



**Universidad  
Latina**

**UNIVERSIDAD LATINA S.C.**  
INCORPORADA A UNAM

---

---

**AUTOMATIZACIÓN PARA EL CONTROL DE  
INSTRUMENTAL DE PROTECCIÓN CIVIL EN LA  
SECRETARÍA DE HACIENDA Y CRÉDITO PÚBLICO**

**DISEÑO DE UN SISTEMA PARA UNA ORGANIZACIÓN**

**QUE PARA OBTENER EL TÍTULO DE:  
LICENCIADO EN INFORMÁTICA**

**P R E S E N T A**

**JUAN TOMAS MEDINA ESPINOZA**

**NOMBRE DEL ASESOR: LIC. HUGO VELÁZQUEZ BRITO**

**MÉXICO, D.F. ABRIL 2011**



Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

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

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

## **AGRADECIMIENTOS**

Son tantas personas a las cuales debo parte de este triunfo, de lograr alcanzar mi culminación académica, la cual es el anhelo de todos los que así lo deseamos.

Definitivamente, Dios, mi Señor, mi Guía, mi Proveedor, mi Fin Último; sabes lo esencial que has sido en mi posición firme de alcanzar esta meta, esta alegría, que si pudiera hacerla material, la hiciera para entregártela, pero a través de esta meta, podré siempre de tu mano alcanzar otras que espero sean para tu Gloria.

A mis padres y mis hermanos, por darme la estabilidad emocional, sentimental; para poder llegar hasta este logro, que definitivamente no hubiese podido ser realidad sin ustedes. GRACIAS por darme la posibilidad de que de mi boca salga esa palabra...FAMILIA. Mamá y Papá, serán siempre mi inspiración para alcanzar mis metas, gracias por enseñarme que todo se aprende y que todo esfuerzo es al final una recompensa. Su esfuerzo, se convirtió en su triunfo y el mío, LOS AMO.

A todos mis amigos pasados y presentes; pasados por ayudarme a crecer y madurar como persona y presentes por estar siempre conmigo apoyándome en todas las circunstancias posibles, también son parte de esta alegría, LOS RECUERDO.

### **Dafne Gissel Vázquez Guerrero**

Por tu apoyo, comprensión y amor que me permite sentir poder lograr lo que me proponga. Gracias por escucharme y por tus consejos (eso es algo que lo haces muy bien). Gracias por ser parte de mi vida; eres lo mejor que me ha pasado.

### **Gracias al C.P. Agustín Torres Campos**

Que participó en mi desarrollo profesional durante mi carrera, sin su ayuda y conocimientos no estaría en donde me encuentro ahora.

### **Gracias a mi asesor Lic. Hugo Brito Velázquez**

Por permitirme, ser parte del grupo de trabajo. Tus consejos, paciencia y opiniones sirvieron para que me sienta satisfecho en mi participación dentro del proyecto de investigación.

Gracias a mis compañeros y hermanos de Tiev, Protección Civil, la Cruz Roja y de radio-experimentadores por su apoyo moral y laboral.

## Introducción

## Antecedentes

## Objetivo

## Justificación

### Capítulo I.-Protección Civil de la Secretaría de Hacienda y Crédito Público:

1.1	Origen de la Protección Civil en la Secretaría de Hacienda y Crédito Público.....	04
1.1.1	Identificación.....	04
1.1.2	Dimensiones.....	05
1.1.3	Instrumental.....	05

### Capítulo II.- Características y aportaciones del sistema a desarrollar:

2.1	Propuesta para la automatización del sistema.....	06
2.2	Estudio de factibilidad.....	06
2.2.1	Factibilidad Técnica.....	06
2.2.2	Operativa.....	06
2.2.3	Financiera.....	07
2.3	Ciclo de vida de los sistemas aplicado a la Automatización para el Control de Instrumental de Protección Civil en la Secretaría de Hacienda y Crédito Público.....	07
2.3.1	Análisis de la situación actual.....	07
2.3.2	Especificación de requerimientos.....	07
2.3.3	Diseño General.....	08
2.3.4	Diseño detallado en el diccionario de datos.....	13
2.3.5	Programación y Desarrollo.....	28
2.3.6	Pruebas y Depuración del Sistema.....	125
2.3.7	Instalación.....	125
2.3.8	Mantenimiento.....	125
	Bibliografía .....	127

