canal de comunicaciones (*bus*). Cada *host* está conectado a un cable común, por lo que se comunican directamente. Una desventaja, es que si se rompe el cable, los *hosts* quedan desconectados (Figura 1.10).

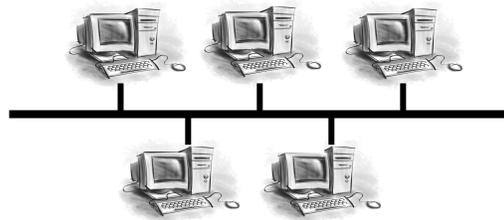


Figura 1.10. Topología de bus.

1.3.2.3. TOPOLOGÍA EN ANILLO

Topología donde cada estación esta conectada a la siguiente y la última conectada a la primera (Figura 1.11). Cada estación transfiere *información* a la estación siguiente a través de un *bucle* en una dirección, esto es por medio de un Token Ring o Pase de Testigo, el cuál pasa de equipo en equipo hasta llegar a uno que contenga información para enviar. El equipo pone una dirección electrónica en los *datos* y envía la *información* hasta llegar con el equipo que contiene dicha dirección. El cuál envía un mensaje de recibido al equipo emisor, liberando la red y creando un nuevo testigo hasta que alguien mas necesite enviar datos.

Si algún *nodo* de esta topología se cae (es decir, que no funciona) o esta en mal funcionamiento, la comunicación se pierde.

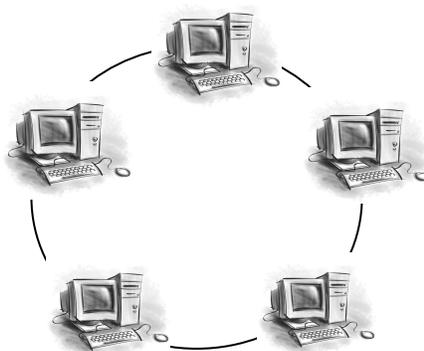


Figura 1.11. Topología de anillo

1.3.2.4. TOPOLOGÍA EN ÁRBOL

Topología en la que los *nodos* están conectados en formas de árbol, donde se encuentra un *hub* o *switch* principal del que ramifican los demás *nodos* (Figura 1.12).

Esta topología puede verse muy similar a la topología en estrella, y a la topología en bus. Donde la *información* se extiende a todas las estaciones, solo que esta se extiende de un punto raíz. Por lo que es necesario integrar un mecanismo que permita identificar al destinatario de (los) mensajes.

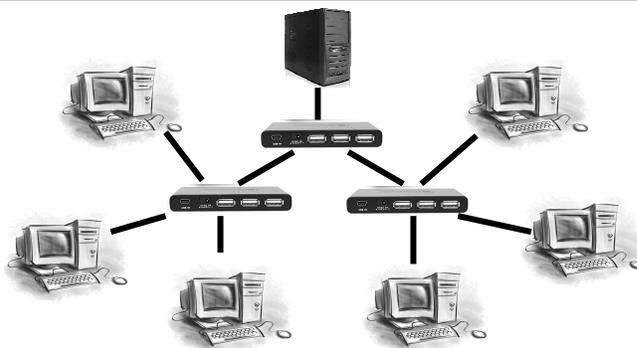


Figura 1.12. Topología en árbol.

1.3.2.5. TOPOLOGÍA EN MALLA

Topología en la que cada *nodo* está conectado a todos los *nodos*. Donde la *información* puede tomar distintas rutas hasta llegar a su destino. De esta manera no puede existir interrupción en las comunicaciones, ya que cada servidor tiene sus propias conexiones con los demás (Figura 1.13.).

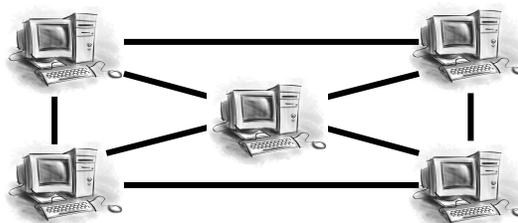


Figura 1.13. Topología en malla.

1.3.3. DISEÑO DE LA RED EN LA SEP

La *topología* de la *Secretaría de Educación Pública del Estado de Hidalgo* es del tipo Estrella distribuida (Figura 1.14). Es decir, los dispositivos están conectados a un *hub* central. Los nodos se comunican en la red a través del *hub*.

La *topología* utilizada es la de estrella distribuida, ya que los *nodos* principales se encuentran concentrados en el *SITE* de *informática*, siendo este el controlador central. Todas las transacciones pasan por este, siendo el encargado de gestionar y controlar todas las comunicaciones.

El *cableado* estructurado se encuentra certificado, bajo la *norma ISO 11801 Clase D, estándar EIA/TIA 568 B-1, categoría 5*. Los componentes de esta son marca *3m/pouyet*.

A su vez la *red* esta conformada por 7 edificios que conservan la misma configuración de estrella, todos los edificios cuentan con *switches* marca CISCO, siendo el enlace entre estos edificios y el *SITE*. Con esta configuración cualquier fallo en alguno de los edificios, no afecta en nada la operación de los demás.

La asignación de *direcciones IP* es por medio de *DHCP* y cada edificio se maneja por segmentos denominados *VLAN's*, así que es fácil de identificar por *dirección IP*, a qué edificio pertenece determinado equipo.

Se encuentra en operación un *WINS*, ya que en la *red* conviven equipos de diferentes tecnologías como son Windows 98, 2000, ME, XP y Vista, así como equipos con versiones de *Linux*.

El enlace hacia el mundo se hace por medio de un *Firewall PIX* el cual esta administrado por el proveedor del Servicio. El servicio de *Internet* se proporciona a los usuarios por medio de un *Proxy*.

1.3.4. ESTRUCTURA DE LA RED EN LA SEP

Estructura de la *red* en general de la *Secretaría de Educación Pública del Estado de Hidalgo (SEP)*. Figura 1.15.

El departamento donde se encuentra el *SITE* es en el Departamento de Informática. Donde se ubica un *rack* con características: Match Panel, CISCO PIX 5509, un *switch* 2950 y cuenta además con una *tarjeta ruteadora*.

Este da conexión a la *red* a los siguientes Edificios: Edificio Naranja, CAPEP, Edificio Principal, Recursos Materiales y Servicios, donde cabe mencionar, que son de dos plantas, cada uno con sus distintas áreas, las cuales se conectan a la *red* a través de otro *switch* ubicado en cada uno de los edificios.

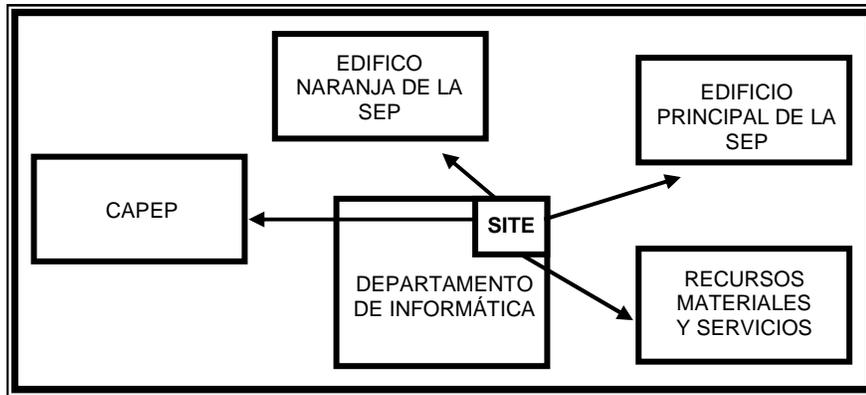


Figura 1.14. Estructura de la red en la SEP

El departamento de *Infraestructura Educativa* se encuentra ubicado en la planta alta del Edificio Principal. En la Figura 1.16 se muestra como están ubicadas las distintas áreas y en donde se encuentra ubicado el *switch* que reparte señal a cada una de las máquinas de dicho edificio. Cabe mencionar que cada planta, tanto alta como baja cuenta con un *switch* con las características del principal.

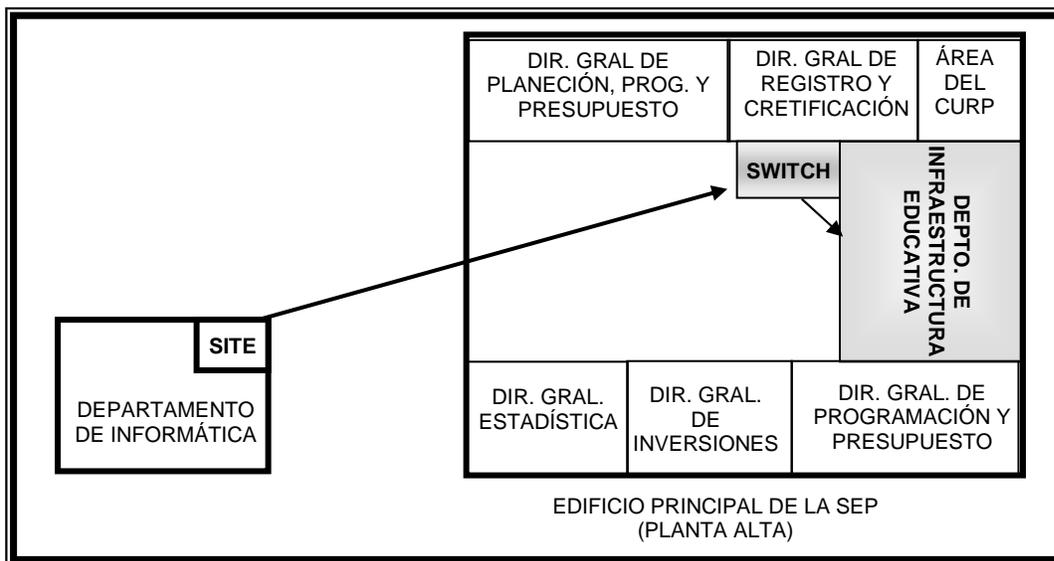


Figura 1.15. Ubicación del Departamento de Infraestructura Educativa de la SEP

El departamento donde se creó el *Sistema de Correspondencia* es el de *Informática* en el área de Desarrollo de Sistemas tal cual lo muestra la Figura 1.17. Las características del equipo donde se generó el sistema son:

- IBM ThinkCentre
- Pentium (R) 4
- 2.80 GHz
- 1.24 GB RAM
- WINDOWS XP PROFESIONAL 2002
- IIS 5.1

Y las del servidor Web:

- HP Proliant ML 150
- Core 2 Duo 2GHz
- 2 GB RAM
- 120 GB HD
- WINDOWS SERVER 2003
- IIS 6.0

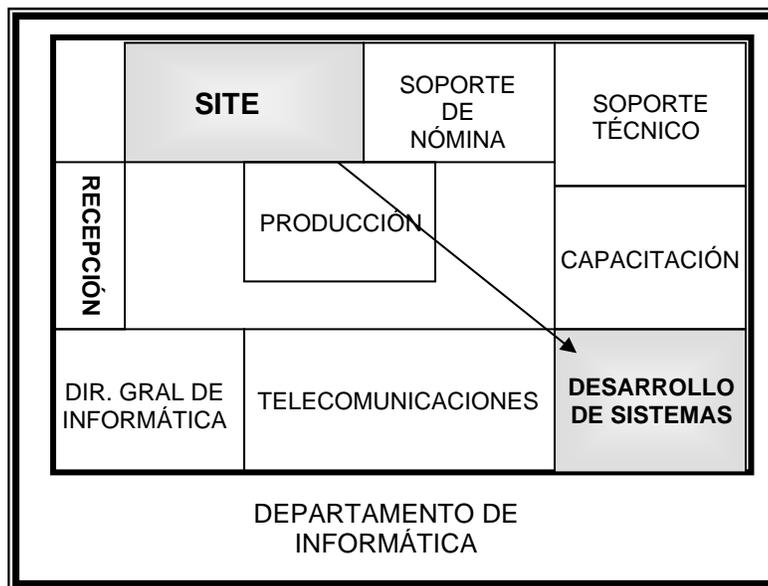


Figura 1.16. Ubicación del Equipo de desarrollo del Sistema de Correspondencia.

1.4. SISTEMA DE CORRESPONDENCIA ANTERIOR

1.4.1. PANTALLA DE ENTRADA

La pantalla de entrada del Sistema anterior, era como lo muestra la Figura 1.18

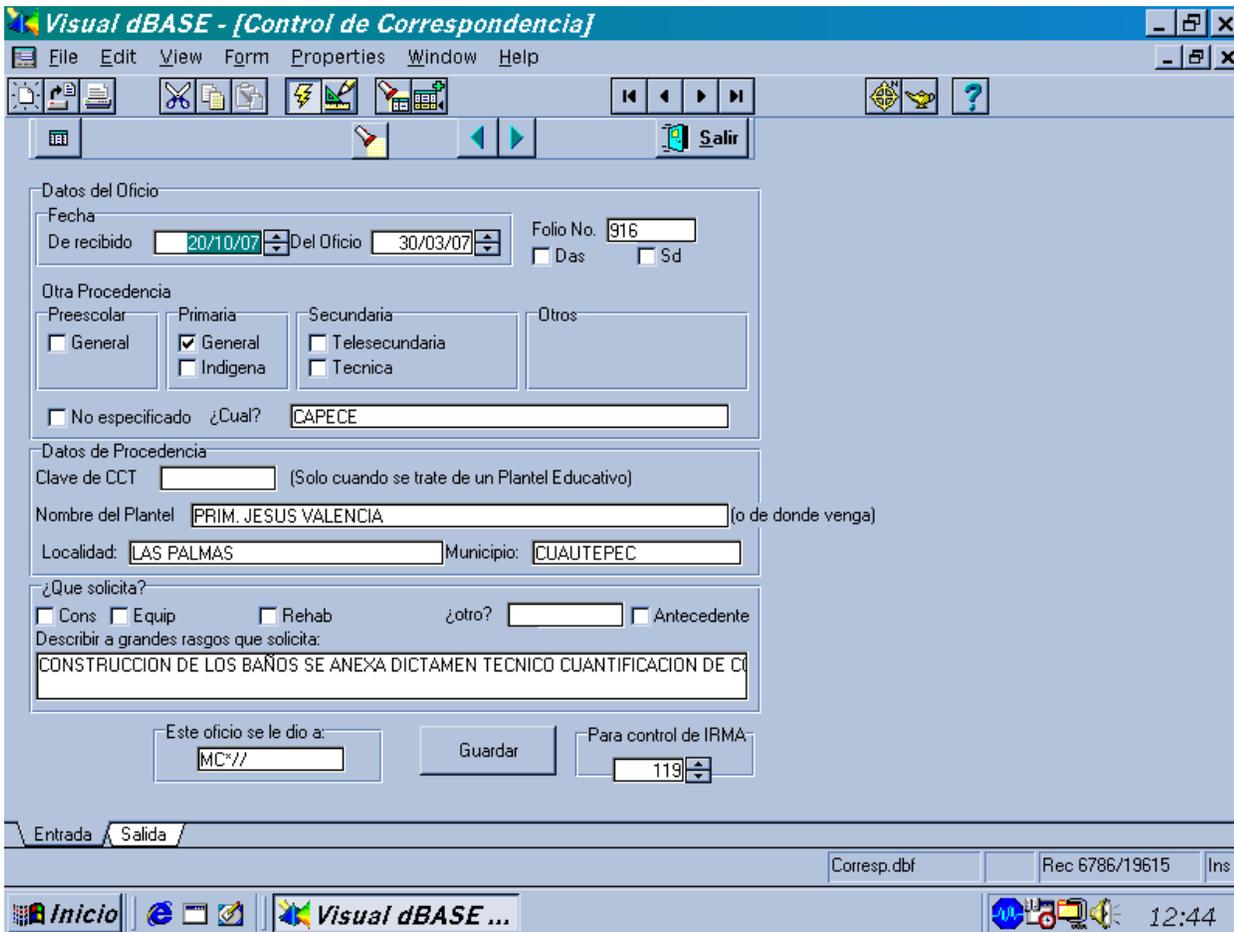


Figura 1.17 Pantalla de Entrada

- **Datos del oficio:** Aquí se capturaban los datos del oficio, la fecha de recibido y la fecha de creación del mismo. El número del oficio se capturaba en la casilla Folio No., muchas veces, los oficios de gobierno empiezan con las siglas Das o SD, por lo que cuando los oficios vienen de gobierno se selecciona una de estas casillas. En otra procedencia se selecciona la casilla correspondiente de donde se envía el oficio, y en caso de que no estuviera se escribe en la caja de texto que esta enseguida de la palabra ¿Cuál?.
- **Datos de procedencia:** Aquí se capturarán los datos del oficio, como la clave de CCT, nombre del plantel o institución que manda la solicitud, la localidad de dicho plantel y su municipio:
- **¿Qué solicita?:** Las casillas de Cons, Equip y Rehab se seleccionan de acuerdo al tipo de solicitud (Construcción, Equipamiento y Rehabilitación), en el cuadro de texto ¿Otro? Se captura cuando el tipo de solicitud es diferente a las mencionadas anteriormente, y la casilla de antecedente se selecciona cuando el oficio ya se ha mandado anteriormente. En la caja de texto de descripción de la solicitud, se describe a grandes rasgos la solicitud.
- **Este oficio se le dio a:** Este apartado se utilizaba solo para poner el nombre de la persona que daba respuesta al oficio, aunque este no lo capturara.
- **Para control de Irma:** Cuando se crearon los formularios, el nombre de la persona a cargo de manejar este sistema se llamaba Irma*, y ella llevaba un control interno de las solicitudes, y cada que ingresaba un oficio aumentaba el numero para un mejor control.

Irma. Era la persona encargada del manejo del Sistema en aquel tiempo.

1.4.2. PANTALLA DE SALIDA:

Para grabar la respuesta a los oficios, se abre la pantalla de salidas, donde aparecen todos los oficios desde que se crearon los formularios (Figura 1.19).

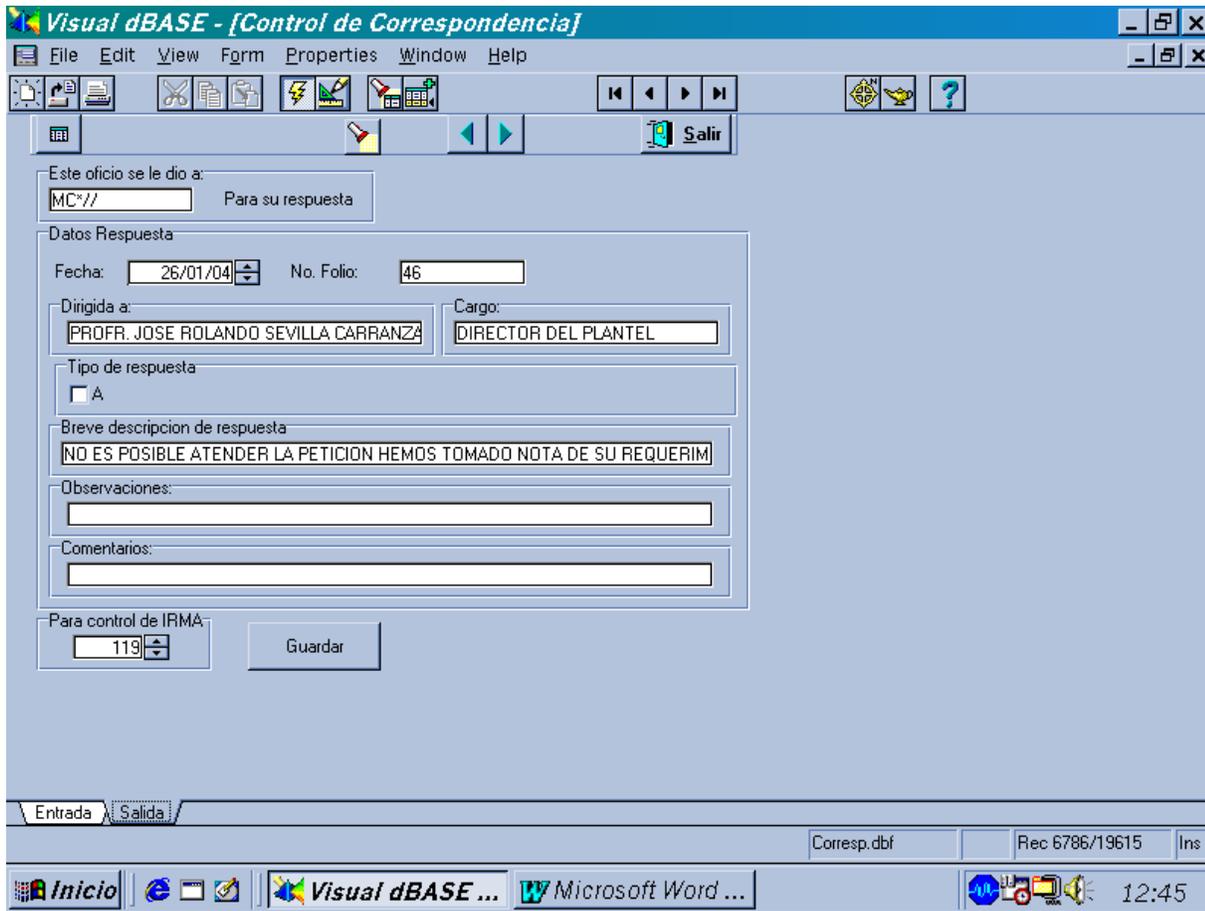


Figura 1.18. Pantalla de Salida

Cuando se buscaba un oficio en particular al que se le desea dar respuesta, se le daba clic en la siguiente imagen:  Automáticamente abría otro formulario de búsqueda (Figura 1.20.).