## **Introducción:**

Debido a la importancia que representa para nuestro país el fomento de la Protección Civil esta tesis se titula “Automatización para el Control de Instrumental de Protección Civil en la Secretaría de Hacienda y Crédito Público”, se presentan las bases a través de las cuales se puede establecer la estrategia idónea para llevar a cabo la introducción del Sistema Integral de Control de Instrumental en el área de Protección Civil en la Secretaría de Hacienda y Crédito Público.

En el primer capítulo se da a conocer la función de Protección Civil y su origen funcional en la Secretaría de Hacienda y Crédito Público.

En el segundo capítulo se presentan las características y elementos con que cuenta la institución, para establecer una Automatización para el Control de Instrumental de Protección Civil en la Secretaría de Hacienda y Crédito Público, así como el desarrollo sistemático en lenguaje de programación y base de datos a través del ciclo de vida de sistemas.

Esperamos que esta tesis sirva como antecedente para las demás instituciones que no cuenten con una Automatización para el Control de Instrumental de Protección Civil en la Secretaría de Hacienda y Crédito Público.

## **Antecedentes:**

A partir de la Primera y Segunda Guerra Mundial, la Población Civil comprendió que durante los conflictos armados, los gobiernos, las instituciones de emergencia y de servicio; se ve siempre totalmente rebasadas dejando siempre a la Población Civil en el desamparo, motivo por el cual, la población de las ciudades en conflicto en su instinto de supervivencia, fueron creando pequeños grupos que asistían a la población en estas situaciones denominándose como Defensa Civil.

## **Objetivo:**

Dadas las condiciones en que la Protección Civil de la Secretaría de Hacienda y Crédito Público opera, surge la inquietud de conocer un procedimiento más efectivo e innovador, para poder controlar sus actividades, por lo cual se consideran los siguientes puntos:

- a) Analizar la situación actual.
- b) Especificar los requerimientos
- c) Establecer una nueva estructura
- d) Diseño de un programa operativo.

De lo anterior destaca la importancia de conocer el funcionamiento actual y tener una visión más amplia de las posibilidades de una nueva forma de trabajo, que bien cabe mencionar la implementación de un nuevo sistema.

La presente tesis “Automatización para el Control de Instrumental de Protección Civil en la Secretaría de Hacienda y Crédito Público”, busca implementar un sistema de información automatizado, que se encargue de realizar procesos de almacenamiento de información proveniente de la captura de adquisiciones, prestamos, solicitudes de servicios, que a través de las cuales se puedan realizar consultas y generar reportes para tener la información en forma rápida y oportuna.

El origen de esta hipótesis surge, de la necesidad de tener un control más estricto en las solicitudes de demanda por parte de los usuarios de las diversas áreas en que Protección Civil tiene lugar para su funcionamiento, cabe señalar que por la forma en que se maneja, se puede decir que existen una serie de irregularidades en el manejo de equipo e instrumental, dando lugar en los faltantes al inventariar este material y equipo.

Por otro lado la demanda de servicios y adquisición de instrumental y equipo, crece en forma desordenada, por lo cual surge la necesidad de llevar la implementación de un nuevo sistema de control que aporte información segura y confiable y que permita al mantener un almacén ordenado y certero.

### **Justificación**

En la actualidad la mayoría de las empresas, tanto privadas como gubernamentales, sostiene sus operaciones, con base en sistemas automatizados, Que pueden ser un tanto rústicos o demasiado complejos por lo cual los resultados en aspectos cualitativos y cuantitativos tienen un menor margen de error, motivo por el cual es la secretaria de hacienda y crédito público, en lo concerniente al área de Protección civil, no cuenta con un sistema automático en informática que controle las operaciones en función a la demanda de servicios y adquisición de equipo e instrumental, dando lugar a mantener procedimientos añejos que provocan un mal manejo de información en los resultados.