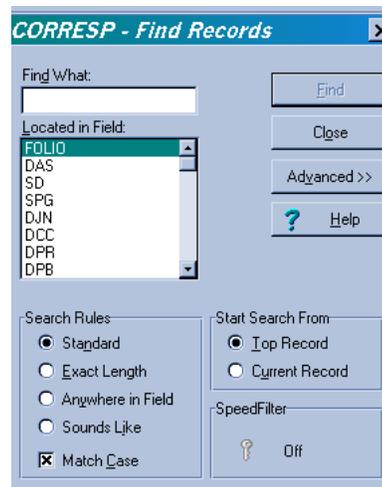


Figura 1.19. Opciones de búsqueda

Donde las búsquedas del oficio son por FOLIO, o por las siglas DAS o SD, que son para diferenciar los oficios de gobierno y las demás instituciones o planteles. Las demás siglas no se ocupaban.

- **Este oficio se le dio a:** En este apartado, aparecía el nombre de la persona quien daba respuesta al oficio.
- **Datos Respuesta:** Se capturaba la fecha en que se dio respuesta al oficio, y el número de folio de respuesta del mismo.
- **Dirigido a:** Nombre de la persona a quien iba dirigido el oficio.
- **Cargo:** Cargo de la persona a quien se dirigía el oficio.
- **Tipo de respuesta:** La casilla A nunca se ocupó, y siempre se dejaba en blanco. En breve descripción de capturaba la descripción de la respuesta, así como sus observaciones y comentarios.
- **Para control de Irma:** Se capturaba solo para llevar un control interno acerca del número de folios capturados.
- **Guardar:** Con este botón se guardaba las respuestas de los oficios.

Irma. Era la persona encargada del manejo del Sistema en aquel tiempo.



Capítulo 2. Análisis

En este capítulo se indica el análisis de requerimientos del Departamento de Infraestructura de la SEP, además, se comenta cómo utilizaba la información el Sistema de Correspondencia anterior, y se evaluarán funciones y rendimientos y restricciones del Sistema de Correspondencia Actual.

La metodología es el estudio del *método* de investigación que se sigue para alcanzar una gama de objetivos específicos. Abarcan planificación y estimación de proyectos, análisis de requisitos, diseño, codificación, pruebas y mantenimiento. [9]

El modelo que se utiliza para el desarrollo del *Sistema de Correspondencia* es el *Lineal Secuencial* y se clasifica dentro de los *Sistemas de Información administrativa*, debido a que proporcionará *información* necesaria, y facilitará su manejo y a la toma de decisiones.

El análisis es donde se reúnen los requisitos que debe cumplir el *software*, extrayendo una clara idea del problema a resolver para su mejora. Lo cuál se obtuvo con las especificaciones que el usuario del sistema anterior y el jefe del Departamento de Infraestructura Educativa de la SEP hicieron conocer al jefe del Departamento de Desarrollo de Sistemas de la misma organización.

2.1. CONTROL DE OFICIOS EN EL DEPARTAMENTO DE INFRAESTRUCTURA EDUCATIVA DE LA SEP.

En el Departamento de Infraestructura Educativa de la Secretaría de Educación Pública del Estado de Hidalgo, llegan solicitudes diarias de construcción, equipamiento, mobiliario, etcétera de las distintas dependencias de la *SEP*.

Por lo que para llevar un mejor control en el año de 1998 se crearon unos *formularios* en *dBase*. Estos formularios se entregaron al Departamento de Infraestructura Educativa, sin *código* y con la finalidad de que no se pudiera realizar cambios sin consultar y dar una paga al propietario. El *Sistema de Correspondencia* anterior, consta de 2 simples *formularios*, el de Entradas y Salidas.

En el formulario de Entradas se capturaba la solicitud con todos los datos que traía el oficio, y en el de Salidas, solo se daba captura de las respuestas a dichas solicitudes.

Dichos *formularios* no cuentan con las siguientes cualidades:

- **Precisión:** La *información* que arrojaban los *formularios* no era precisa, ya que almacenaba información basura o capturaba campos en blanco. Ya que la precisión a exigir dependerá de una aplicación concreta que arroje resultados con información necesaria.

- **Exactitud:** La *información* guardada en los *formularios* no era exacta, ya que mucha de la *información*, no era la *información* requerida, es decir no se encontraba clasificada, por lo que al buscar cierta *información* el usuario se encontraba con muchas dificultades.
- **Integridad:** La *información* que contenían los *formularios* era muy pobre, y para que una aplicación cuente con la integridad debe de ser lo mas completa posible, y por lo mencionado anteriormente, esta aplicación era la excepción.
- **Oportunidad:** La *información* ha de ser oportuna, es decir, debe llegar al usuario con el tiempo necesario para que éste pueda actuar (en función de dicha *información*). El tiempo disponible para llegue oportunamente varía mucho en función de la aplicación y puede ser desde pocos a varios microsegundos. En esta aplicación, la *información* guardada es del año de 1998 al 2007, por lo que la respuesta se tardaba en visualizarse, ya que era demasiada guardada en la PC y muchas veces también porque la información no se encontraba a la primera búsqueda.
- **Significatividad:** para que el sistema contara con esta cualidad, la información que manejaba tendría que ser clara y relevante, pero como mucha de la información guardada quedaba en espacios en blanco, dicha información no se comprendía, ya que en varios registros lo único que se logró recuperar, fue los números de los registros.

Los problemas con la aplicación surgen cuando el usuario requiere de la impresión de los informes detalladamente. Otro de los problemas es que al momento de mandar a impresión, se imprimían todos los folios generados desde 1998 hasta el 2007, lo cual lógicamente al usuario, le generaba muchos gastos.

Otro problema fue que la información guardada en la computadora era demasiada, por lo que la computadora se hizo lenta, y al usuario le era tedioso manejar los formularios.

Por lo mencionado anteriormente, es necesario, crear una base de datos para que el personal administrativo cuente con la información necesaria para darle solución a dichas solicitudes, centralizando la *información*, y se hará mediante tablas en *SQL Server 2000*, ya que esto hace que la *información* sea precisa, exacta, integra, oportuna y significativa. Y tendrá una interfaz realizada en *Visual .NET 2003*, la cual nos ayudará a un mejor manejo de la información, ya que será amigable y fácil de utilizar. Además de proporcionar al usuario los informes solicitados, esto a través de consultas programadas.

2.2. DESCRIPCIÓN DEL CONTROL DEL SISTEMA DE CORRESPONDENCIA

- Las solicitudes de oficios que llegan al Departamento de Infraestructura Educativa de la SEP, se dividen en:
 - Construcción
 - Equipamiento
 - Rehabilitación
 - Mobiliario
 - Construcción, Equipamiento y Rehabilitación
 - Sin especificar
 - Otros (en esta opción se podrá capturar en una *caja de texto*)
- Se capturará la solicitud, ya que existen distintos centros de trabajo que piden lo mismo, y a dicha solicitud se le podrá anexar una o varias claves del CCT (Clave del Centro de Trabajo).
- Cada oficio contará con un id_folio, como número identificador, el cuál será consecutivo automático y no aceptará duplicados.
- En caso de ser un oficio para varios CCT se creará una tabla donde se guardará el id_folio junto con la clave del CCT, sin importar la duplicación del id_folio.
- Se crearán varios usuarios con sus respectivas contraseñas para que la persona encargada de capturar los oficios, les haga llegar el oficio para que capturen la respuesta o solución.
- Los usuarios solo podrán dar de alta las respuestas y consultas que laboran. Es decir, que no podrán consultar la información de otros usuarios, con excepción del usuario principal, que es quien capturará los oficios.
- Cada año, se creará una nueva base de datos para empezar a contabilizar los oficios desde el id_folio 1.

2.3. DESCRIPCIÓN DE PANTALLAS

De acuerdo a los requerimientos del Departamento de Infraestructura Educativa, se identifican en éste análisis, las pantallas o módulos fundamentales para la estructura del sistema a desarrollar.

2.3.1. PANTALLA DE ENTRADAS

En este modulo o pantalla se capturarán los oficios que lleguen al Departamento de Infraestructura Educativa de la SEP por el usuario principal. Donde se podrá anexar la(s) clave(s) del centro de trabajo (CCT) que mando dicho oficio. También se podrán consultar todos los oficios capturados, y hacerle modificaciones o borrarlos de la *BD*. Además se podrá mandar el documento al usuario encargado de dar dicha respuesta.

2.3.2. PANTALLA DE SALIDAS

En este módulo se podrá buscar el(los) oficios necesarios para capturar la respuesta a dicho oficio; donde podrá hacer la captura solo de los oficios que le sean mandados a dicho usuario.

2.3.3. PANTALLA DE CONSULTAS DE INFORMACIÓN ACTUAL

Se permitirá hacer *consultas* de la información actual, al igual que en la captura de respuestas, solo aparecerán los oficios que le correspondan al usuario que este dentro del sistema. Donde además de buscar el oficio por distintas opciones, podrá imprimir el(los) oficio(s) requeridos por el usuario.

2.3.4. PANTALLA DE CONSULTAS DE INFORMACIÓN ANTERIOR

Aquí se llevara a cabo las *consultas* de la *información* obtenida de los formularios en *dBase*, donde se podrá también imprimir el(los) oficio(s) requeridos por el usuario. No se podrá borrar ni hacer modificaciones.

2.3.5. PANTALLA DE CONSULTAS DE INFORMES X CCT

La *consulta* se podrá realizar buscado una CCT (Clave del Centro de Trabajo) en particular, con la que mostrara en un *grid*, los oficios capturados con dicha clave y tendrá la opción de imprimir el oficio solicitado por el usuario.



Capítulo 3. Diseño y Construcción

Este capítulo contiene el diseño de cómo se realizó el sistema de Correspondencia Actual en Visual Basic .NET 2003 y la Base de Datos en SQL Server 2000, con sus respectivas tablas y vistas.

En esta parte se presenta la descripción del diseño de la base de datos "Corresp_plan", además de las tablas correspondientes para que el Sistema funcione adecuadamente.

3.1. PROGRAMACIÓN

3.1.1. CONCEPTO DE PROGRAMACIÓN

Los programas son un conjunto de instrucciones (código fuente) escrito en un lenguaje de programación, el cuál se prueba y depura para crear el *software* o programa informático, esperando que devuelva el resultado esperado. Existen diferentes Técnicas de Programación, que mejoran el proceso de creación de *software* y su mantenimiento, algunas de ellas son:

- Programación Orientada a Objetos (POO): Técnica de programación que utiliza un conjunto de *objetos* que colaboran entre si haciendo el programa más fácil de escribir, mantener y reutilizar. Los *objetos*, son entidades que combinan estado (uno o varios atributos a los que se les asigna valores concretos o datos), comportamiento (procedimiento o método que utiliza un objeto) e identidad (propiedad del *objeto*).
- Programación Estructurada: es una forma de programar utilizando solo 3 estructuras: secuencial, selectiva e iterativa. Con estas estructuras, se pueden construir todos los programas y aplicaciones posibles. Un inconveniente, es que este método utiliza solo un bloque de programación, por lo que cuando el *código* es demasiado grande, puede resultar problemático su manejo.
- Programación Modular: históricamente es una evolución de la programación estructurada, y consiste en dividir un programa complejo en módulos o subprogramas con el fin de hacerlo manejable y comprensible. Algunos de estos módulos necesitan de otros para poder operar, ya que cada uno, tiene una función bien definida, llamados procedimientos o funciones.
- Programación dirigida por Eventos: Crea programas de windows, donde la ejecución del programa va determinado por los sucesos que ocurran en el sistema o que los usuarios provoquen, es decir que los usuarios mismos tendrán el manejo del sistema, ya que al iniciar el sistema, éste llevara a cabo las inicializaciones y demás *código* inicial, pero después, quedará bloqueado hasta que el usuario produzca un evento. Un ejemplo, sería una opción de *menú*.

3.1.2 VISUAL STUDIO .NET 2003 (VB.NET)

Conjunto de herramientas de desarrollo para la construcción de aplicaciones Web ASP, servicios Web XML, Visual Basic.NET, Visual C++.NET, Visual C#.NET y Visual J#.NET utilizando el contorno IDE. Es un lenguaje de programación orientado a objetos que utilizan funciones de .NET Framework. [3]

Entorno .NET Framework

El Framework.NET es un entorno multilinguaje que genera, implanta y ejecuta *Servicios Web* y *Aplicaciones XML*, desarrollado por Microsoft que crea *plataformas* con transparencia de *redes*, independencia de hardware y rápido desarrollo de aplicaciones: *económica, segura y robusta*. [3]

Es el encargado de proveer servicios automáticos al *código* que se ejecuta, como son:

- Permitir cargar en la memoria las clases.
- Eliminar de memoria *objetos* no utilizados.
- Controlar que las variables usen el área de memoria que tienen asignado.
- Transformar *código* intermedio de alto nivel a *código máquina* propio del *dispositivo* que lo ejecuta.
- Manejar los errores que se producen en la ejecución del *código*.
- Permitir ejecutar el *código* con lenguajes distintos.
- Administrar seguridad al *código* que se ejecuta.
- Ejecutar el *código* en forma paralelo
- Coordinar comunicación con los componentes .COM
- Manejar estructuras como el java y .NET.

3.1.2.1. CREACIÓN DE UN PROYECTO EN VISUAL STUDIO.NET 2003

El sistema de Correspondencia, se creó realizando los siguientes pasos:

1. Abrir Microsoft Visual Studio.net (Figura 3.1)



Figura 3.1. Pantalla de Bienvenida de Visual Studio.NET

2. Para crear un proyecto, se crea una Solución en blanco. Seleccionando menú Archivo, Nuevo y Solución en blanco (Figura 3.2)

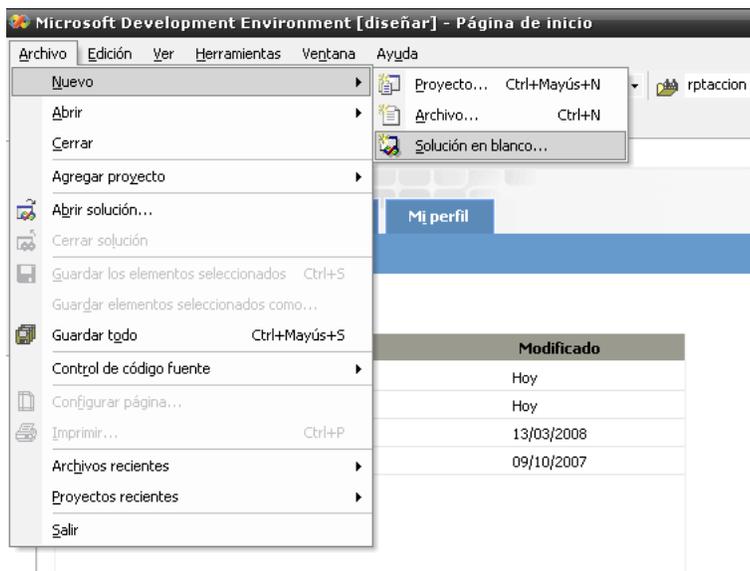


Figura 3.2. Creación de una Solución

3. En seguida, aparecerá la pantalla (Figura 3.3.), donde en Nombre, se escribirá el nombre de la solución.



Figura 3.3. Pantalla de Solución

4. Se da clic en el botón Aceptar y la solución aparecerá en el Explorador de Soluciones (Figura 3.4.).

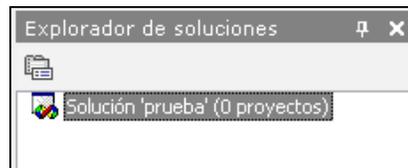


Figura 3.4. Explorador de Soluciones

5. En el Explorador de Soluciones, se dará clic con el botón derecho del mouse sobre la solución, se seleccionará Agregar, Nuevo proyecto para empezar a programar (Figura 3.5).

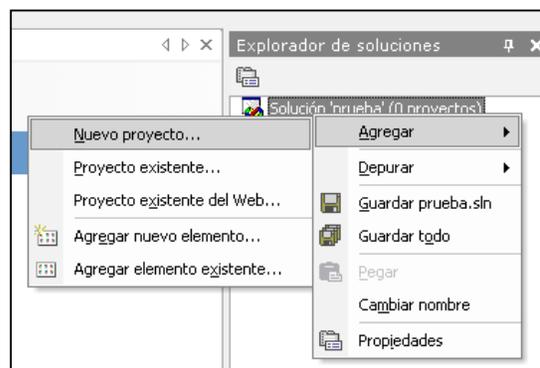


Figura 3.5. Creación de un proyecto.

6. Aparecerá la siguiente pantalla (Figura 3.6), donde se escribirá el nombre y la ruta del proyecto (el cuál es recomendable el mismo nombre de la solución).

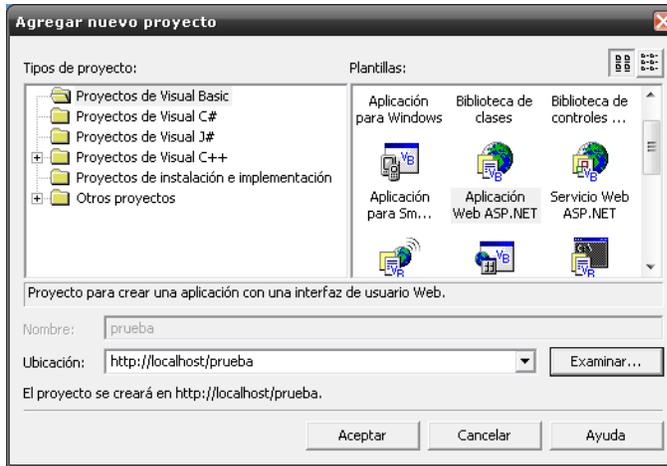


Figura 3.6. Ubicación y nombre del proyecto.

7. Al darle clic en el botón Aceptar, aparecerá la pantalla en la que se podrá empezar a programar para realizar el proyecto (Figura 3.7.).

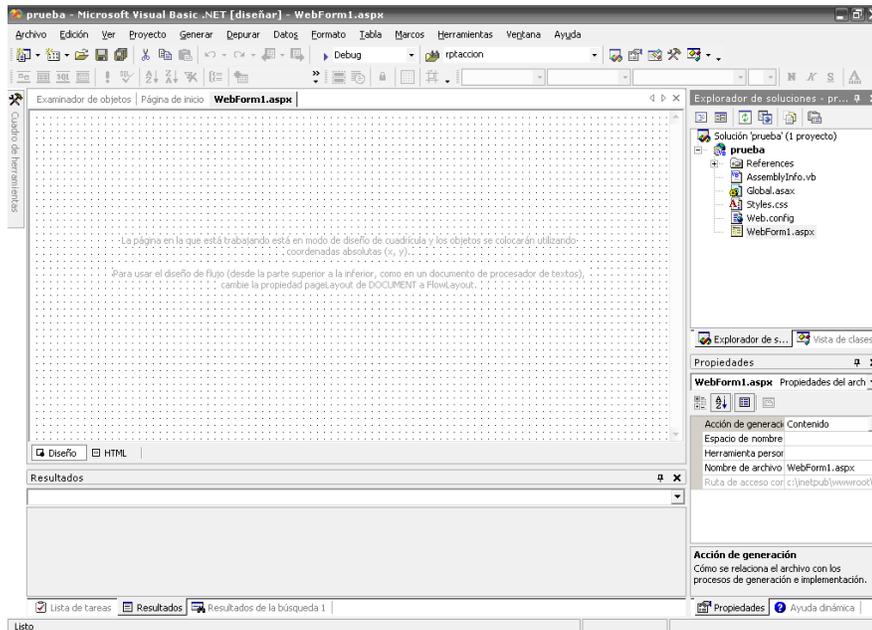


Figura 3.7. Pantalla para empezar a programar.

3.2. MICROSOFT SQL SERVER 2000

Potente motor de bases de datos de alto rendimiento capaz de soportar millones de registros por tabla con interfaz intuitivo y con desarrollo de herramientas integradas como Visual Studio 6.0 o Visual Studio.net. Además que se puede crear cualquier aplicación que manipule componentes de SQL Server, todo esto desde el administrador de SQL desde Visual C++ hasta Visual Basic, ASP y .NET. Las aplicaciones o una base de datos deben de ser cuidadosamente creadas, ya que un mal diseño, hace ser de SQL Server una mala herramienta para el manejo de la información. [4]

SQL Server 2000 instala por default 6 bases de datos, 2 de ejemplos donde se pueden realizar pruebas y 4 de sistemas: master, model, tempdb y msdb. Los cuales se describen a continuación:

- Master: Esta base es la más importante de todas, ya que guarda la información de todas las bases de datos que tiene SQL Server, además de gestionar usuarios, permisos y parámetros de configuración.
- Model: Plantilla para crear nuevas bases de datos.
- Tempdb: Espacio temporal para procesar consultas y tareas de SQL Server.
- Msdb: Mantiene datos de alertas. Trabajos, notificaciones y tareas programadas.

SQL Server tiene varias herramientas de administración y utilidades. La más importante es el administrador corporativo, en el que se puede realizar operaciones necesarias, como son:

- Definir grupos de servidores SQL Server
- Registrar servidores individuales dentro de un grupo
- Configurar todas las opciones de SQL Server en los servidores registrados.
- Crear y administrar todas las bases de datos, usuarios, objetos, inicios de sesión y permisos de SQL Server.
- Diseñar y probar de forma interactiva instrucciones SQL, archivos por lotes y secuencias de comandos al invocar el Analizador de consultas SQL.

3.2.1. CREACIÓN DE UNA BASE DE DATOS

1. Abrir el Administrador corporativo, donde se seleccionará el servidor a utilizar posicionándose en Bases de datos, y con el botón derecho del mouse dar clic en Nueva base de datos (Figura 3.8). [10]



Figura 3.8. Administrador corporativo.

2. Enseguida aparecerá la pantalla “Propiedades de la Base de Datos” (Figura 3.9). En donde se escribirá el nombre de la BD y el orden de intercalación, que por defecto, se deja el predeterminado del servidor.

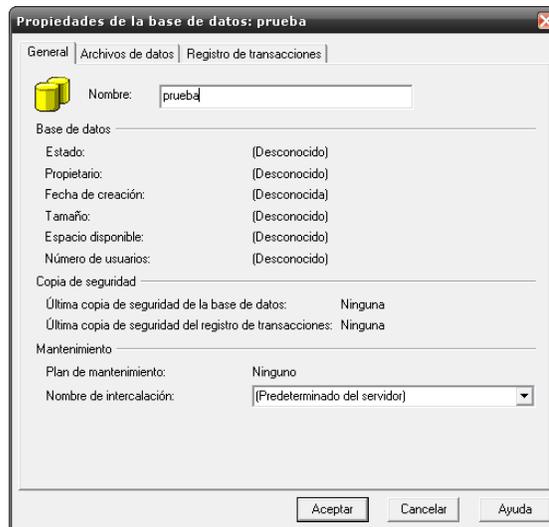


Figura 3.9. Propiedades de la Base de Datos.

3. Se pulsa el botón de aceptar, y se abre la pantalla de “Archivo de datos” (Figura 3.10) donde se seleccionará la ruta donde estará el fichero físico de datos, y se le da clic en Aceptar.

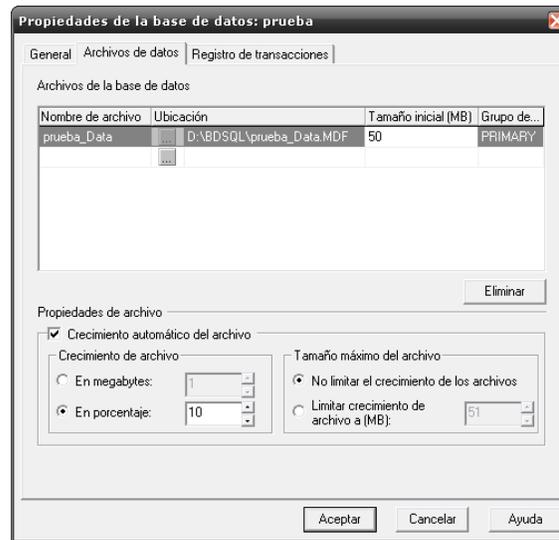


Figura 3.10. Archivo de Datos.

4. Se abre la pantalla de “Registro de Transacciones” (Figura 3.11) donde se selecciona la ruta donde estará el fichero de transacciones.



Figura 3.11. Registro de Transacciones.

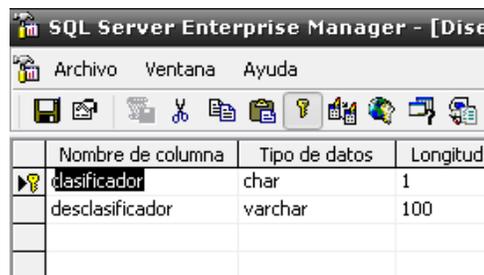
5. Se pulsa el botón de aceptar, y la base de datos estará creada en el servidor, lista para usarse.

3.2.2. DISEÑO DE LA BASE DE DATOS “CORRESP_PLAN”

3.2.2.1. CREACIÓN DE TABLAS

En esta parte del capítulo se presenta la descripción del diseño de la base de datos “Corresp_plan” así como las tablas correspondientes para que el Sistema funcione adecuadamente.

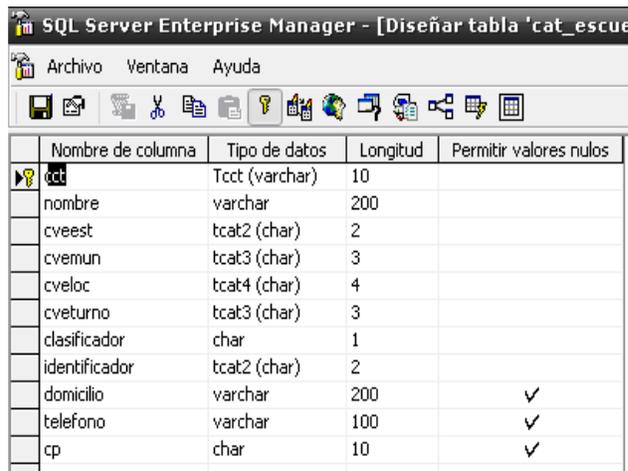
- **CAT_CLASIFICADOR:** Catálogo del tipo de Centro de Trabajo (Figura 3.12.)



Nombre de columna	Tipo de datos	Longitud
clasificador	char	1
desclasificador	varchar	100

Figura 3.12. Catálogo “cat_clasificador”

- **CAT_ESCUELAS:** Catalogo de las Escuelas del Edo. de Hidalgo (Figura 3.14).



Nombre de columna	Tipo de datos	Longitud	Permitir valores nulos
tcat	Tcct (varchar)	10	
nombre	varchar	200	
cveest	tcat2 (char)	2	
cvemun	tcat3 (char)	3	
cveloc	tcat4 (char)	4	
cveturno	tcat3 (char)	3	
clasificador	char	1	
identificador	tcat2 (char)	2	
domicilio	varchar	200	✓
telefono	varchar	100	✓
cp	char	10	✓

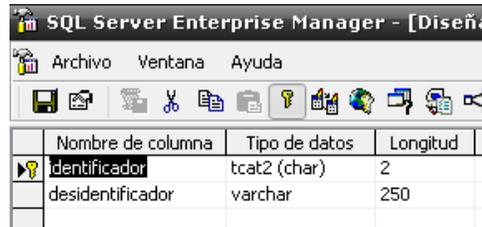
Figura 3.14. Catálogo “cat_escuelas”

- **CAT IDENTIFICADOR:** Catálogo de identificador del tipo del centro de trabajo (Figura 3.15). Ejemplo:

Del catálogo “cat_clasificador” tipo Administrativo, el identificador, sería:

AA - Academia de Artes.

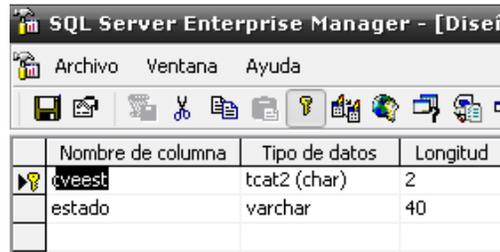
AB - Asociación de Padres de Familia (Educación Preescolar)



Nombre de columna	Tipo de datos	Longitud
Identificador	tcat2 (char)	2
desidentificador	varchar	250

Figura 3.15. Catálogo “cat_identificador”

- **CAT EDOS:** Catálogo de los distintos Estados de la República Mexicana (Figura 3.13).



Nombre de columna	Tipo de datos	Longitud
cveest	tcat2 (char)	2
estado	varchar	40

Figura 3.13. Catalogo “cat_edos”

- **CAT LOCALIDADES:** Catálogo de las Localidades del Estado de Hidalgo (Figura 3.16).



Nombre de columna	Tipo de datos	Longitud
cveest	tcat2 (char)	2
cvemun	tcat3 (char)	3
cveloc	tcat4 (char)	4
localidad	varchar	200

Figura 3.16. Catálogo “cat_localidades”

- **CAT MUN:** Catálogo de Municipios del Estado de Hidalgo (Figura 3.17).

Nombre de columna	Tipo de datos	Longitud
cvceest	tcat2 (char)	2
cvemun	tcat3 (char)	3
municipio	varchar	60

Figura 3.17. Catálogo “cat_mun”

- **CAT STAT:** Catálogo de los Estados de Hidalgo (Figura 3.18).

Nombre de columna	Tipo de datos	Longitud
cvestat	tcat2 (char)	2
status	varchar	35

Figura 3.18. Catálogo “cat_stat”

- **CAT TPROC:** Catálogo de lugar de procedencias de los Oficios. Ejemplo: *CAPECE, Escuela, Presidencia Municipal*, entre otros (Figura 3.19).

Nombre de columna	Tipo de datos	Longitud
cvetproc	tcat2 (char)	2
descripcion	varchar	100

Figura 3.19. Catálogo “cat_tproc”

- **CAT_TSOLIC:** Catálogo de Solicitudes. Ejemplo: *Construcción, Equipamiento, Mobiliario, entre otros* (Figura 3.20).

Nombre de columna	Tipo de datos	Longitud
cvesolic	tcat2 (char)	2
descripcion	varchar	100

Figura 3.20. Catálogo “cat_tsolic”

- **CAT_TURNO:** Catálogo de Turnos de las Escuelas. Ejemplo: *Matutino, Vespertino* (Figura 3.21).

Nombre de columna	Tipo de datos	Longitud
cveturno	tcat3 (char)	3
desturno	varchar	50

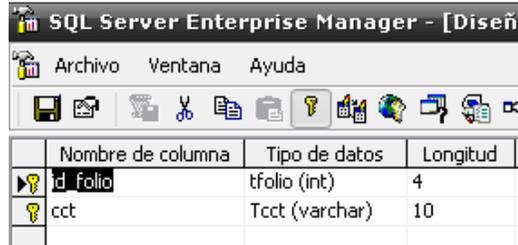
Figura 3.21. Catálogo “cat_turno”

- **CAT_USUARIOS:** Catálogo de los usuarios a los que van dirigidos los oficios para que se les den respuesta (Figura 3.22).

Nombre de columna	Tipo de datos	Longitud
idusr	varchar	15
nombre	varchar	50
pwd	varchar	16

Figura 3.22. Catálogo “cat_usuarios”

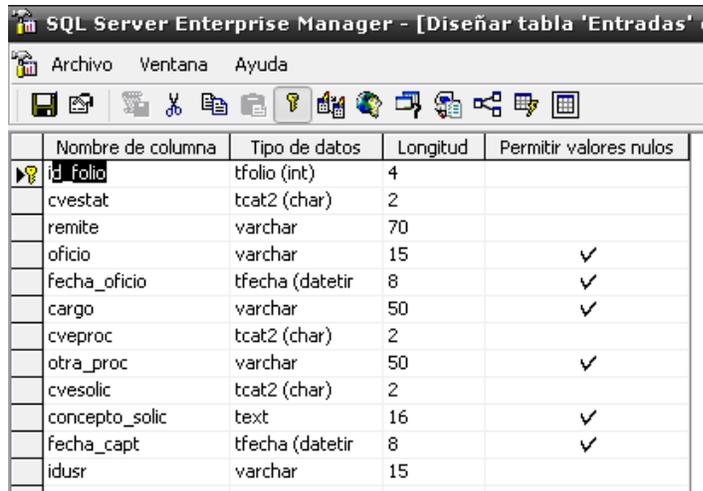
- **DET ENTRADAS:** Catálogo donde detecta por número de folio consecutivo del sistema que Clave de Centro de Trabajo (CCT) mandó el oficio (Figura 3.23).



	Nombre de columna	Tipo de datos	Longitud
PK	id_folio	tfolio (int)	4
PK	cct	Tcct (varchar)	10

Figura 3.23. Catálogo “det_entradas”

- **ENTRADAS:** Catálogo donde se almacena por un número consecutivo de folio toda la información enviada al usuario que le dará contestación a dicho oficio (Figura 3.24).



	Nombre de columna	Tipo de datos	Longitud	Permitir valores nulos
PK	id_folio	tfolio (int)	4	
	cvestat	tcat2 (char)	2	
	remite	varchar	70	
	oficio	varchar	15	✓
	fecha_oficio	tfecha (datetir)	8	✓
	cargo	varchar	50	✓
	cveproc	tcat2 (char)	2	
	otra_proc	varchar	50	✓
	cvesolic	tcat2 (char)	2	
	concepto_solic	text	16	✓
	fecha_capt	tfecha (datetir)	8	✓
	idusr	varchar	15	

Figura 3.24. Catálogo “Entradas”

- **ENTRADAS HISTORICO:** Catálogo donde se almacenó la información recuperada del sistema anterior de los oficios mandados por las diferentes instituciones de la SEP (Figura 3.25).

Nombre de columna	Tipo de datos	Longitud	Permitir valores nulos
id_folio	tfolio (int)	4	
cvestat	tcat2 (char)	2	
remite	varchar	70	
oficio	varchar	15	✓
fecha_oficio	tfecha (datetir)	8	✓
cargo	varchar	50	✓
cveproc	tcat2 (char)	2	
otra_proc	varchar	50	✓
cvesolic	tcat2 (char)	2	
concepto_solic	text	16	✓
fecha_capt	tfecha (datetir)	8	✓
idusr	varchar	15	
n_clavecct	varchar	80	✓
n_localidad	varchar	80	✓
n_municipio	varchar	60	✓
n_folio	varchar	20	✓
n_fec_folio	char	10	✓
n_fec_capt	char	10	✓

Figura 3.25. Catálogo “Entradas_histórico”

- **SALIDAS:** Catálogo donde se almacena la información capturada por el usuario encargado de dar respuesta a los oficios, la relación de lleva a cabo a través del Folio automático del sistema y el usuario (Figura 3.26).

Nombre de columna	Tipo de datos	Longitud	Permitir valores nulos
id_folio	tfolio (int)	4	
cvestat	tcat2 (char)	2	
idusr	varchar	15	
fecha_resp	tfecha (datetir)	8	✓
oficio_resp	varchar	15	✓
recibe	varchar	60	✓
cargo	varchar	50	✓
respuesta	text	16	✓
observaciones	text	16	✓
comentarios	text	16	✓
fecha_capt	tfecha (datetir)	8	✓

Figura 3.26. Catálogo “Salidas”

- **SALIDAS HISTORICO:** Catálogo donde se almacenó la información recuperada del sistema anterior de los oficios a los que se les dio respuesta (Figura 3.27).

	Nombre de columna	Tipo de datos	Longitud	Permitir valores nulos
PK	id_folio	tfolio (int)	4	
	cvestat	tcat2 (char)	2	
	idusr	varchar	15	
	fecha_resp	tfecha (datetir)	8	✓
	oficio_resp	varchar	15	✓
	recibe	varchar	60	✓
	cargo	varchar	50	✓
	respuesta	text	16	✓
	observaciones	text	16	✓
	comentarios	text	16	✓
	fecha_capt	tfecha (datetir)	8	✓

Figura 3.27. Catálogo “salidas_historico”

3.2.2.2. DIAGRAMA DE LA BD DEL SISTEMA DE CORRESPONDENCIA.

Diagrama final del sistema de correspondencia (Figura 3.28).

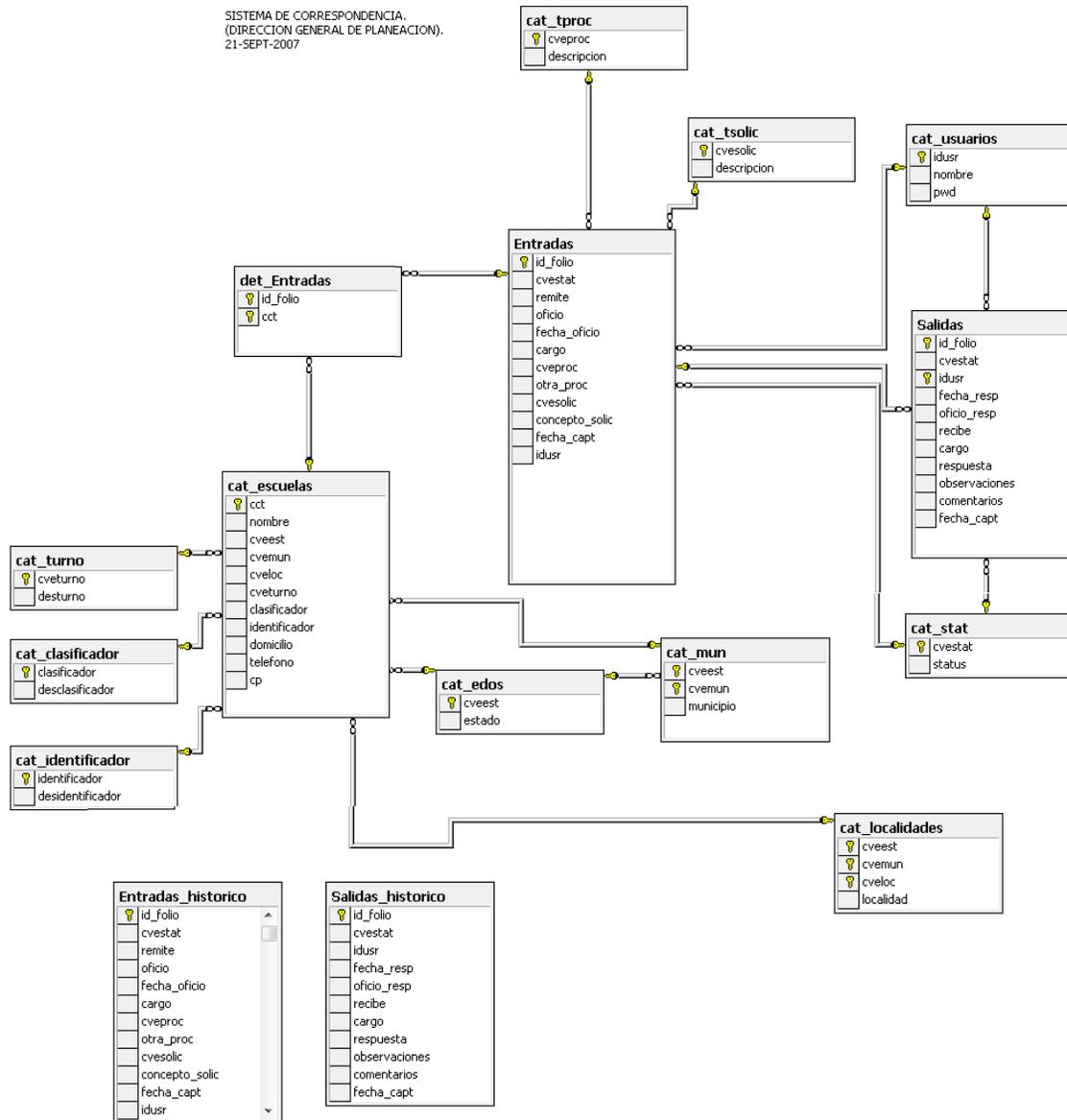


Figura 3.28. Diagrama del Sistema de Correspondencia

3.2.2.3. CREACIÓN DE VISTAS

Una vista en *SQL* es un objeto de *Base de Datos* al que se puede hacer referencia del mismo modo que a una tabla en instrucciones *SQL* mediante una instrucción *SELECT*. [4]

A continuación se describirá la definición y el como se crearon las *vistas* de la *BD* “Corresp_plan”:

- **VBÚSQUEDA:** *Vista* que contiene toda la *información* necesaria para obtener a detalle un informe requerido por el usuario (Figura 3.29).