## **CAPÍTULO I. - PROTECCIÓN CIVIL DE LA SECRETARÍA DE HACIENDA Y CRÉDITO PÚBLICO**

### **1.1. - Origen de la Protección Civil en la Secretaría de Hacienda y Crédito Público.**

A raíz de los sismos ocurridos en Septiembre de 1985, el Ejecutivo Federal convocó a un grupo, denominándolo Comisión Nacional de Reconstrucción, en cuyo seno se desarrollaron los trabajos del Comité de Prevención de Seguridad Civil, el cual generó el documento central contenido en el decreto por el que se aprobaron las bases para el establecimiento del Sistema Nacional de Protección Civil publicado en el Diario Oficial de la Federación del 6 de mayo de 1986.

#### **1.1.1. -Identificación**

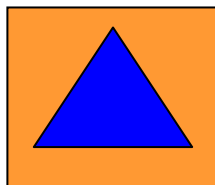
El signo distintivo internacional de Protección Civil consiste en un triángulo equilátero azul sobre fondo color naranja, cuando se utilice para la protección de organismos de Protección Civil, de su personal, sus edificios y su material o para la protección de refugios civiles.

Signo distintivo Internacional

1. - El signo distintivo internacional de Protección Civil previsto en el párrafo 4° del artículo 66 del protocolo N°1 de los convenios de Ginebra; será un triángulo equilátero azul sobre fondo naranja. En la figura que a continuación, aparece en el modelo.

2. - Se recomienda:

- Que si el triángulo azul se utiliza en una bandera, brazalete o dorsal, estos constituyen su fondo naranja.
- Que uno de los ángulos del triángulo apunte hacia arriba, verticalmente.
- Que ninguno de sus tres ángulos tenga contacto con el borde del fondo naranja.



### 1.1.2 - Dimensión de las áreas de participación.

Es un área específica con 20 Unidades Internas de Protección Civil con un total de 7294 personas de la Secretaría de Hacienda y Crédito Público.

### 1.1.3. Instrumental que ocupan las Brigadas y Equipo para las áreas de trabajo

#### EQUIPO PARA BRIGADISTAS

La relación de equipo que a continuación se detalla es la sugerida en la Guía de Implementación de Programas de Protección Civil, de la Secretaría de Gobernación, la cual no es restrictiva y puede ser adaptada según las necesidades.

##### **BRIGADA DE PRIMEROS AUXILIOS**

- Chaleco identificador
- Casco o cachucha
- Lentes protectores
- Botas tipo minero
- Brazaletes
- Botiquín portátil de Primeros auxilios
- Camilla

##### **BRIGADA DE PREVENCIÓN Y COMBATE DE INCENDIOS**

- Chaquetón y pantalón de bombero
- Casco con careta de plástico de alto impacto
- Guantes
- Botas de bombero
- Equipo de aire autónomo
- Hacha
- Lámpara sorda

##### **BRIGADA DE EVACUACIÓN DE INMUEBLES**

- Chaleco identificador
- Overol
- Cachucha
- Brazaletes
- Lámpara sorda
- Silbato

##### **BRIGADA DE BUSQUEDA Y RESCATE**

- Casco
- Lentes protectores
- Botas tipo minero
- Mascarilla para polvos
- Guantes
- Brazaletes
- Pico y pala
- Hacha
- Barreta
- Lámpara sorda
- Silbato
- Cuerda perlón de 9mm o 11mm de 50 mts
- Descensores tipo ocho
- Mosquetones con seguro
- Arneses individuales



## **CAPÍTULO II.- CARACTERÍSTICAS Y APORTACIONES DEL SISTEMA A DESARROLLAR.**

### **2.1.-AUTOMATIZACIÓN PARA EL CONTROL DE INSTRUMENTAL DE PROTECCIÓN CIVIL EN LA SECRETARÍA DE HACIENDA Y CRÉDITO PÚBLICO**

Es un sistema que se desarrollo con la finalidad de llevar a cabo un control automatizado del instrumental que maneja el personal de Protección Civil y que tiene la característica de ser útil y rápido para efectuar consultas a través de pantalla u hojas de reporte.