La vista se creó de la siguiente manera:

```
CREATE VIEW dbo.vbusqueda
AS
SELECT      dbo.Entradas.id_folio, dbo.Entradas.cvestat,
dbo.cat_stat.status,
            UPPER(dbo.Entradas.remite) AS remite,
UPPER(dbo.Entradas.oficio) AS oficio
            .
            .
            .
FROM        dbo.vdet_entradas RIGHT OUTER JOIN
            .
            .
            .
            dbo.cat_stat.cvestat = dbo.Salidas.cvestat ON
            dbo.vdet_entradas.id_folio = dbo.Entradas.id_folio
```

Columna	Alias	Tabla	Resultado	Tipo
id_folio		Entradas	✓	
cvestat		Entradas	✓	
status		cat_stat	✓	
UPPER(dbo.Entradas.remite)	remite		✓	
UPPER(dbo.Entradas.oficio)	oficio		✓	
fecha_oficio		Entradas	✓	
UPPER(dbo.Entradas.cargo)	cargo		✓	
cvetproc		Entradas	✓	
descripcion	procedencia	cat_tproc	✓	
UPPER(dbo.Entradas.otra_proc)	otra_proc		✓	
cvesolic		Entradas	✓	
descripcion	tiposolicita	cat_tsolic	✓	
UPPER(CONVERT (varchar(800), dbo.Entradas.concepto_solic))	concepto_solic		✓	
fecha_capt		Entradas	✓	
idusr		Entradas	✓	
fecha_resp		Salidas	✓	
oficio_resp		Salidas	✓	
recibe		Salidas	✓	
cargo	cargo_respuesta	Salidas	✓	
respuesta		Salidas	✓	
observaciones		Salidas	✓	
comentarios		Salidas	✓	
cct		vdet_entradas	✓	
nombre		vdet_entradas	✓	
estado		vdet_entradas	✓	
municipio		vdet_entradas	✓	
localidad		vdet_entradas	✓	
desturno		vdet_entradas	✓	
desclasificador		vdet_entradas	✓	
desidentificador		vdet_entradas	✓	
domicilio		vdet_entradas	✓	
telefono		vdet_entradas	✓	
cp		vdet_entradas	✓	
nombre	usuario	cat_usuarios	✓	

Figura 3.29. Vista “vbusqueda”

- **VDET_ENTRADAS:** De la *tabla* “det_entradas” y de la *vista* “det_escuelas” se integran en una sola *vista* para incorporarlas que servirá para seleccionar de una mejor manera la clave del **CCT** en el sistema (figura 3.30).

La vista se creó de la siguiente manera:

```

CREATE VIEW dbo.vdet_entradas
AS
SELECT dbo.det_Entradas.id_folio, dbo.det_Entradas.cct,
dbo.vescuelas.nombre, dbo.vescuelas.estado,
.
.
.
FROM    dbo.det_Entradas INNER JOIN
        dbo.vescuelas ON dbo.det_Entradas.cct = dbo.vescuelas.cct
    
```

Columna	Alias	Tabla	Resultado
id_folio		det_Entrada	✓
cct		det_Entrada	✓
nombre		vescuelas	✓
estado		vescuelas	✓
municipio		vescuelas	✓
localidad		vescuelas	✓
desturno		vescuelas	✓
desclasificador		vescuelas	✓
desidentificador		vescuelas	✓
domicilio		vescuelas	✓
telefono		vescuelas	✓
cp		vescuelas	✓

Figura 3.30. Vista “vdet_entradas”

- **VENTRADAS:** *vista* donde se selecciona la *información* necesaria de distintas *tablas* para poder obtener la *información* necesaria al programar el sistema y/o para generar la impresión de los oficios necesarios. (Figura 3.31).

La *vista* se crea de la siguiente manera:

```

CREATE VIEW dbo.ventradas
AS
SELECT dbo.Entradas.id_folio, dbo.Entradas.cvestat, dbo.cat_stat.status,
UPPER(dbo.Entradas.remite) AS remite,
.
.
.
FROM    dbo.vescuelas INNER JOIN
        dbo.vdet_entradas ON dbo.vescuelas.cct = dbo.vdet_entradas.cct
        RIGHT OUTER JOIN
        .
        .
        .
        dbo.cat_usuarios cat_usuarios_1 ON dbo.Salidas.idusr =
        cat_usuarios_1.idusr
    
```

Columna	Alias	Tabla	Resultado
id_folio		Entradas	✓
cvestat		Entradas	✓
status		cat_stat	✓
UPPER(dbo.Enti	remite		✓
UPPER(dbo.Enti	oficio		✓
fecha_oficio		Entradas	✓
UPPER(dbo.Enti	cargo		✓
cveproc		Entradas	✓
descripcion	procedencia	cat_tproc	✓
UPPER(dbo.Enti	otra_proc		✓
cvesolic		Entradas	✓
descripcion	tiposolicita	cat_tsolic	✓
UPPER(CONVER	concepto_s		✓
fecha_capt		Entradas	✓
idusr		Entradas	✓
fecha_resp		Salidas	✓
oficio_resp		Salidas	✓
recibe		Salidas	✓
cargo	cargo_resp	Salidas	✓
respuesta		Salidas	✓
observaciones		Salidas	✓
comentarios		Salidas	✓
nombre		vdet_entrac	✓
estado		vdet_entrac	✓
municipio		vdet_entrac	✓
localidad		vdet_entrac	✓
desturno		vdet_entrac	✓
desclasificador		vdet_entrac	✓
desidentificador		vdet_entrac	✓
domicilio		vdet_entrac	✓
telefono		vdet_entrac	✓
cp		vdet_entrac	✓
nombre	usuario	cat_usuario:	✓
idusr	idusr_salida	Salidas	✓
nombre	usuario_sali	cat_usuario:	✓
cvemun		vescuelas	✓
cveloc		vescuelas	✓
cct		vdet_entrac	✓

Figura 3.31. Vista “ventradas”

- **VENTRADAS_HISTORICO:** Vista de donde se seleccionan los campos de la tabla “Entradas_historico” y el usuario de la tabla “cat_usuarios para poder generar el oficio dependiendo del usuario del sistema anterior (Figura 3.32).

La vista fue generada de la siguiente manera:

```
CREATE VIEW dbo.ventradas_historico
AS
SELECT
dbo.Entradas_historico.id_folio,dbo.Entradas_historico.otra_proc,
.
.
.
FROM  dbo.Entradas_historico INNER JOIN
      dbo.cat_usuarios ON dbo.Entradas_historico.idusr =
      dbo.cat_usuarios.idusr
```

Columna	Alias	Tabla	Resultado
id_folio		Entradas_hi	✓
otra_proc		Entradas_hi	✓
concepto_solic		Entradas_hi	✓
n_clavecct		Entradas_hi	✓
n_localidad		Entradas_hi	✓
n_municipio		Entradas_hi	✓
n_folio		Entradas_hi	✓
n_fec_folio		Entradas_hi	✓
n_fec_capt		Entradas_hi	✓
idusr		Entradas_hi	✓
nombre		cat_usuario:	✓

Figura 3.32. Vista “ventradas_historico”

- **VENTRADAS_SINDET:** Vista donde se une la *información* de las *tablas* “Entradas” y “Salidas” para el manejo de la *información* en el sistema y así poder generar ciertos *oficios* necesarios para el usuario (Figura 3.33).

La *vista* se creó de la siguiente manera:

```
CREATE VIEW dbo.ventradas_sindet
AS
SELECT  dbo.Entradas.id_folio, dbo.Entradas.cvestat,
        dbo.cat_stat.status, dbo.Entradas.remite,
        .
        .
        .
FROM  dbo.Entradas INNER JOIN
      dbo.Salidas ON dbo.Entradas.id_folio = dbo.Salidas.id_folio AND
      dbo.cat_stat.cvestat = dbo.Salidas.cvestat
```


Columna	Alias	Tabla	Resultado
cat		cat_escuela	✓
nombre		cat_escuela	✓
cveest		cat_escuela	✓
estado		cat_edos	✓
cvemun		cat_escuela	✓
municipio		cat_mun	✓
cveloc		cat_escuela	✓
localidad		cat_localida	✓
cveturno		cat_escuela	✓
desturno		cat_turno	✓
clasificador		cat_escuela	✓
desclasificador		cat_clasifica	✓
identificador		cat_escuela	✓
desidentificador		cat_identific	✓
domicilio		cat_escuela	✓
telefono		cat_escuela	✓
cp		cat_escuela	✓

Figura 3.34. Vista “vescuelas”

- **VMPIOS_CAT_ESC:** Vista donde se seleccionan los campos necesarios para manejar de la mejor manera la *información* de los CCT de una vista “vescuelas” catalogada por municipio (Figura 3.35).

La vista se creo de la siguiente manera:

```
CREATE VIEW dbo.vmpios_cat_esc
AS
SELECT DISTINCT TOP 100 PERCENT cvemun, municipio, clasificador,
desclasificador, identificador, desidentificador
FROM      dbo.vescuelas
ORDER BY municipio, desclasificador
```

Columna	Alias	Tabla	Resultado	Tipo de orden	Orden
cvemun		vescuelas	✓		
municipio		vescuelas	✓	Ascendente	1
clasificador		vescuelas	✓		
desclasificador		vescuelas	✓	Ascendente	2
identificador		vescuelas	✓		
desidentificador		vescuelas	✓		

Figura 3.35. Vista “vmpios_cat_esc”

- **VSALIDAS:** Vista donde se selecciona la *información* necesaria para generar los oficios de salidas con toda la *información* capturada por el usuario. Donde se explica a detalle cuales fueron las instrucciones o respuesta que se le dio a dicho oficio (Figura 3.36).

La vista se creó de la siguiente manera:

```
CREATE VIEW dbo.vsalidas
AS
SELECT dbo.Salidas.id_folio, dbo.Salidas.cvestat, dbo.cat_stat.status,
dbo.Salidas.idusr,
.
.
.
FROM    dbo.Salidas INNER JOIN
.
.
.
dbo.entradas_sindet ON dbo.Salidas.id_folio =
dbo.entradas_sindet.id_folio
```

Columna	Alias	Tabla	Resultado
id_folio		Salidas	✓
cvestat		Salidas	✓
status		cat_stat	✓
idusr		Salidas	✓
nombre		cat_usuarios	✓
fecha_resp		Salidas	✓
oficio_resp		Salidas	✓
recibe		Salidas	✓
cargo		Salidas	✓
respuesta		Salidas	✓
observaciones		Salidas	✓
comentarios		Salidas	✓
fecha_capt		Salidas	✓
remite		entradas_sindet	✓
oficio		entradas_sindet	✓
fecha_oficio		entradas_sindet	✓
procedencia		entradas_sindet	✓
otra_proc		entradas_sindet	✓
concepto_solic		entradas_sindet	✓
tiposolicita		entradas_sindet	✓
usuario_salida		entradas_sindet	✓

Figura 3.36. Vista “vsalidas”

3.2.3. DISEÑO DE PANTALLAS

3.2.3.1. PANTALLA DE ACCESO AL SISTEMA DE CORRESPONDENCIA

La pantalla de Acceso al sistema (Figura 3.37) funcionará para autorizar al usuario el acceso al Sistema de Correspondencia y saber con que privilegios contará. La *pantalla* esta compuesta por 2 *cajas de texto* Usuario y Password; 2 botones “Entrar” y “Salir”.



Figura 3.37 “Pantalla de Acceso al Sistema de Correspondencia”

3.2.3.2. PANTALLA PRINCIPAL

La pantalla principal (Figura 3.38) estará conformada por un *menú* principal que permitirá acceder a cualquier parte del sistema.



Figura 3.38. "Pantalla Principal del Sistema de Correspondencia"

3.2.3.3. PANTALLA DE ENTRADAS

La pantalla de Entradas (Figura 3.39) permitirá al usuario la captura de los oficios que lleguen al Departamento de Infraestructura Educativa de la SEP, donde se podrá anexar la(s) clave(s) del centro de trabajo (CCT) respectivas que manda dicho oficio. Además de consultar, modificar, eliminar y enviar los oficios capturados al usuario encargado de dar la respuesta.

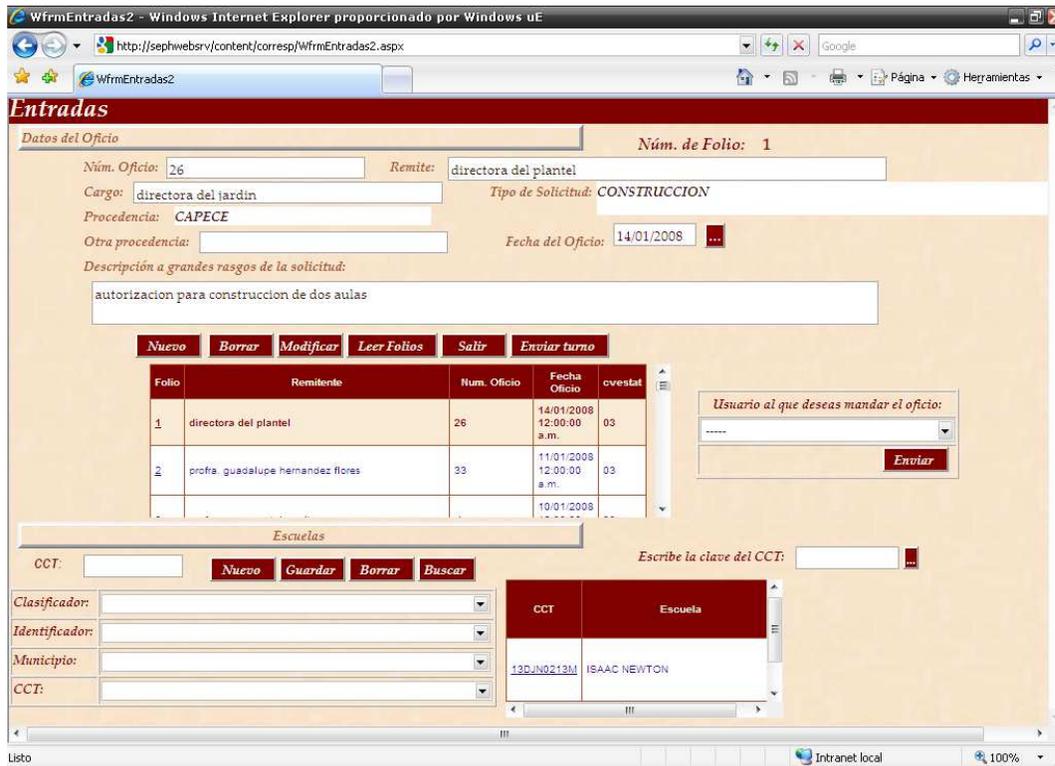


Figura 3.39. "Pantalla de Entradas"

3.2.3.4. PANTALLA DE SALIDAS

La pantalla de Salidas (Figura 3.40) permitirá al usuario la captura de respuestas de los oficios que lleguen al Departamento de Infraestructura Educativa de la SEP, haciendo la búsqueda necesaria la cuál tendrá como resultados, todos los oficios que sean mandados al usuario que este utilizando el sistema.

Figura 3.40. “Pantalla de Salidas”

3.2.3.5. PANTALLA DE CONSULTAS DE INFORMACIÓN ACTUAL

La Pantalla de Consultas de Información Actual (Figura 3.41) permitirá hacer *consultas* de la información actual, al igual que en la captura de respuestas, solo aparecerán los oficios que le correspondan al usuario que este dentro del sistema. Donde además de buscar el oficio por distintas opciones, podrá imprimir el(los) oficio(s) requeridos por el usuario.

Consulta de Entradas

Datos del Oficio

Folio: 1 Oficio: 26

Estado del oficio: RESUELTO Fecha de captura del Oficio: 14/01/2008

Remite: DIRECTORA DEL PLANTEL

Cargo: DIRECTORA DEL JARDIN

Procedencia: CAPECE Solicitud: CONSTRUCCION

Respuesta: Este depto. no tiene inconveniente en que se realicen dichos trabajos

Municipio: AJACUBA

Localidad: SANTIAGO TEZONTLALE

Escuela: ISAAC NEWTON

Usuario: CLAUDIA

Imprimir Búsqueda Nueva Búsqueda Salir

Folio	CCT	Nombre Escuela	Municipio	Localidad	Estado	
1	13DJN0213M	ISAAC NEWTON	AJACUBA	SANTIAGO TEZONTLALE	RESUELTO	26

Figura 3.41. "Pantalla de Consultas de Información Actual"

3.2.3.6. PANTALLA DE DE CONSULTAS DE INFORMACIÓN ANTERIOR

La Pantalla de Consultas de Información Anterior (Figura 3.42) permitirá hacer *consultas* de información obtenida de los formularios en dBase (del sistema de correspondencia anterior), donde se podrá también imprimir el(los) oficio(s) requeridos por el usuario. No se podrá borrar ni hacer modificaciones.

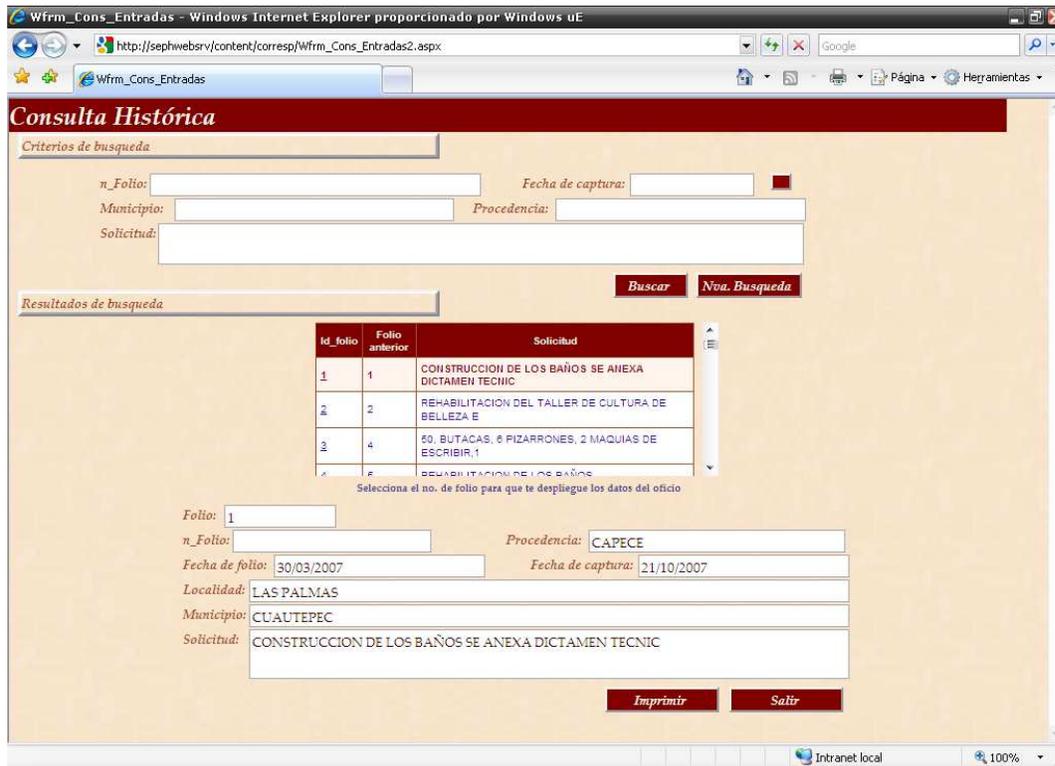


Figura 3.42. "Pantalla de Consultas de Información Anterior"

3.2.3.7. PANTALLA DE CONSULTAS DE INFORMES X CCT

La Pantalla de Consultas de Informes x CCT (Figura 3.43) permitirá realizar búsquedas por la clave del CCT (Clave del Centro de Trabajo) del que se desea saber información y tendrá la opción de imprimir el oficio solicitado por el usuario.

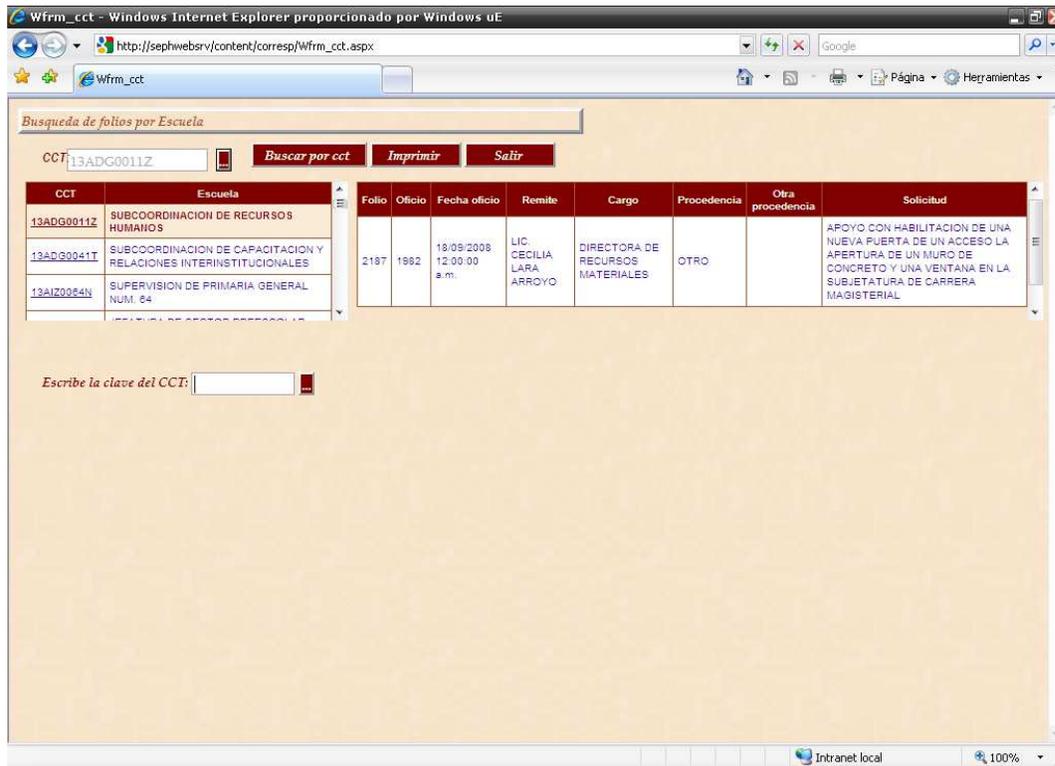


Figura 3.43. "Pantalla de Consultas de Informes x CCT"



Capítulo 4. Sistema de correspondencia

En este capítulo se hablará de cómo se realizó el Sistema de Correspondencia Actual, y donde mostrará parte del código utilizado para su funcionamiento.

4.1. SISTEMA DE CORRESPONDENCIA ACTUAL

El sistema de correspondencia, consta de una *pantalla* de acceso “Registro de Usuario” (Figura 4.1), la cual autoriza al *usuario* el acceso al sistema de correspondencia con sus respectivos privilegios anotando el nombre del usuario o cuenta y el password del mismo, pasando después al menú principal.

La *pantalla* esta compuesta por 2 *cajas de texto* “txtusr” que pertenece a la etiqueta *Usuario* y la que pertenece a la etiqueta *Password* “ltxtpsw” (que cuenta con la propiedad modo de texto *password*) que evita visualizar lo que el *usuario* escribe; 2 botones de “Entrar” (permite acceder al sistema) y “Salir” (donde se cierra la sesión por completo, ingresando a la página principal de la SEP “http://www.hgo.sep.gob.mx”).



Figura 4.1. “Pantalla de Acceso al Sistema de Correspondencia”

Código del Botón Entrar:

```
'Se arma la cadena de conexión con los valores obtenidos del web.config
y los introducidos por el usuario

sPathImg = ConfigurationSettings.AppSettings("PathImagenes")
sServer = ConfigurationSettings.AppSettings("Server")
sDB = ConfigurationSettings.AppSettings("DataBase")
sUsr = txtusr.Text
sPwd = txtpsw.Text
sDSN = "Server=" & sServer & ";Database=" & sDB & ";User=" & sUsr &
";Password=" & sPwd
    Session("DsnAcceso") = sDSN
Try
    'Se manda a llamar una función en la capa de Data_conec.vb, donde
hace una consulta en el catálogo de usuarios para ver si puede acceder al
sistema.

cPrincipal = New Cd_Corresp.Dato_conec(Session("Dsnacceso"))
Dim drCatusuarios As DataRow =
Principal.funCatusuarios(txtusr.Text.ToString)

If Not (drCatusuarios Is Nothing) Then

    'Se inicializan los objetos de session para el usuario.

    Session("Usuario") = drCatusuarios("nombre")
    Session("Login") = Trim(txtusr.Text.ToString)

    'Se autentifica al usuario y se continua con el proceso.

    FormsAuthentication.RedirectFromLoginPage(Trim(txtusr.Text), False)

Else

    'En caso contrario manda un mensaje de error

    MsgBox2.ShowMessage("El usuario no existe ó la contraseña no es
válida.")
    Exit Sub
End If
End Try
```

Una vez, dentro del sistema de correspondencia, se abre la pantalla principal, conformada por un menú. *Figura 4.2.*



Figura 4.2. "Pantalla Principal del Sistema de Correspondencia"

El *menú* Mantenimiento (Figura 4.3) permitirá capturar los oficios (Entradas) así como capturar la respuesta de los mismos (Salidas).



Figura 4.3. Menú de Mantenimiento

El *menú* Consultas (Figura 4.4) permitirá consultar los oficios actuales, además de imprimir un informe de dicho oficio (Información Actual), además de poder consultar la *información* que tenía en el sistema anterior así como imprimir un reporte del mismo, aunque cabe mencionar que la *información* recogida fue la más completa y se dejó afuera la *información* obsoleta (Anterior); así como buscar por la clave de CCT para la impresión del reporte (Consulta de Informes). Cada informe fue construido de acuerdo a los requerimientos del usuario.



Figura 4.4. Menú Consultas

El *menú* Salir permitirá cerrar la sesión actual, y así poder ingresar al sistema con una cuenta diferente.

4.1.1. PANTALLA DE ENTRADAS

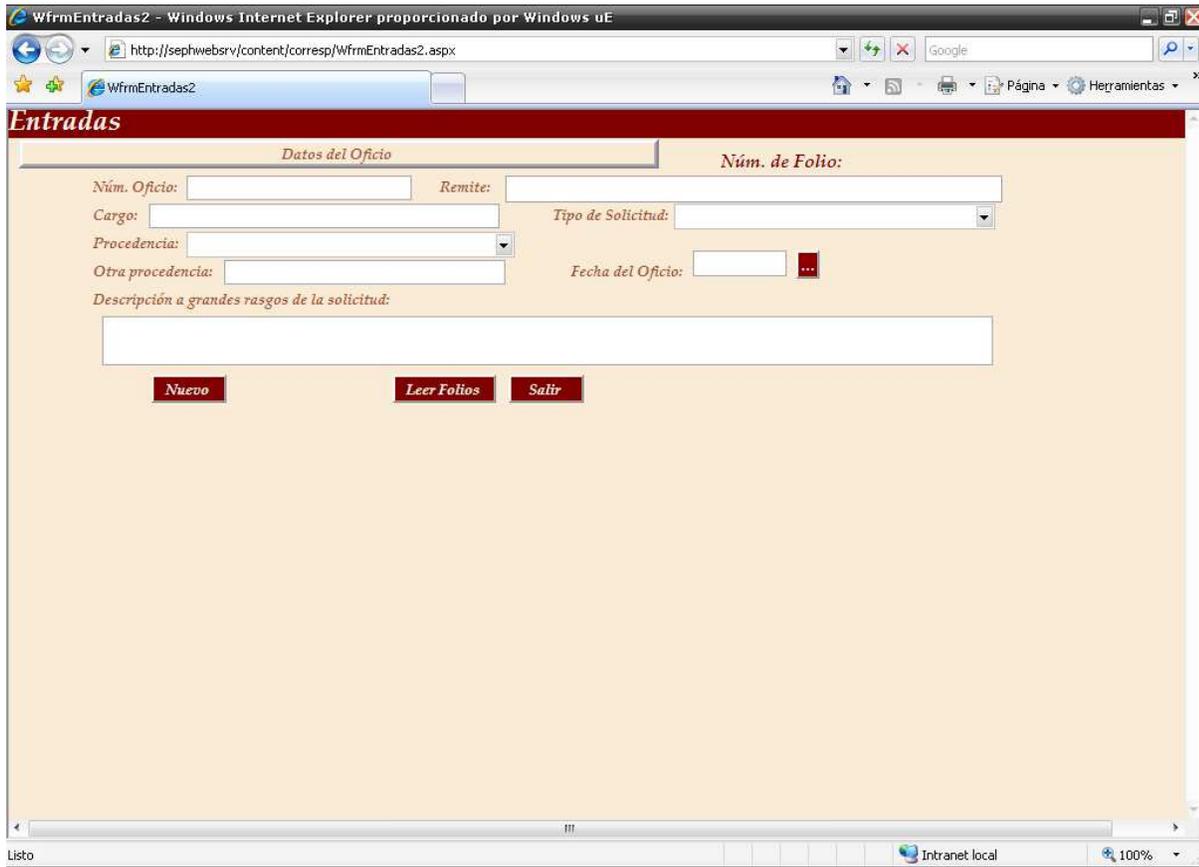


Figura 4.5. Pantalla de Entradas

Para capturar un oficio, lo primero que se tiene que hacer es darle clic en el botón nuevo (Figura 4.6.), para que genere el nuevo número de folio a grabar, dependiendo de los registros existentes.

Código del botón Nuevo:

'Se manda a llamar una función almacenada dentro de la capa de datos, la cuál realiza un conteo del no. de registros existentes, dependiendo de los campos asignados a la función.

```
Dim maxi As DataRow = cprincipal.Sel_maximo("id_folio", "entradas")
Try
    If maxi.IsNull(0) Then
```

```

`En caso de no existir ningún registro inicializa con el núm. 1
    max_reg = 1
Else
    max_reg = maxi(0) + 1
End If

```

Enseguida aparece el número máximo delante de la etiqueta Núm. de Folio

Núm. de Folio: 1, y enseguida aparece el botón de guardar (Figura 3.9)



Figura 4.6. Botón Guardar

Se captura el oficio de acuerdo a los datos que nos pide la pantalla de Entradas.

Datos del Oficio:

- *Num. Oficio:* Caja de texto, de tipo *varchar*. Donde se captura el número del oficio. Ejemplo: SPG-023831 ó 35.
- *Remite:* Caja de texto, de tipo *varchar*. Donde se captura el nombre de quien remite el oficio.
- *Cargo:* Caja de texto, de tipo *varchar*. Se captura el cargo de la persona que remite el oficio.
- *Tipo de Solicitud:* combo de selección, de tipo *tchar2* (definido por un programador, de tipo *char*, longitud 2, y que no permite valores nulos). Donde se selecciona el tipo de solicitud: construcción, equipamiento, rehabilitación, mobiliario, construcción y equipamiento, sin especificar y otro, de acuerdo al catalogo de solicitudes (*cat_tsolic*). El cuál se llena con el siguiente *código*:

`Se manda a llamar una función desde la capa de datos "Cd_corresp"

```

cbo_solicitud.DataSource = cprincipal.SelCatCboStr("cvesolic",
"descripcion", "cat_tsolic")

```

```

cbo_solicitud.DataValueField = "cvesolic"
cbo_solicitud.DataTextField = "descripcion"
cbo_solicitud.DataBind()

```

```

Función SelCatCboStr
...
Dim Ssql As String = ("SELECT " & Clave & "'00'," & descripcion & "'---
--' union all SELECT " & Clave & "," & descripcion & " FROM " & tabla & "
WHERE " & Clave & "<> '99'")
...
End Function

```

- **Procedencia:** combo de selección, de tipo *tchar2*. Donde se selecciona la procedencia del oficio: CAPECE, escuela, presidencia municipal, SEDESO, sin especificar, otro. El código, es el siguiente:

```

cbo_procedencia.DataSource = cprincipal.SelCatCboStr("cveproc", "descripcion",
"cat_tproc")
cbo_procedencia.DataValueField = "cveproc"
cbo_procedencia.DataTextField = "descripcion"
cbo_procedencia.DataBind()
cbo_procedencia.Enabled = True

```

- **Otra procedencia:** en caso de seleccionar otro en procedencia, en esta caja de texto, de tipo *varchar*, se captura la procedencia del oficio. Ejemplo: liga de comunidades agrarias, casas quuma, etc.
- **Fecha del Oficio:** caja de texto, de tipo *datetime*. El cuál se llena eligiendo la fecha de un calendario (Figura 4.7), el cuál aparece al darle clic a la siguiente figura: 
- **Descripción a grandes rasgos de la solicitud:** caja de texto, de tipo *Text*. Donde se captura a detalle la solicitud del oficio.



Figura 4.7. Selección de la fecha a través de un calendario

Una vez, capturado todo, se procede a guardar la información, dándole clic en el botón guardar, el cuál su código es:

'Se verifica que de acuerdo a los criterios, las cajas que son de importancia llenar, no se queden vacías.

```
If Len(txt_remite.Text) = "0"
...
Or Len(txt_concepto.Text) = "0" Then
    MsgBox1.ShowMessage("Verifica tus datos")
    Exit Sub
End If
```

'Función Ins_Entradas de la capa de datos Cd_corresp.

```
Dim Ssql As String = "Insert Entradas values (" & id_folio & ", '00', '" &
Trim(remite) & "', '" & Trim(oficio) & ...
Trim(cvesolic) & "', '" & Trim(concepto_solic) & "', getdate(), '" &
Trim(idusr) & "')
```

Al guardar la información en la tabla “Entradas”, aparece el botón “Leer Folios”, tal cuál lo muestra la Figura 4.8.



Figura 4.8. Botón leer folios

Hay oficios que no son enviados por alguna escuela en particular, pero la mayor parte de las solicitudes, son por planteles educativos.

Para anexar el nombre de una escuela en particular, debemos darle clic en el botón “Leer Folios”, el cuál abre un *grid* con los datos del oficio (Figura 4.9), como se muestra a continuación:

Folio	Remitente	Num. Oficio	Fecha Oficio	cvestat
1	directora del plantel	26	14/01/2008 12:00:00 a.m.	03
2	profra. guadalupe hernandez flores	33	11/01/2008 12:00:00 a.m.	03
			10/01/2008	

Figura 4.9. Datos del Oficio desde un grid.

Al seleccionar un oficio (clic sobre el número de folio), abrirá un *panel* con etiquetas y *combos de selección*, además de una *caja de texto*, donde una vez seleccionada la escuela, nos aparecerá la clave del *CCT* seleccionado, además de los botones de nuevo, guardar, borrar y buscar. Figura 4.10.

Figura 4.10. Anexado de Escuelas

Para ingresar una clave *CCT*, lo primero que se deberá hacer, es darle clic en el botón nuevo, para que el combo del clasificador se inicialice con el siguiente código:

```
cbo_nivel.DataSource = cprincipal.SelCatCboClas()
cbo_nivel.DataValueField = "clasificador"
cbo_nivel.DataTextField = "desclasificador"
cbo_nivel.DataBind()
cbo_nivel.Enabled = True
```

Seleccionando el clasificador, se activa el combo de identificador, tomando en cuenta lo que se selecciona en el combo del clasificador, para hacer una búsqueda más exacta, el *combo* se llena de la siguiente manera:

```
cbo_ident.DataSource = cprincipal.SelCatCboIden(cbo_nivel.SelectedValue)
cbo_ident.DataValueField = "identificador"
cbo_ident.DataTextField = "desidentificador"
cbo_ident.DataBind()
cbo_ident.Enabled = True
```

Al seleccionar el identificador, procedemos a llenar el combo del municipio, en el llenado se toma en cuenta el clasificador y el identificador seleccionados anteriormente:

```
cbo_mun.DataSource = cprincipal.SelCatCboMun(cbo_nivel.SelectedValue,
cbo_ident.SelectedValue)
cbo_mun.DataValueField = "cvemun"
cbo_mun.DataTextField = "municipio"
cbo_mun.DataBind()
cbo_mun.Enabled = True
```

Y para finalizar al seleccionar el municipio, se llena el *combo* del *CCT*, donde se hace una búsqueda, de acuerdo a lo seleccionado en los *combos* anteriores:

```
cbo_cct.DataSource = cprincipal.SelCatCboCct(cbo_nivel.SelectedValue,
cbo_ident.SelectedValue, cbo_mun.SelectedValue)
cbo_cct.DataValueField = "cct"
cbo_cct.DataTextField = "nombre"
cbo_cct.DataBind()
cbo_cct.Enabled = True
```

La función utilizada es la siguiente:

```
Public Function SelCatCboCct(ByVal clasificador As String, ByVal
identificador As String, ByVal municipio As String) As DataTable

Dim Ssql As String = "SELECT cct='000',nombre='-----' union all select
cct, nombre from Vescuelas"
...
Ssql = Ssql & " and cvemun ='" & Trim(municipio) & "'"
```

Al seleccionar la escuela, la clave del *CCT* aparece en la *caja de texto*

CCT:

, la cual no podemos modificar por seguridad a que el usuario no se equivoque en la escritura del *CCT*. Procedemos a integrar la

escuela al folio seleccionado, dando clic en el botón , el cuál lo hace a través de la instrucción:

```
cprincipal.Ins_EntradasCct(lblid_folio.Text, txt_cct.Text)
```

Función "Ins_EntradasCct"

```
Dim Ssql As String = "Insert det_Entradas values (" & id_folio & ",'" &
Trim(cct) & "'")"
```

Al insertar la clave del CCT, pasa a llenarse un *grid* (4.11), el cual se llena con la(s) escuela(s) de dicho oficio:

CCT	Escuela
13DJN0213M	ISAAC NEWTON

Figura 4.11. Grid con la información del CCT

Función de llenado del *grid*:

```
Dim Ssql As String = "Select cct, nombre from vdet_entradas where id_folio = '" & Trim(id_folio) & "'"
```

Para dar de baja una escuela del oficio capturada, se selecciona la clave del CCT ubicada dentro del *grid*, y se da clic en el botón de **Botón** el cual manda a llamar a la siguiente *función*:

```
Dim Ssql As String = "Delete Entradas Where id_folio = " & id_folio
```

Para realizar una búsqueda de CCT, basta con darle clic en el botón **Buscar**, el cuál hace visible una etiqueta, una *caja de texto* y un *botón*, en la *caja de texto* se puede anotar, parte de una clave CCT y al darle clic en el *botón*, desglosará en un *grid* (Figura 4.12) todas las escuelas que tengan parte de la clave, escrita en la *caja de texto*, como se muestra a continuación:

CCT	Escuela
13DJN1000R	JEDZEELEX
13DJN1001Q	MANUEL CARBAJAL
13DJN1002P	REFUGIO INFANTIL
13DJN1003O	UETSKA KONEME

Figura 4.12. Búsqueda de CCT

La instrucción con la que busca todas las CCT parecidas a lo que se escribe en la *caja de texto* es:

```
Dim Ssql As String = "Select * from cat_escuelas where cct like '%" & cct & "%' order by cct"
```

Al seleccionar la clave del CCT, automáticamente aparece en la *caja de texto* CCT, donde podemos anexar el plantel al oficio seleccionado inicialmente.

4.1.3. PANTALLA DE SALIDAS

Pantalla inicial de Salidas (Figura 4.13).

The screenshot shows a web browser window titled "Salidas" with the URL "http://sephwebsrv/content/corresp/WfrmSalidas.aspx". The page has a red header with the word "Salidas" in white. Below the header, there is a search section titled "Opciones de búsqueda:" with four input fields: "No. Folio:", "Fecha del oficio:", "Procedencia:", and "Tipo de solicitud:". To the right of these fields are two buttons: "Buscar" and "Nva. Búsqueda". Below this is a section titled "Oficios para Contestar:" which is currently empty. At the bottom of the page, there is a form with several input fields: "Núm. Oficio:", "No. del Oficio:", "Fecha de Respuesta:", "Oficio de Respuesta:", "Estado del oficio:", "A quien se enviar Respuesta:", "Cargo:", "Respuesta:", "Observaciones:", and "Comentarios:". At the bottom right of this form are three buttons: "Grabar", "Cancelar", and "Salir". The browser's status bar at the bottom shows "Listo" and "Intranet local".

Figura 4.13. Pantalla Principal de Salidas

Para dar respuesta a un oficio, lo que se tiene que hacer es irse al *menú* principal, y en la parte de mantenimiento, darle clic en Salidas, donde se tiene 2 apartados: *Opciones de búsqueda* y *Oficios para contestar*.

En el primer apartado existen 4 opciones de búsqueda de un oficio, ya sea por el No. Folio, Fecha del Oficio, Procedencia y Tipo de solicitud, el *botón* Buscar y Nueva Búsqueda.

Para buscar un oficio, es necesario escribir en las *cajas de texto* las palabras claves para buscar dicho oficio o en la opción *Fecha del Oficio* seleccionar alguna fecha del calendario que aparece al darle clic en el *botón* 

(como ya lo hemos visto anteriormente) y darle clic en el *botón* . El cuál manda a llamar la siguiente función:

'La función se encarga de validar las cajas de texto, si tienen información, va creando una cadena con la información capturada por el usuario para realizar la búsqueda con dicha información.

```
Public Function busca_salidas (ByVal idusr As String, ByVal folio As String, ByVal fecha As String, ByVal solicitud As String, ByVal procedencia As String) As DataTable

Dim ban As Integer
Try
ban = False
Dim Ssql As String = "Select * from vsalidas"
If folio <> 0 Then
    Ssql = Ssql & " where id_folio = " & folio
    ban = True
End If
...
If (folio = 0) Or (fecha = "0") Or (solicitud = "0") Or (procedencia = "0") Then
    If ban Then
        Ssql = Ssql & " and "
    Else
        Ssql = Ssql & " where "
    End If
ban = True
Ssql = Ssql & " idusr = '" & idusr & "' order by id_folio"
End If
    cEjecucion.CrearCursor(Ssql, strConnection)
    Return cEjecucion.Records
End Function
```

Una vez realizada la consulta, aparecerá en un *grid*, todos los oficios que contengan la *información* capturada por el usuario, tal cual lo muestra la figura 4.14.

The screenshot shows a web interface titled "Salidas". At the top, there is a section "Opciones de búsqueda:" with several input fields: "No.Folio:" containing "20", "Fecha del oficio:" (empty), "Procedencia:" containing "SEDESO", and "Tipo de solicitud:" (empty). Below these are two buttons: "Buscar" and "Nva. Busqueda". Underneath is a section "Oficios para Contestar:" with a table of results. The table has the following data:

Folio	Remite Oficio	Núm. Oficio	Fecha Of.	Procedencia	Otra Proc.	Tipo Solicitud	fecha_resp	oficio_resp
20	Lic. mario alberto cuatepotzo	42	10/12/2007 12:00:00 a.m.	SEDESO		OTRO		

Below the table is a horizontal scrollbar and a note: "Selecciona el no. de folio para darle respuesta al oficio".

Figura 4.14. Resultado de la búsqueda

Si se requiere realizar otra búsqueda, es necesario dar clic en el *botón*

Nva. Busqueda

el cuál se encarga de limpiar las *cajas de texto* y el *grid*.

Para dar respuesta a un documento u oficio, es necesario, seleccionar el número del asunto del *grid* y automáticamente aparecerá el número del folio delante de la etiqueta Num. Oficio, como lo muestra la figura 4.15.