La Automatización para el Control de Instrumental de Protección Civil en la Secretaría de Hacienda y Crédito Público, tiene la ventaja de ser un sistema sencillo y fácil de manejar ya que el personal tiene las opciones de ingresar con una clave personalizada y revisar las operaciones que han efectuado por fecha y hora.

### **2.2.- ESTUDIO DE FACTIBILIDAD**

Para llevar a cabo la elaboración del presente proyecto, se tuvo la necesidad de analizar la posibilidad de llevar a cabo su desarrollo en función a los recursos humanos, técnicos así como los financieros con que cuenta la institución.

#### **2.2.1.- FACTIBILIDAD TÉCNICA**

Actualmente la organización cuenta con tecnología adecuada para implementar un sistema que controle el manejo de Instrumental de Protección Civil.

El equipo tiene las características que solicita la propuesta por parte de sistemas para su desarrollo, el personal también tiene las características necesarias ya que cuentan con conocimientos de computación y en captura de datos, así mismo, existe la posibilidad de agrandar el sistema de acuerdo a los requerimientos en el crecimiento de la misma.

#### **2.2.2.- FACTIBILIDAD OPERATIVA**

La necesidad de mantener el orden en las adquisiciones y distribución de equipo e instrumental y servicios de Protección Civil une al personal del área mencionada a participar en forma incondicional al desarrollo y uso del sistema.

### **2.2.3.- FACTIBILIDAD FINANCIERA**

En realidad el costo de llevar a cabo el desarrollo del sistema, es mínimo ya que solo se requiere de actualizar la forma de trabajo en lo que refiere a tener un control más estricto en el inventario de equipo, instrumental y servicios, pero el beneficio es mayor se puede decir ya que el inventario terminara cada corte con un mínimo de faltantes ya que existe la eficacia en el guardado de la información lo que ocasionará un ahorro por algún porcentaje que puede ser significativo.

### **2.3.- CICLO DE VIDA DE LOS SISTEMAS APLICADO A LA AUTOMATIZACIÓN PARA EL CONTROL DE INSTRUMENTAL DE PROTECCIÓN CIVIL EN LA SECRETARÍA DE HACIENDA Y CRÉDITO PÚBLICO**

En función al estudio de factibilidad este proyecto se llevo a cabo a través de la metodología del proceso del ciclo de vida de un sistema, ya que por su origen y funcionalidad tendrá lugar en una organización emprendedora e innovadora en sus actividades de control de información.

#### **2.3.1.- ANÁLISIS DE LA SITUACIÓN ACTUAL**

Actualmente los informes solicitados por la dirección se obtienen a través de la apertura de una hoja de cálculo que contiene información en forma desordenada, no se obtienen los datos solicitados con la exactitud necesaria.