Salidas

Opciones de búsqueda:

No.Folio: Fecha del oficio:

Procedencia: Tipo de solicitud:

Oficios para Contestar:

Folio	Remite Oficio	Núm. Oficio	Fecha Of.	Procedencia	Otra Proc.	Tipo Solicitud	fecha_resp	oficio_resp
20	Lic.mario alberto cuatepotzo	42	10/12/2007 12:00:00 a.m.	SEDESO		OTRO		

Selecciona el no. de folio para darle respuesta al oficio

Núm.Oficio: Fecha de Respuesta:

Figura 4.15. Aparición del número de Oficio

Una vez visto el número del oficio, se procede a llenar las *cajas de texto* y el *combo* de la sección *Oficios para Contestar*. Como lo son:

- **Fecha de Respuesta:** *Caja de texto*, de tipo *tfecha* (datetime). Donde se captura la fecha de respuesta del oficio al seleccionarla de un calendario.
- **Oficio de Respuesta:** *Caja de texto*, de tipo *varchar*. Donde se captura el número del oficio de respuesta.
- **Estado del Oficio:** *Combo de Selección*, de tipo *tcat2* (char). Se selecciona el estado del Oficio, ya sea: *Para conocimiento*, *En trámite*, *Improcedente*, *Resuelto*, *Sin especificar*.
- **A quien se envía Respuesta:** *Caja de texto*, de tipo *varchar*. Aquí se captura el nombre de la persona a quien va dirigido el documento.
- **Cargo:** *Caja de texto*, de tipo *varchar*. Donde se escribe el cargo de la persona a quien va dirigido el documento.
- **Respuesta:** *Caja de texto*, de tipo *Text*. Donde se escribe la respuesta que le dieron al oficio.
- **Observaciones:** *Caja de texto*, de tipo *Text*. Se escriben las observaciones del oficio.
- **Comentarios:** *Caja de texto*, de tipo *Text*. Se escriben los comentarios que se tengan del oficio.

Para grabar lo modificar a respuesta a un documento, es necesario pulsar el botón de  el cual esta programado de la siguiente manera:

```
'Se hace una validación de las cajas de texto para que si alguna de ellas  
esta vacía, mande un mensaje de error para que se capture dicha  
información.
```

```
If Len(lblid_folio.Text) = "0"  
    ...  
Or Trim(cbo_status.Selectedvalue) = "00" Then  
    MsgBox1.ShowMessage("Verifica tus datos")  
    Exit Sub  
End If
```

```
'Manda a llamar una función la cual me hace una modificación en la tabla  
de Salidas llamada "Upd_Salidas"
```

```
cprincipal.Upd_Salidas(txt_Fechresp.Text, txt_ofresp.Text,  
txtrecibe.Text, txtcargo.Text, txtrespuesta.Text, txtobservaciones.Text,  
txtcomentarios.Text, cbo_status.Selectedvalue, lblid_folio.Text,  
Session("Login"))
```

Cuando uno esta escribiendo una respuesta nueva a un documento o simplemente haciendo una modificación, se podrá borrar lo escrito con darle clic en el botón  el cuál se encarga solo de limpiar las *cajas de texto*.

Para salir de la pantalla Salidas, es necesario darle clic en el botón



el cuál nos regresa al *menú* principal.

4.1.3. PANTALLA DE INFORMACIÓN ACTUAL

Pantalla inicial de Información Actual (Figura4.16).

Wfrm_Cons_Entradas - Windows Internet Explorer proporcionado por Windows uE

http://sephwebsrv/content/corresp/Wfrm_Cons_Entradas.aspx

Google

Wfrm_Cons_Entradas

Página Herramientas

Consulta de Entradas

Datos del Oficio

Folio: Oficio:

Estado del oficio: Fecha de captura del Oficio:

Remite:

Cargo:

Procedencia: Solicitud:

Respuesta:

Municipio:

Localidad:

Escuela:

Usuario:

Busqueda Nueva Busqueda Salir

Listo Intranet local 100%

Figura 4.16. Pantalla de Información Actual

Para consultar un oficio de la información actual (oficios que se han dado de alta en el sistema desde su implantación en el Departamento de Infraestructura Educativa de la SEP), lo primero que se tiene que hacer es darle clic en el menú principal y seleccionar la opción de *Información Actual* la cuál nos envía a su respectiva pantalla (Figura 3.19.), donde se puede buscar e imprimir el/los oficios necesarios con toda la *información* requerida por el usuario. Para realizar una consulta de un/os oficios, es necesario, llenar cualquiera de las siguientes *cajas de texto* con la *información* necesaria con la cual se desea realizar la búsqueda:

- **Folio:** *Caja de Texto*, de tipo *tfolio* (int). Llenar esta caja de texto, es conveniente cuando se desea buscar un oficio por su num. de folio cuando este se sabe.
- **Oficio:** *Caja de texto*, de tipo *varchar*. Aquí se captura el número del oficio que se desea buscar.
- **Estado del Oficio:** *Combo Box*, de tipo *tcat2* (char). En el cual se selecciona el estado de los oficio(s) a buscar. El combo se llena a través del siguiente código:

'Se manda a llamar una función la cual hace la consulta de una vista llamada "ventradas", donde se tiene toda la información de todos los oficios que se dan de alta en el sistema, y donde se toma solo los estados de los oficios existentes.

```
SELECT cvestat='00',descripcion='-----' union all SELECT DISTINCT  
cvestat,status FROM ventradas ORDER BY cvestat
```

- **Fecha de captura del oficio:** *Caja de texto*, de tipo *tfecha* (datetime). Aquí se captura la fecha de captura del oficio a buscar, el cuál se selecciona de un calendario prediseñado por el contorno de *visual .net 2003*.
- **Remite:** *Caja de texto*, de tipo *varchar*. Si se gusta buscar un oficio por este campo, es necesario escribir el nombre de la persona que elaboro el oficio.
- **Cargo:** *Caja de texto*, de tipo *varchar*. Aquí se escribe el cargo de la persona que mando el oficio.
- **Procedencia:** *Combo Box*, de tipo *tcat2* (char). Se selecciona el nombre de la procedencia del oficio. El cual se llena haciendo una consulta de la *vista* de "ventradas" de los oficios que ya se han dado de alta.
- **Solicitud:** *Combo Box*, de tipo *tcat2* (char). Se selecciona el nombre de la solicitud con la que se desea buscar el(los) oficio(s), el *combo* se llena a través de una función que realiza una consulta a la *vista* "ventradas" en donde obtiene solo las solicitudes existentes a la hora de dar de alta un oficio.
- **Respuesta:** *Caja de texto*, de tipo *text*. La cual se llena con tan solo una(as) palabras que se escribieron en la respuesta al oficio que se esta buscando.
- **Municipio:** *Combo Box*, de tipo *tcat3* (char). Se selecciona el municipio del que proviene el(los) documento(s) a buscar. El cual se llena del catalogo "Cat_mun"
- **Localidad:** *Combo Box*, de tipo *tcat4* (char). Se selecciona la localidad del que proviene el(los) documento(s) a buscar. El cual se llena del catálogo "cat_localidades".
- **Escuela:** *Caja de texto*, de tipo *varchar*. Se escribe el nombre de la escuela de donde proviene el(los) documento(s). El cual se puede escribir con el nombre de la primera palabra de la escuela, ya que la búsqueda se hace

por la palabra *like* (busca todos los registros de determinado campo que contengan la palabra de la caja de texto).

- *Usuario: Combo Box*, de tipo *varchar*. El cuál se llena del catálogo "Cat_usuarios". Con todos los usuarios existentes.

Para realizar la búsqueda de un documento, basta con seleccionar el *botón*

Busqueda

y el sistema, hace una validación de las *cajas de texto* con el siguiente *código* para concatenar los criterios de búsqueda:

```
Private Sub btn_leer_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btn_leer.Click
```

```
    Dim folio As Integer
    Dim status As String
    Dim oficio As String
    Dim fechaof As String
    Dim remite As String
    Dim cargo As String
    Dim procedencia As String
    Dim solicitud As String
    Dim respuesta As String
    Dim esc As String
```

...

```
    Try
        If (Len(Trim(txt_folio.Text)) > 0) Then
            folio = txt_folio.Text
        End If

        If (Len(Trim(cbo_status.SelectedValue)) = "00") Then
            status = "00"
        Else
            status = cbo_status.SelectedValue
        End If

        If (Len(Trim(txt_oficio.Text)) > "0") Then
            oficio = txt_oficio.Text
        End If
```

...

```
    If cbo_usu.SelectedValue = "" Then
        txt_usu.Text = "00"
    Else
        txt_usu.Text = cbo_usu.SelectedValue
    End If
```

'Función que lee:

```
Dim dtcal As DataTable = cprincipal.busca_todo(folio, status, oficio,  
fechaof, remite, cargo, procedencia, solicitud, respuesta, txt_mun.Text,  
txt_loc.Text, esc, txt_usu.Text)
```

```
    If Not dtcal Is Nothing Then  
        If dtcal.Rows.Count > 0 Then  
            grd_foliosE.DataSource = dtcal  
            grd_foliosE.DataBind()  
            grd_foliosE.Visible = True  
        Else  
            grd_foliosE.Visible = False  
        End If
```

...

```
End Sub
```

'Función busca_todo:

```
Public Function busca_todo(ByVal folio As String, ByVal status As String,  
ByVal oficio As String,
```

...

```
ByVal usu As String) As DataTable  
Dim ban As Integer
```

```
    Try  
        ban = False  
        Dim Ssql As String = "Select * from ventradas"  
        If folio <> 0 Then  
            Ssql = Ssql & " where id_folio = " & folio  
            ban = True  
        End If
```

```
        If status <> "00" Then  
            If ban Then  
                Ssql = Ssql & " and "  
            Else  
                Ssql = Ssql & " where "  
            End If  
            ban = True  
            Ssql = Ssql & " cvestat = " & status  
        End If
```

...

```
        Ssql = Ssql & " order by id_folio"
```

```
        cEjecucion.CrearCursor(Ssql, strConnection)  
        Return cEjecucion.Records  
    Catch ex As Exception  
        Throw ex  
    End Try  
End Function
```

Al realizar la búsqueda, se llena un *grid* (Figura 4.17), donde se colocan todos los oficios con los criterios de búsqueda capturados por el usuario:

Folio	CCT	Nombre Escuela	Municipio	Localidad	Estado	
1	13DJN0213M	ISAAC NEWTON	AJACUBA	SANTIAGO TEZONTLALE	RESUELTO	26

Figura 4.17. Grid con la información de los criterios de búsqueda

Una vez, llenado el *grid*, basta con seleccionar uno de los documentos, para que aparezca el botón de imprimir, y la información del mismo, en cada una de las *cajas de texto*, tal como lo muestra la figura 4.18.

Consulta de Entradas

Datos del Oficio

Folio: 1 Oficio: 26

Estado del oficio: RESUELTO Fecha de captura del Oficio: 14/01/2008

Remite: DIRECTORA DEL PLANTEL

Cargo: DIRECTORA DEL JARDIN

Procedencia: CAPECE Solicitud: CONSTRUCCION

Respuesta: Este depto. no tiene inconveniente en que se realicen dichos trabajos

Municipio: AJACUBA

Localidad: SANTIAGO TEZONTLALE

Escuela: ISAAC NEWTON

Usuario: CLAUDIA

Imprimir Busqueda Nueva Busqueda Salir

Folio	CCT	Nombre Escuela	Municipio	Localidad	Estado	
1	13DJN0213M	ISAAC NEWTON	AJACUBA	SANTIAGO TEZONTLALE	RESUELTO	26

Figura 4.18. Pantalla de Consulta de Entradas con información seleccionada de un grid.

Al mandar a imprimir el documento, la instrucción del botón es abrir otra pantalla del sistema, con el parámetro del número de folio que se desea imprimir:

```
Response.Redirect("Wfrm_impresion.aspx?id_folio=" & txt_folio.Text)
```

Al abrir la pantalla de "Wfrm_impresion.aspx" en la parte de page load, se llama el número del oficio, con el siguiente código:

```
Dim id_folio As String = Request.QueryString("id_folio")
Dim da As New SqlClient.SqlDataAdapter("select * from ventradas WHERE
id_folio=" + id_folio, oConn)
```

Al realizar la búsqueda muestra el documento de la siguiente forma (Figura 4.19):

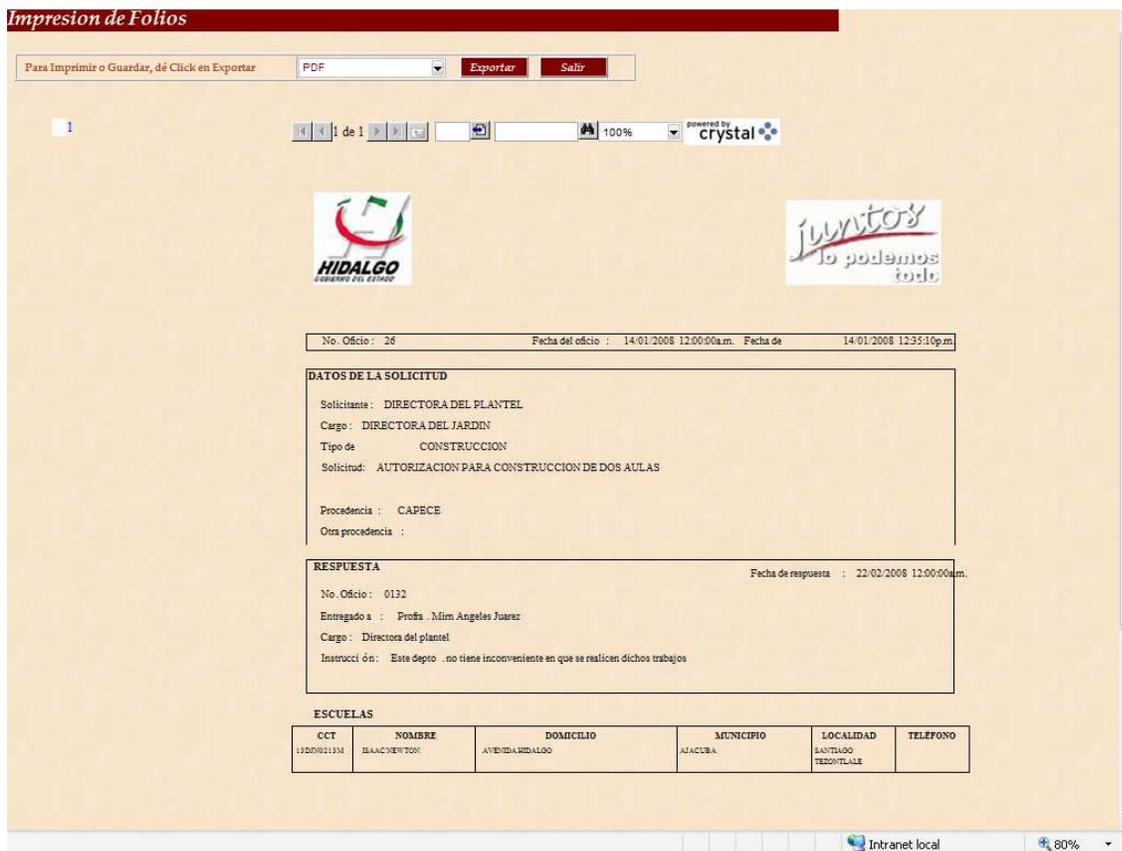


Figura 4.19. Informe de Consultas de Salida

Para imprimir el documento, hay que exportarlo en cualquiera de los formatos que se muestra en el *combo* de la Figura 4.20.

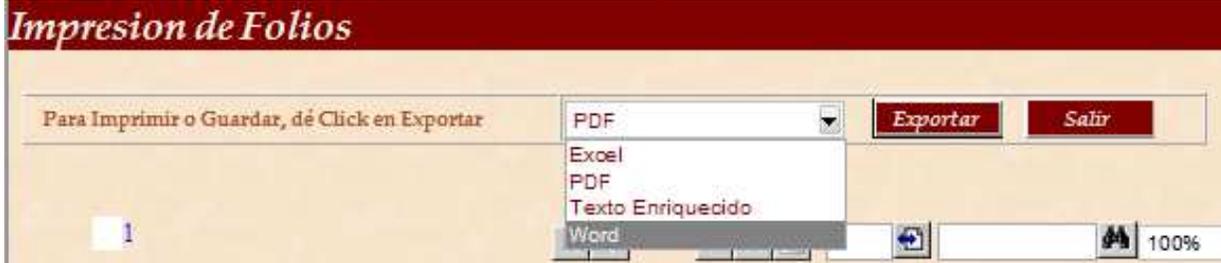


Figura 4.20. Exportación del oficio.

Y al darle clic en el *botón* de Exportar, abre un cuadro de diálogos. Figura 4.21.

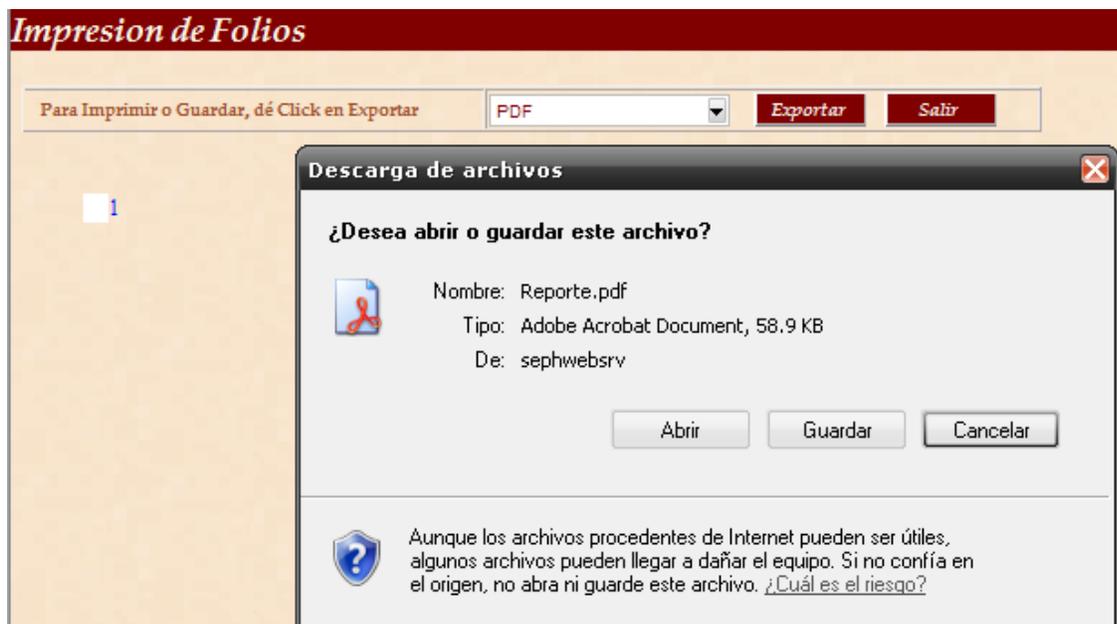


Figura 4.21. Cuadro de diálogos para exportar

Y a continuación abre el archivo (Figura 4.22) de la siguiente manera dependiendo del formato que se haya elegido (en este ejemplo formato: PDF):



Figura 4.22. Oficio exportado en formato PDF

4.1.4. PANTALLA DE INFORMACIÓN ANTERIOR

Pantalla inicial de Información Anterior (Figura 4.23).

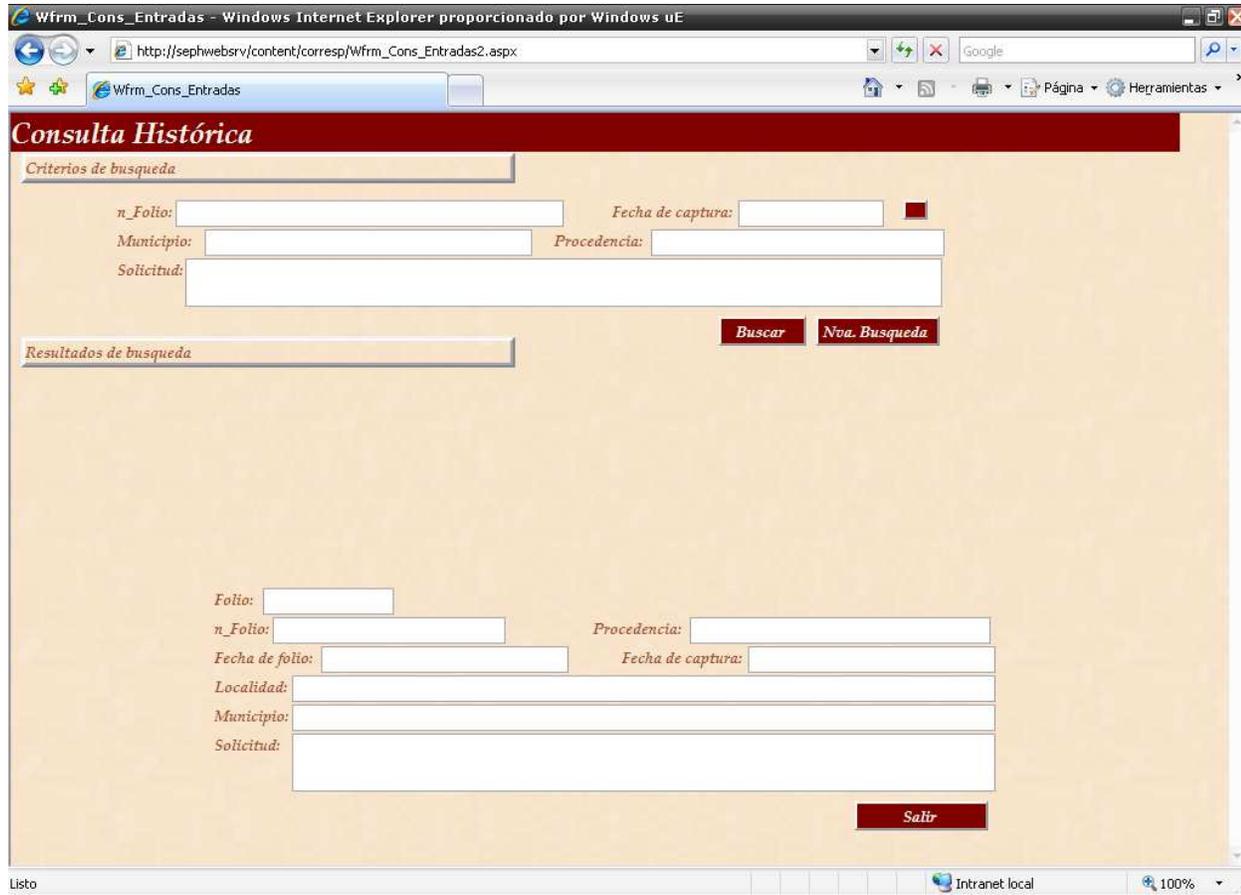


Figura 4.23. Pantalla de Información Anterior

Para consultar un oficio de la *información* anterior (oficios que se dieron de alta en el sistema anterior hecho en Dbase), lo primero que se tiene que hacer es darle clic en el menú principal y seleccionar la opción de Información Anterior la cuál nos envía a su respectiva pantalla, donde se puede buscar e imprimir el/los oficios necesarios con toda la *información* requerida por el usuario. Para realizar una consulta de un/os oficios, es necesario, llenar cualquiera de las siguientes *cajas de texto* con la *información* necesaria con la cual se desea realizar la búsqueda:

- *n_Folio*: Caja de Texto, de tipo *tfolio* (int). Llenar esta caja de texto, es conveniente cuando se desea buscar un oficio por el num. de folio del sistema anterior.
- *Fecha de Captura*: Caja de texto, de tipo *tfecha* (datetime). Aquí se captura la fecha en que se capturo el oficio en el sistema anterior.
- *Municipio*: Caja de texto, de tipo *varchar*. Se captura el nombre del municipio del(los) oficio(s) que se desean buscar.
- *Procedencia*: Caja de texto, de tipo *varchar*. Se captura el nombre de la procedencia del(los) oficio(s) a buscar.
- *Solicitud*: Caja de texto, de tipo *text*. Se captura el concepto de la solicitud de(los) oficio(s) buscar.

Al darle clic en el botón  automáticamente el sistema valida las *cajas de texto* y dependiendo de ello, manda a llamar la siguiente función, y estructura las opciones de búsqueda:

```
Public Function busca_historico(ByVal folio As String, ByVal municipio As
String, ByVal fecha As String, ByVal solicitud As String, ByVal
procedencia As String) As DataTable

    Dim ban As Integer
    Try
        ban = False
        Dim Ssql As String = "Select * from ventradas_historico"
        If folio <> 0 Then
            Ssql = Ssql & " where n_folio = '" & folio & "'"
            ban = True
        End If
        ...
        Ssql = Ssql & " order by id_folio"
        cEjecucion.CrearCursor(Ssql, strConnection)
        Return cEjecucion.Records
    Catch ex As Exception
        Throw ex
    End Try
End Function
```

Automáticamente, se llena un *grid* (Figura 4.24.), donde aparecen el(los) oficios encontrados de acuerdo a los criterios de búsqueda:

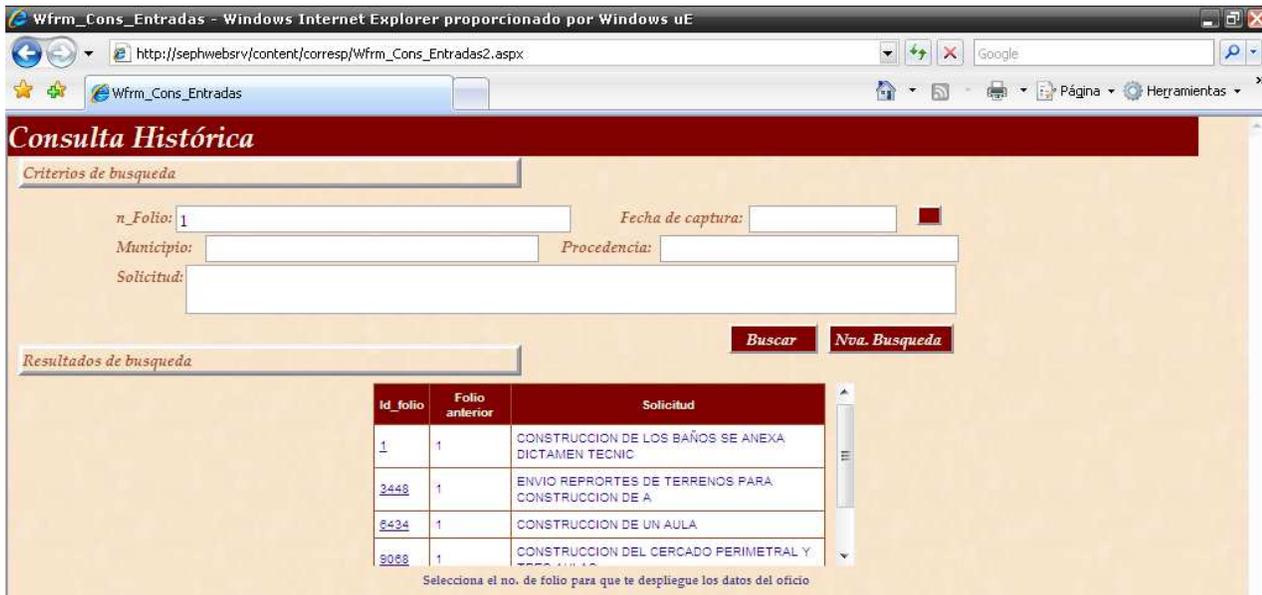


Figura 4.24. Grid en la pantalla de Información Anterior.

Y al seleccionar el oficio deseado de la *información* se despliega en la *cajas de texto* de la página, como lo vemos en la Figura 4.25.



Figura 4.25. Despliegue de la información.

Si se desea realizar otra búsqueda con otros criterios, basta con seleccionar el botón **Nva. Búsqueda** el cual limpia las *cajas de texto* y el *grid* lo limpia y lo hace invisible.

Para imprimir el oficio, basta con seleccionar el *botón* el cual abre otra pantalla del sistema “Wfrm_impresion_anterior”, con el parámetro del número de folio que se desea imprimir:

```
Response.Redirect("Wfrm_impresion_anterior.aspx?id_folio=" &
txt_idfolio.Text)
```

Al abrir la pantalla de “Wfrm_impresion_anterior.aspx” en la parte de page load, se manda a llamar el número del oficio, con el siguiente *código*:

```
Dim servreg As String = Request.QueryString("id_folio")
Dim da As New SqlClient.SqlDataAdapter("select * from ventradas_historico
WHERE id_folio=" + servreg, oConn)
```

Al realizar la búsqueda me muestra el documento como en la Figura 4.26.

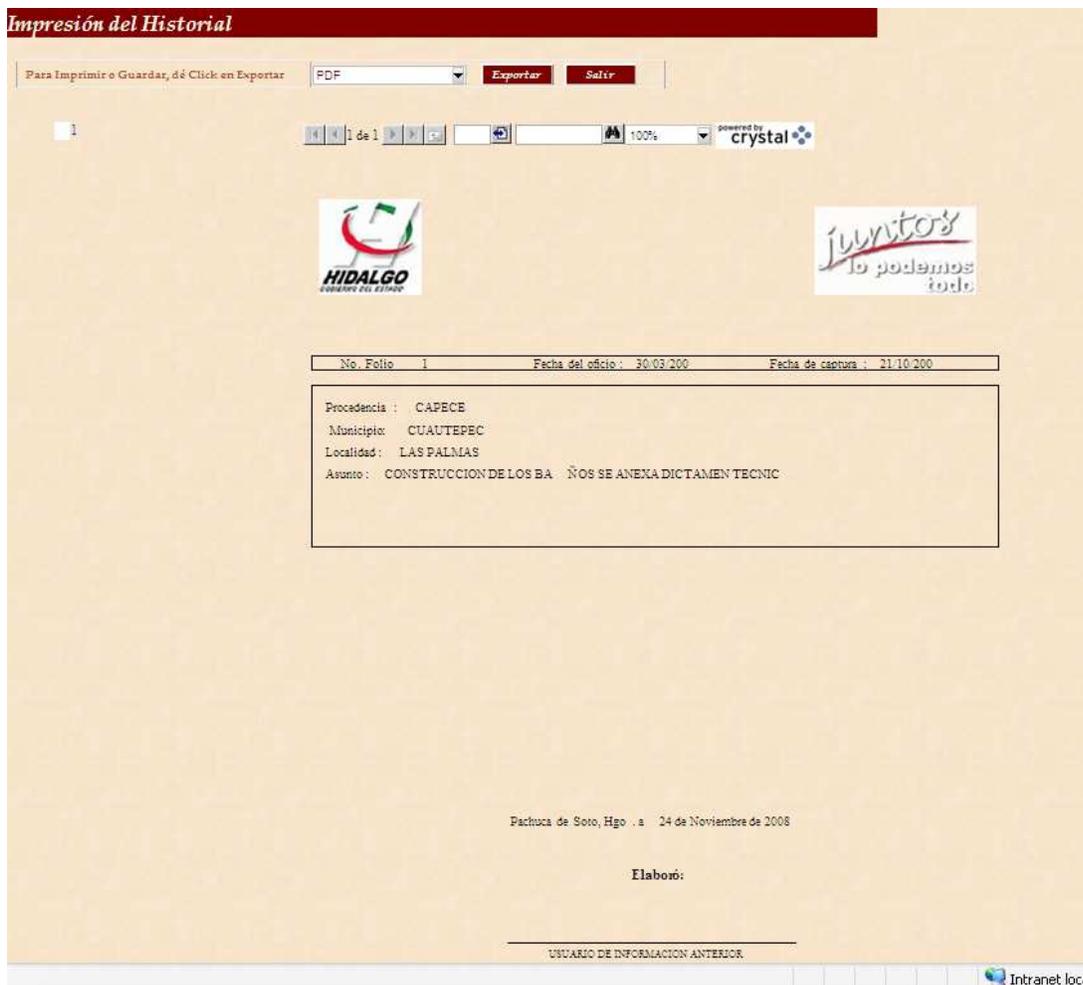


Figura 4.26. Diseño del Oficio de la Consulta de la información Anterior.

Para imprimir el documento, hay que exportarlo en cualquiera de los formatos que se muestra en el *combo* de la Figura 4.27.

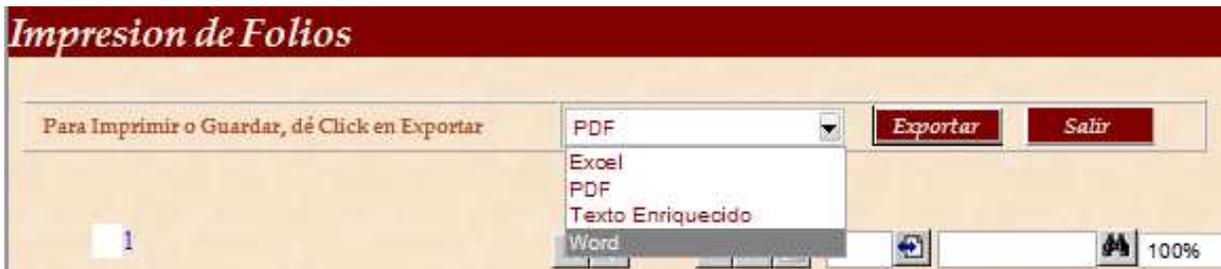


Figura 4.27. Exportación de Oficios de la pantalla de Información Actual.

Y al darle clic en el *botón* de Exportar, manda el siguiente cuadro de diálogos (Figura 4.28).

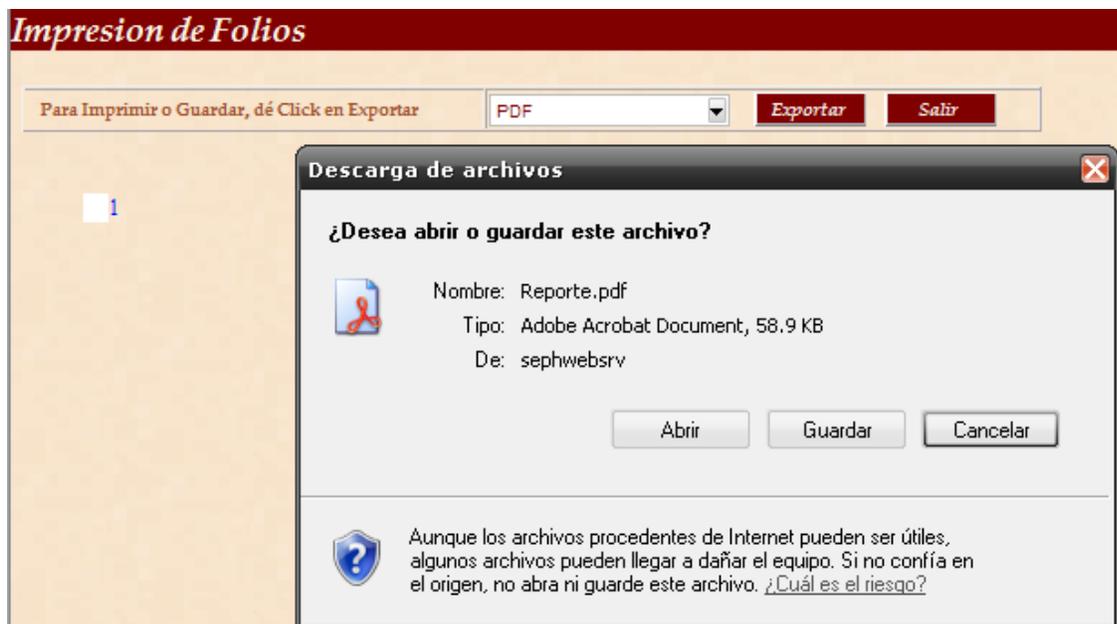


Figura 4.28. Cuadro de diálogos para la Exportación de un Oficio

Y a continuación abre el archivo como en la Figura 4.29 dependiendo del formato que se haya elegido por el usuario (en este ejemplo formato .PDF):



4.29. Exportación del Oficio de Consulta Anterior en PDF

4.1.5. PANTALLA DE CONSULTAS DE INFORMES X CCT

Pantalla de Consulta de Informes por CCT (Figura 4.30)

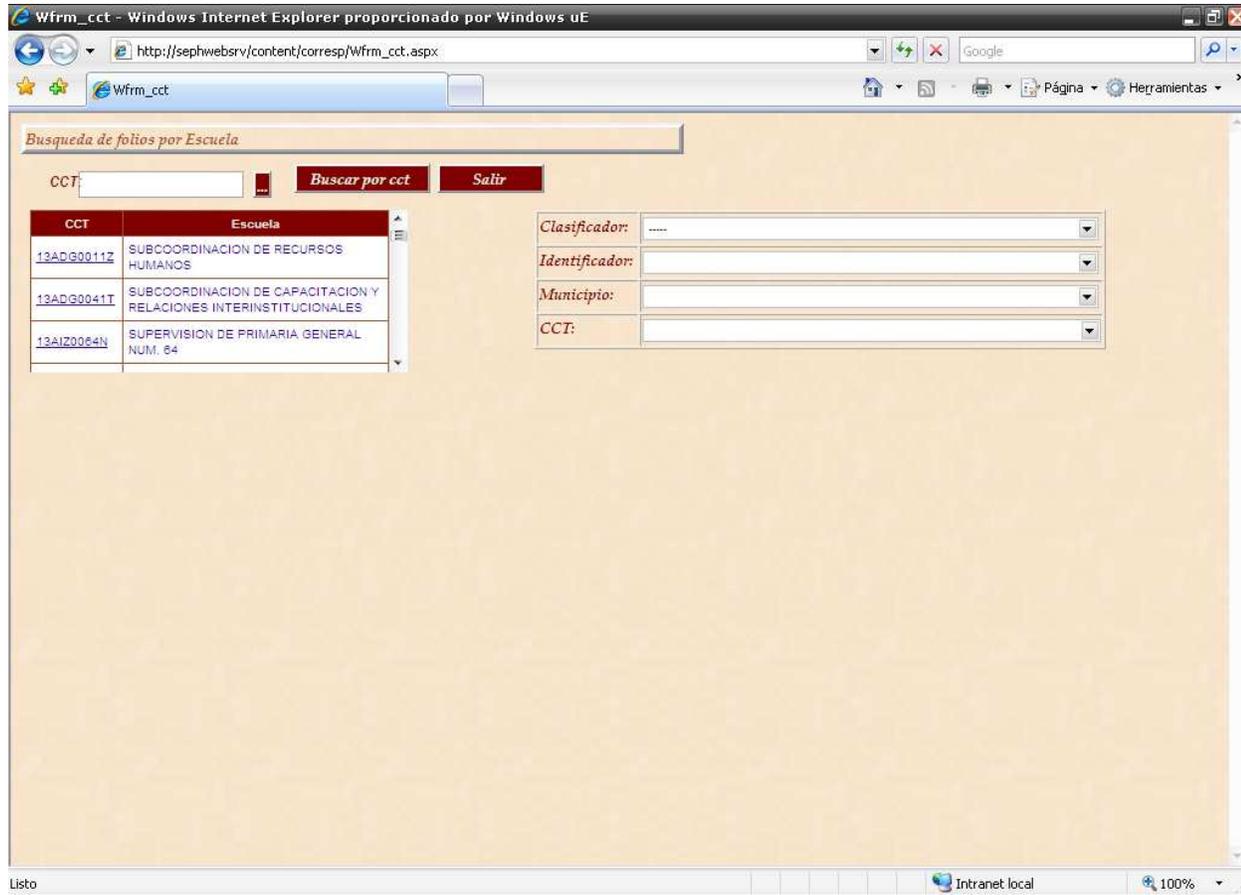


Figura 4.30. Pantalla de Consultas de Informes por CCT

Para consultar un oficio de la *información* actual por las claves de los centros de trabajo (CCT), lo primero que se tiene que hacer es darle clic en el menú principal y seleccionar la opción de Consultas de Informes x CCT la cuál nos envía a su respectiva pantalla (Figura 3.13.), donde se puede buscar e imprimir el/los oficios necesarios con toda la *información* requerida por el usuario. Para realizar una consulta de un(os) oficios, es necesario, seleccionar del *grid* la clave del CCT con la cual se desea realizar la búsqueda, el cual se llena con la siguiente función:

Función, que busca dentro de una vista "ventradas"

```
Public Function Sel_grdcct(ByVal tabla As String) As DataTable
Try
Dim Ssql As String = "Select distinct cct, nombre from ventradas where
cct is not null"
...
End Try
End Function
```

Al seleccionar la clave del centro de trabajo (CCT), automáticamente aparece en la *caja de texto* CCT. Y aparece el *botón* de Imprimir y la *información* del oficio en un *grid*. Figura 4.31.



Figura 4.31. Información del CCT y la aparición del botón imprimir.

Al darle clic en el *botón* **Imprimir** abre otra pantalla, donde manda a llamar el informe del CCT, con toda la *información* solicitada por el usuario. Figura 4.32.