#### **2.3.2- ESPECIFICACIÓN DE REQUERIMIENTOS**

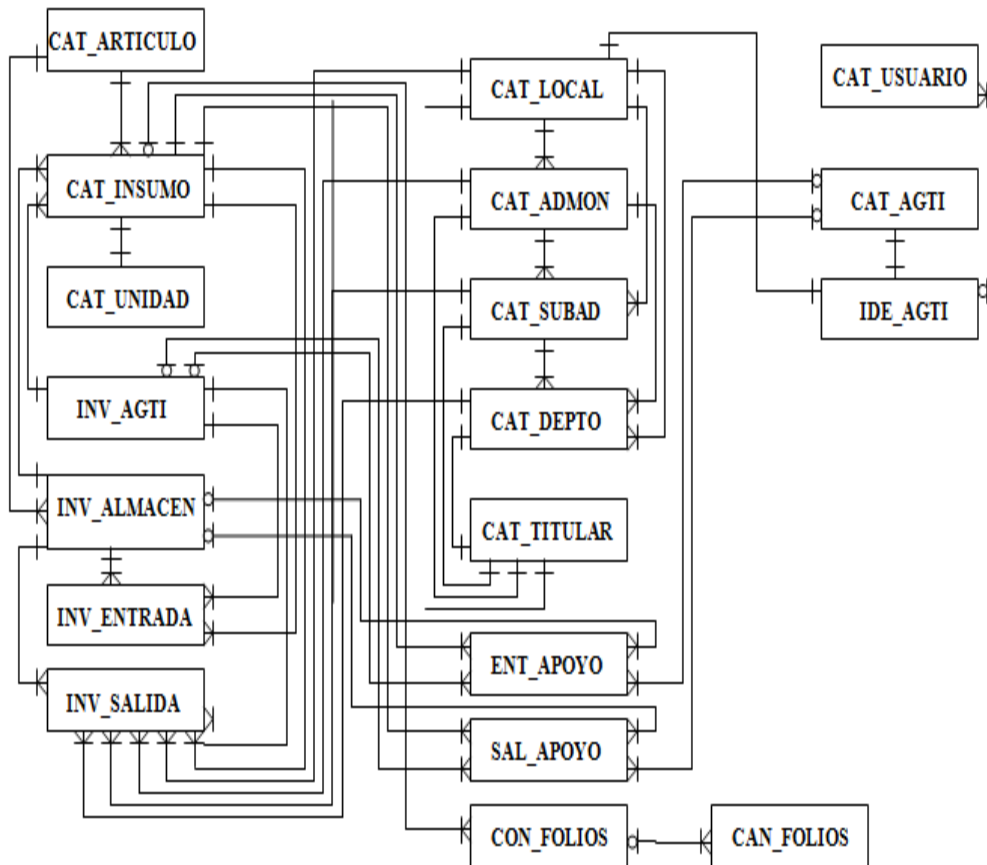
A través del estudio de factibilidad operativa, se llega a la conclusión de que los usuarios requieren de una nueva forma manejar esta actividad de control.

### 2.3.3.- DISEÑO GENERAL.

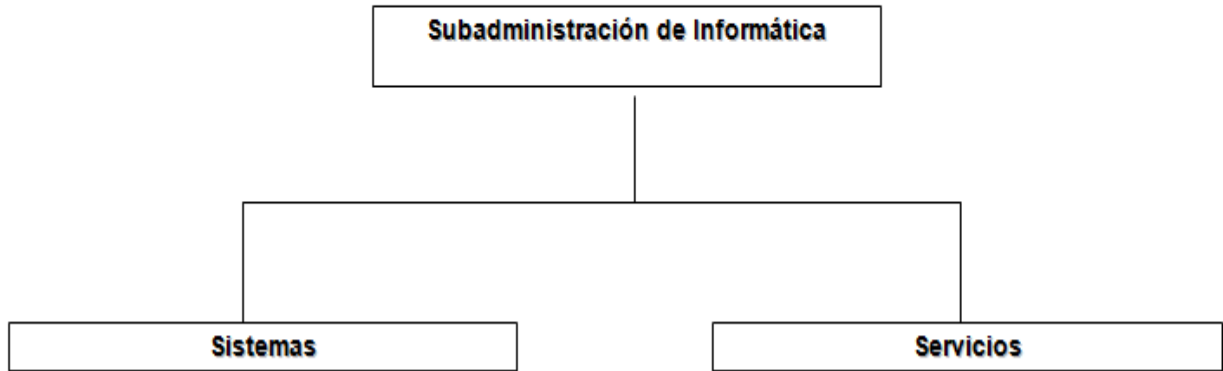
A) El diagrama jerárquico se estructura de la siguiente manera:

#### SISTEMA DE CONTROL DE INSUMOS

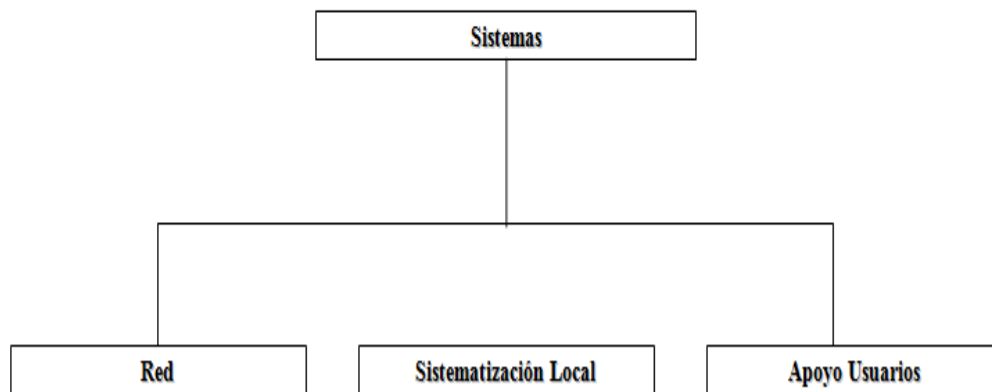
#### DIAGRAMA ENTIDAD-RELACIÓN



## Diagrama de flujo de datos

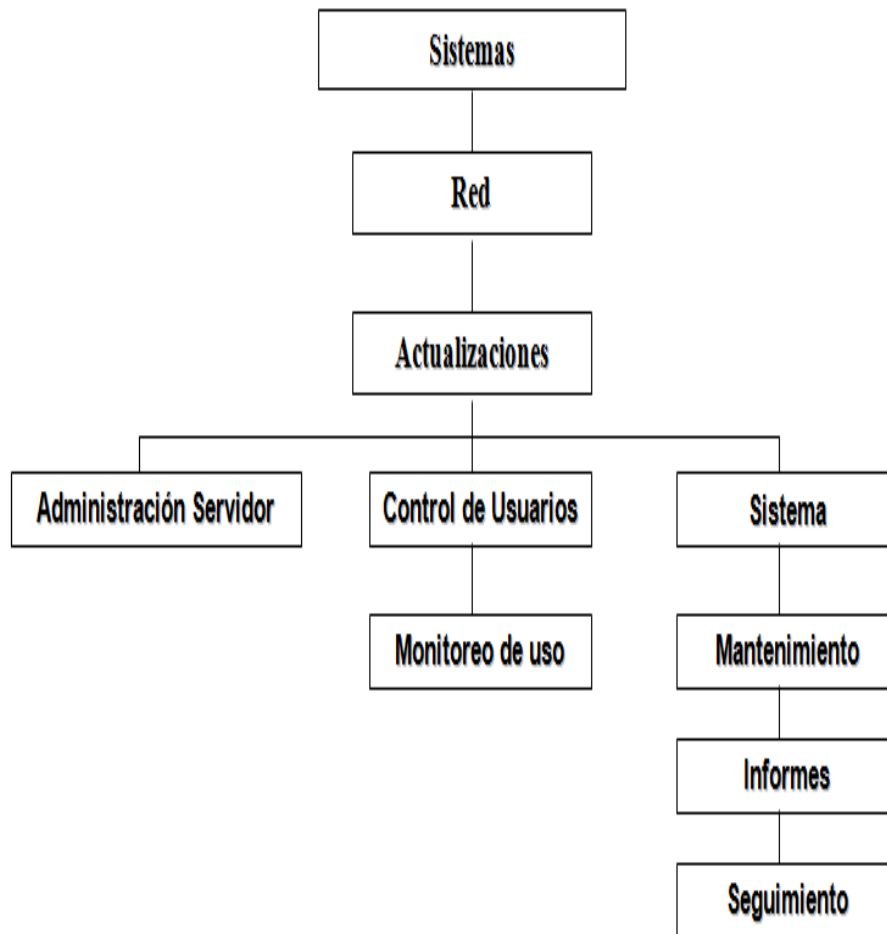


## Funciones de Area:

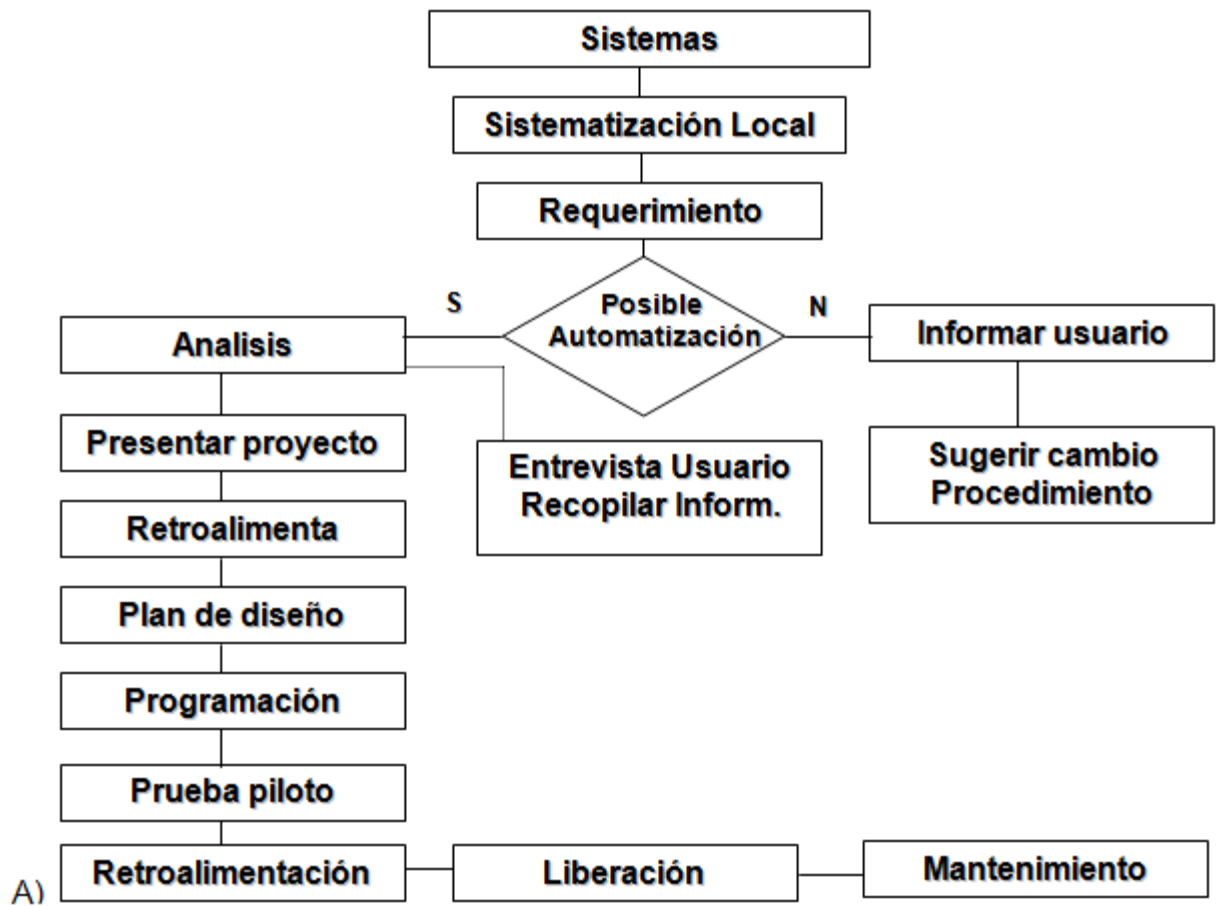


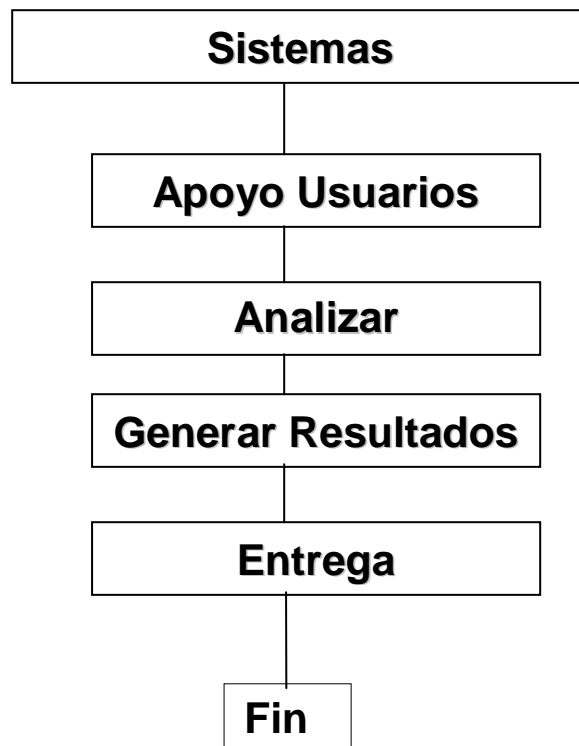
## Funciones de Area

### A) DIAGRAMA DE INTERRELACIONES



## DESCRIPCION DE LOS DATOS A PROCESAR





**DEFINIR EL SOFTWARE (LENGUAJE DE PROGRAMACION  
A UTILIZAR)**

- A) VISUAL BASIC**
- B) SQL SERVER**
- C) SPREAD**
- D) ACCESS**
- E) OFFICE**
- F) WINDOWS XP SP 3**

## 2.3.4.- DISEÑO DETALLADO EN EL DICCIONARIO DE LA BASE DE DATOS.

Diseño de registros (longitud de registro, longitud de campos y llaves principales

**OFICIALIA MAYOR  
DIRECCION GENERAL DE RECURSOS MATERIALES Y SERVICIOS GENERALES  