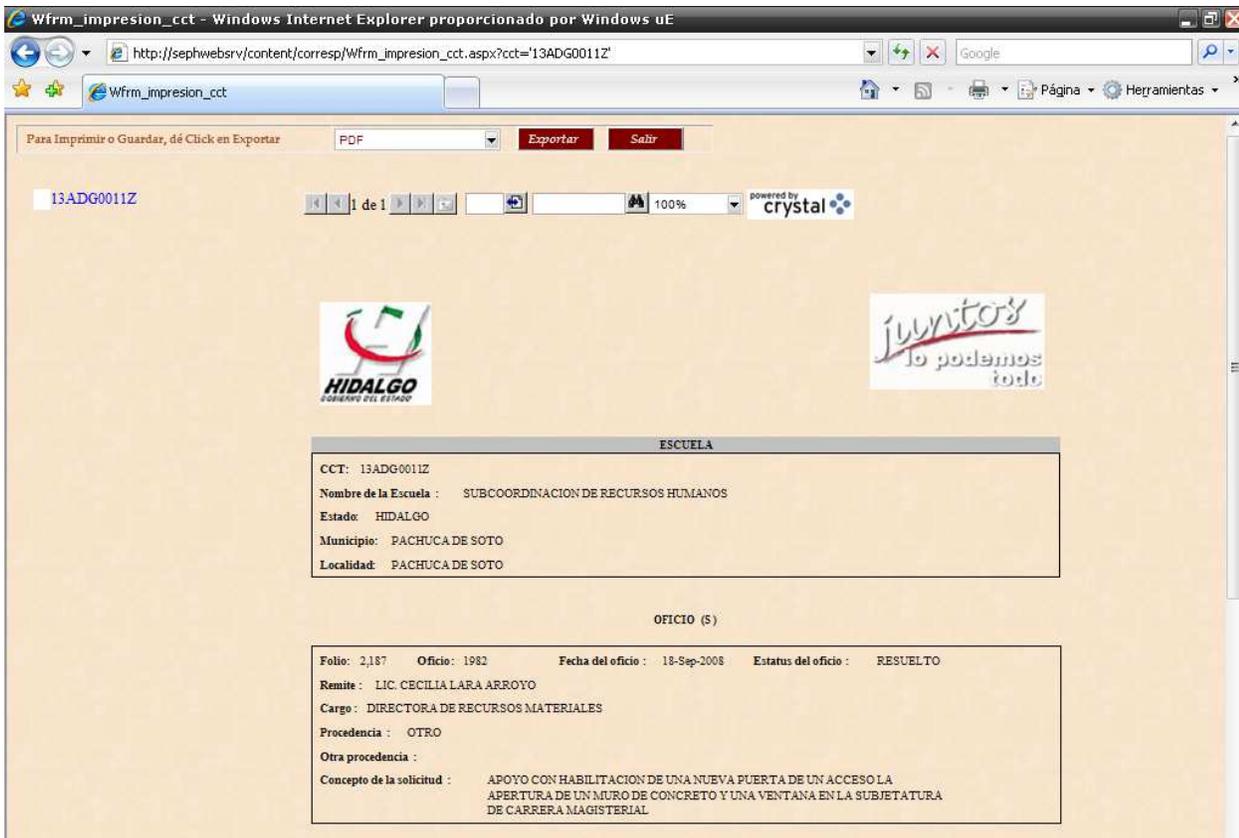


Figura 4.32. Formato del Reporte para las Consultas por CCT

Para imprimir el documento, hay que exportarlo en cualquiera de los formatos que se muestra en el *combo* de la Figura 4.33.

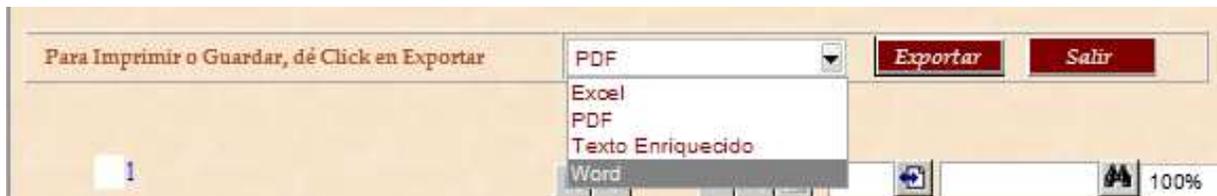


Figura 4.33. Como Exportar el documento en Consultas por CCT.

Y al darle clic en el *botón* de Exportar, manda un cuadro de diálogos (Figura 4.34).

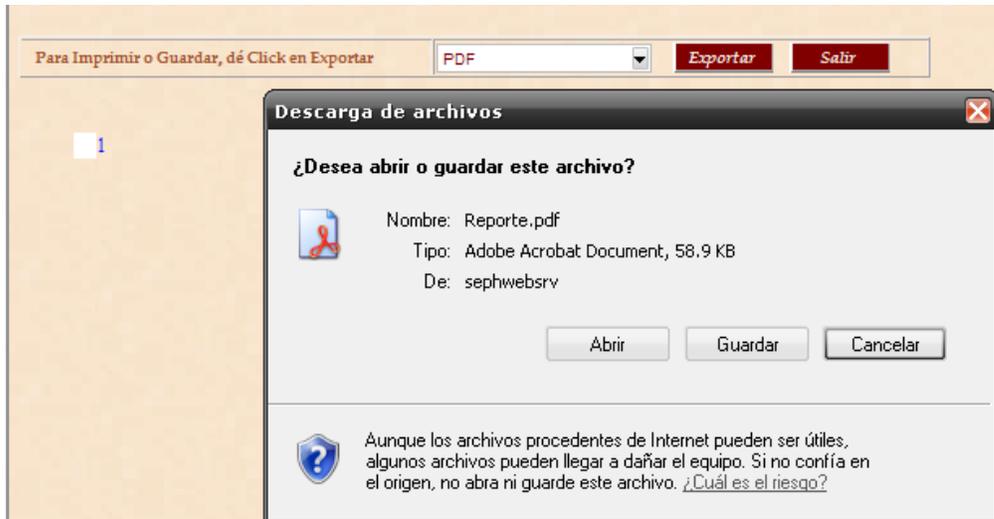


Figura 4.34. Cuadro de Descarga de archivos de Consultas por CCT

Y a continuación se abre el archivo como en la Figura 4.35 dependiendo del formato que se haya elegido por el usuario (en este ejemplo formato .PDF):

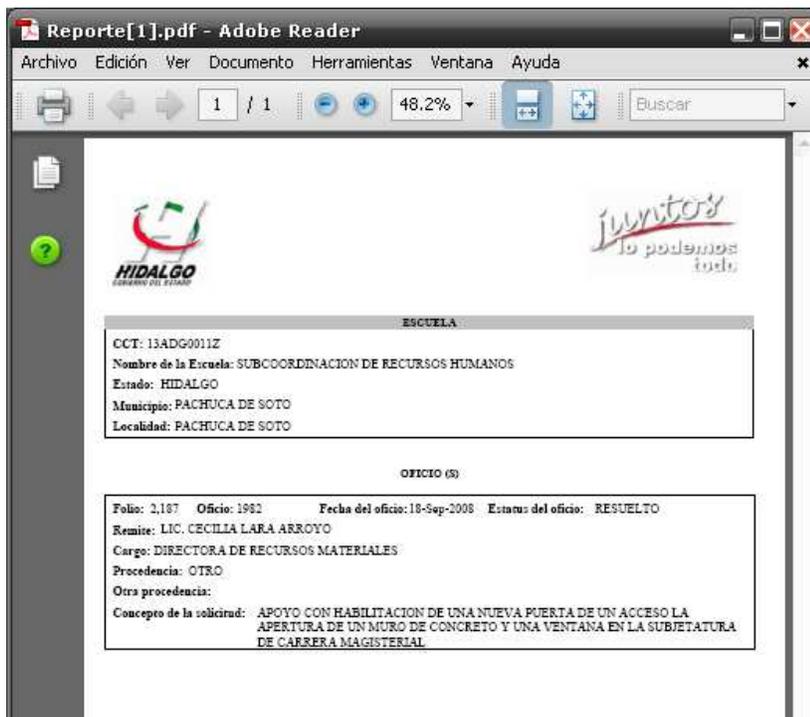


Figura 4.35. Exportación del Oficio por consultas del CCT en PDF.

Si se le da clic en el botón **Salir** del sistema, nos regresa a la pantalla de Consulta de Informes x CCT. Donde también podemos buscar el oficio llenando los combos de selección: *clasificador*, *identificador*, *municipio* y *CCT*. Figura 4.36.

Clasificador:	A- ADMINISTRATIVO
Identificador:	TV-TELESECUNDARIA
Municipio:	HUEJUTLA DE REYES
CCT:	----- ----- INSPECCION DE ZONA DE TELESECUNDARIA NUM. 5 INSPECCION DE ZONA DE TELESECUNDARIA NUM. 9 INSPECCION DE ZONA DE TELESECUNDARIA NUM. 27 INSPECCION DE ZONA DE TELESECUNDARIA NUM. 29

Figura 4.36.

Combos

de selección para obtener una clave de CCT.

El *combo* de Clasificador se llena, a través de la siguiente función:

```
cbo_nivel.DataSource = cprincipal.SelCatCboClas()
cbo_nivel.DataValueField = "clasificador"
cbo_nivel.DataTextField = "desclasificador"
cbo_nivel.DataBind()
cbo_nivel.Enabled = True
```

Seleccionando el clasificador, se activa el combo de identificador, tomando en cuenta lo que se selecciona en el combo del clasificador, para hacer una búsqueda más exacta, el *combo* se llena de la siguiente manera:

```
cbo_ident.DataSource = cprincipal.SelCatCboIden(cbo_nivel.SelectedValue)
cbo_ident.DataValueField = "identificador"
cbo_ident.DataTextField = "desidentificador"
cbo_ident.DataBind()
cbo_ident.Enabled = True
```

Al seleccionar el identificador, procedemos a llenar el *combo* del municipio, en el llenado tomamos en cuenta el clasificador y el identificador seleccionados anteriormente:

```
cbo_mun.DataSource = cprincipal.SelCatCboMun(cbo_nivel.SelectedValue,
cbo_ident.SelectedValue)
cbo_mun.DataValueField = "cvemun"
cbo_mun.DataTextField = "municipio"
cbo_mun.DataBind()
```

```
cbo_mun.Enabled = True
```

Y para finalizar al seleccionar el municipio, se llena el *combo* del CCT, realiza una búsqueda, de acuerdo a lo seleccionado en los combos anteriores:

```
cbo_cct.DataSource = cprincipal.SelCatCboct(cbo_nivel.SelectedValue,
cbo_ident.SelectedValue, cbo_mun.SelectedValue)
cbo_cct.DataValueField = "cct"
cbo_cct.DataTextField = "nombre"
cbo_cct.DataBind()
cbo_cct.Enabled = True
```

La función utilizada es la siguiente:

```
Public Function SelCatCboct(ByVal clasificador As String, ByVal
identificador As String, ByVal municipio As String) As DataTable

Dim Ssql As String = "SELECT cct='000',nombre='-----' union all select
cct, nombre from Vescuelas"
...
Ssql = Ssql & " and cvemun =' " & Trim(municipio) & "'"
```

Al seleccionar la escuela, la clave del CCT aparece en la *caja de texto*

CCT: , la cual no se podrá modificar por seguridad a que el usuario no se equivoque en la escritura del CCT. Y automáticamente aparecerá el *botón* de imprimir, y se sigue el procedimiento especificado anteriormente.

Existe otra manera de buscar el oficio, al darle clic en el *botón*

Buscar por cct

aparecerá la *caja de texto* CCT (Figura 4.37) donde se podrá escribir una parte de la clave del CCT, para de esta forma al darle clic en el *botón*



el sistema hará una búsqueda de todos los Centros de Trabajo similares al que se escribió en la *caja de texto* del CCT.

Escribe la clave del CCT:

Figura 4.37. Como buscar un centro de trabajo

El código de búsqueda del CCT es el siguiente:

```
Dim Ssql As String = "Select * from cat_escuelas where cct like '%" & cct & "%' order by cct"
```

Se llena un *grid* con la *información* encontrada con la función (Figura 4.38) donde al seleccionar la Clave del Centro de Trabajo (CCT) aparecerá en la *caja de texto* **CCT:** y aparece el *botón* de imprimir.

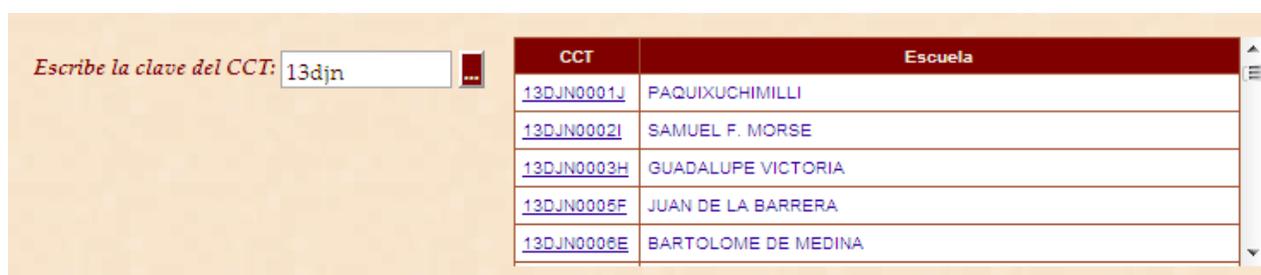


Figura 4.38. Resultado de la Búsqueda de los CCT

Para imprimir el archivo se sigue el procedimiento especificado anteriormente.

Si se le da clic en el *botón* **Salir** del sistema, nos regresa a la pantalla del Menú Principal (Figura 4.39).



Figura 4.39. Menú Salir

El menú Salir permitirá cerrar la sesión actual, y así poder ingresar al sistema con una cuenta diferente, o bien salir totalmente del sistema.

CONCLUSIONES

El surgimiento de nuevas tecnologías, en especial de tecnologías de información, ha abierto nuevos espacios para el desarrollo de sistemas, cuyas capacidades de manejo, distribución y automatización de datos, son un todo para las empresas, permiten una comunicación rápida, tal es el caso del Sistema mostrado en este trabajo, pues acceden a un mayor desempeño en el manejo de la información.

A lo largo de este documento, se ha intentado mostrar la visión general de los cambios, al haber realizado un análisis y diseño, basados en el Sistema de Correspondencia anterior, que dieron como resultado que el Actual fuera desarrollado de acuerdo a las necesidades del usuario, tanto en el manejo de la información como en la sencillez del sistema.

Este Sistema de Correspondencia es una herramienta de gran apoyo en el Departamento de Infraestructura de la SEP, pues permite manejar un mayor número de oficios, a los cuales se debe dar respuesta en tiempo y forma breve.

Por lo que hace de Microsoft SQL Server 2000 y Visual Studio .NET 2003, se comprueba que son herramientas ideales para manejar información vía intranet bajo la red de la SEP.

Asimismo, con la puesta en marcha del Sistema de Correspondencia en este departamento, los usuarios pueden almacenar y acceder a la información de manera inmediata, de forma fácil.

Para concluir con esta parte, vale la pena comentar que el uso de las tecnologías de información sigue ganando terreno, utilizando todas las herramientas disponibles, y con el uso adecuado permitirá continuar creciendo a pasos agigantados para facilidad del usuario y ahorro de tiempo.

REFERENCIAS BIBLIOGRÁFICAS

- [1] TICs en Centroamérica
Impacto de la adopción de las tecnologías de la Información y
Comunicación en el desempeño de las empresas
2005
- [2] Tecnologías y Redes de transmisión de datos
Editorial Limusa
- [3] La Biblia de Microsoft Visual Studio .NET
Ediciones Anaya Multimedia
Julian Templeman / David Vitter
2002
- [4] Ayuda del Administrador corporativo de Microsoft SQL Server
Introducción a la arquitectura de SQL Server

REFERENCIAS ELECTRÓNICAS

- [5] Informática – Wikipedia, la enciclopedia libre
<http://es.wikipedia.org/wiki/Computaci%C3%B3n>
- [6] Calsificación de sistemas - Wikilearning
http://www.wikilearning.com/monografia/teoria_de_sistemas-clasificacion_de_los_sistemas/12557-3
- [7] http://kybele.escet.urjc.es/documentos/SI/%5BSI-2006-07%5DT2_Rol_y_Tipos_de_SI.pdf
- [8] Software – Wikipedia, la enciclopedia libre
http://es.wikipedia.org/wiki/Software#Modelos_de_Proceso_o_Ciclo_de_Vida
- [9] Definición de metodología – Que es, Significado y Concepto
<http://definicion.de/metodologia/>
- [10] SQL Server, Administrador Bases de Datos
http://www.netveloper.com/contenido2.aspx?IDC=66_0&IDP=3&P=64
creacion de una base de datos

GLOSARIO

BD. Base de Datos.

Botón. Interfáz gráfica de usuarios que permiten comenzar un evento como buscar, aceptar una tarea, eliminar información, etc.

Bucle. Conjunto de instrucciones contenidas en un programa o rutina que se repite un número determinado de veces.

Caja de texto. Elemento típico en las interfaces gráficas que permite la inserción de texto.

CCT. Clave de Centro de Trabajo.

Código. Conjunto de instrucciones que componen un programa, escrito en cualquier lenguaje.

Código máquina. Codificación de programas en sistema binario.

Combo. Elemento que permite al usuario seleccionar una opción de una lista existente de opciones.

Consulta. Extraer datos de una base de datos o solicitud para realizar una acción en los datos.

Contraseña. Código utilizado para acceder a un sistema restringido. Pueden contener caracteres alfanuméricos e incluso otros símbolos. Se destaca que la contraseña no es visible en la pantalla al momento de ser tecleada con el propósito de que solo pueda ser conocida por el usuario.

Datetime. Tipo de datos de combinación de fecha y hora. El formato de almacenamiento es de día/mes/año horas:minutos:segundos.

DBase. Sistema de gestión de base de datos.

Direcciones IP. Número que identifica de manera lógica y jerárquica a una interfaz de un dispositivo dentro de una red que utilice el protocolo IP.

Dispositivos. Aparatos que van conectados a la computadora como la tarjeta madre, el CPU, controlador de video, disco duro, módem, etc.

DHCP. Es un protocolo que permite a los administradores de red automatizar y gestionar de manera centralizada la asignación de direcciones del protocolo Internet (IP) en una red de una organización.

Firewall PIX. PIX es el acrónimo de Private Internet Exchange y es una sigla utilizada por el fabricante tecnológico Cisco, para referirse a sus modelos de equipos Cortafuegos (FireWalls).

Formulario. Es la ventana sobre la que se colocan los controles de la interfaz gráfica que el usuario utilizará para comunicarse con la aplicación.

Función. Rutina de software independiente que realiza una tarea para el programa en que está escrita.

Grid. Cuadrícula para presentar datos n forma de tabla.

Hardware. Componentes físicos de una computadora o una red.

Host. Computador central o principal en un entorno de procesamiento distribuido. Es decir un computador central de tiempo compartido que controla una red.

Hub. Dispositivo que se utiliza como punto de conexión entre los componentes de una red de área local.

Información. Conjunto organizado de datos procesados, que constituyen un mensaje sobre un determinado fenómeno.

Informática. Ciencia que tiene como objetivo estudiar el tratamiento automático de la información a través de la computadora.

Interfaz. Zona de contacto o conexión entre dos componentes de "hardware"; entre dos aplicaciones; o entre un usuario y una aplicación. Apariencia externa de una aplicación informática.

Internet. Red de computadora de trabajo que está interconectada con otras para compartir información de cualquier índole.

Like. Instrucción de SQL, que se utiliza para realizar búsquedas con el criterio, "parecido a".

Menú. Herramienta gráfica en la interfaz de páginas web y aplicaciones que consiste de una lista de opciones que puede desplegarse para mostrar opciones y acceder a las mismas.

Método. Procedimiento para alcanzar objetivos.

Nodos. Punto de unión de elementos que confluyen en el mismo lugar. Cada red de computadora constituye un nodo.

Pantalla. Dispositivo de salida, su función es la de representar la información con la que se está trabajando.

Plataforma. Software y/o Hardware con el cuál una aplicación es compatible y permite ejecutarla.

Proxy. Hace referencia a un programa o dispositivo que realice una acción en representación de otro. En este caso permite el acceso a internet a todos los equipos de una organización cuando sólo se puede disponer de un único equipo conectado, es decir, una única dirección de IP.

Red. Manera de conectar equipos (computadoras y/o dispositivos) entre sí, compartiendo recursos, información, y servicios ya sea de Internet, Chat, juegos, etc. a través de cables, señales o cualquier transporte de datos.

Reporte. Objeto de una base de datos que imprime información con formato y organizada de acuerdo a sus especificaciones.

SE. Sistema Experto

Servicio Web. Conjunto de protocolos y estándares para el intercambio de datos entre distintas aplicaciones de software en lenguajes diferentes y que pueden utilizarse en la Internet.

SEP. Secretaria de Educación Pública del Estado de Hidalgo.

SI. Sistemas de información.

Software. Se refiere a los programas en general, aplicaciones, juegos, sistemas operativos, antivirus, etc. Lo que se pueda ejecutar en la computadora.

Switch. Dispositivo que permite la interconexión de redes sólo cuando esta conexión es necesaria.

Tabla. Es una relación o vínculo entre dos o más entidades que describe alguna interacción entre las mismas

Tarjeta ruteadora. Dispositivo conectado a la computadora que permite que los mensajes a través de la red se envíen de un punto a otro (emisor-destinatario).

Tchar2. Tipo de datos char, longitud 2, y que no permite valores nulos.

Text. Tipo de datos de texto con un máximo de 65535 caracteres.

Tfecha. Tipo de datos datetime, solo que guarda solo la fecha sin la hora.

Tfolio. Tipo de datos int para almacenar números enteros.

TI. Tecnologías de información.

Topología de Red. Forma en cómo se comunican los distintos nodos de una red.

Varchar. Tipo de datos que almacena una cadena de longitud variable. La cadena podrá contener desde 0 a 255 caracteres.

Vista. Consulta de un contenido de una tabla de una base de datos, a la que se puede acceder a la información como si fuera una tabla.

Windows. Sistema operativo desarrollado por la empresa Microsoft.