DIRECCION DE OPERACIONES Y SERVICIOS  
SUBDIRECCION DE SEGURIDAD Y PROTECCION CIVIL  
DEPARTAMENTO DE PROTECCION CIVIL**

#iNUM!

```

Set dbData =
Residencia de la Base de Datos Workspaces(0).OpenDatabase("D:\P_CIVIL\INSUMO.mdb")
Set dbData =
Nombre de la Base de Datos Workspaces(0).OpenDatabase("D:\P_CIVIL\INSUMO.mdb")
    
```

<b>Nombre de la Tabla</b>		<b>CAT_ADM</b>			
<b>Contenido de Información</b>		<b>Catálogo de Administraciones Internas</b>			
<b>Campo</b>	<b>Nombre</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Descripción</b>	<b>Comentarios</b>
1	CVE_ADM	ENTERO	2	Clave de la Administración	Llave de la Tabla
2	DES_ADM	TEXTO	100	Descripción de la Administración	

<b>Nombre de la Tabla</b>		<b>CAT_AGTI</b>			
<b>Contenido de Información</b>		<b>Catálogo de Importancia de los Trámites</b>			
<b>Campo</b>	<b>Nombre</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Descripción</b>	<b>Comentarios</b>
1	CVE_AGTI	ENTERO	2	Clave de la AGTI	Llave de la Tabla
2	DES_AGTI	TEXTO	25	Descripción de la AGTI	

<b>Nombre de la Tabla</b>		<b>CAT_LOCAL</b>			
<b>Contenido de Información</b>		<b>Catálogo de Tipos de Embargo</b>			
<b>Campo</b>	<b>Nombre</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Descripción</b>	<b>Comentarios</b>
1	CVE_LOCAL	ENTERO	2	Clave del LOCAL	Llave de la Tabla
2	DES_LOCAL	TEXTO	20	Descripción del LOCAL	



Nombre de la Tabla		CAT_SUBAD			
Contenido de Información		Catálogo de Tipos de Suspensión			
Campo	Nombre	Tipo	Tamaño	Descripción	Comentarios
1	CVE_SUBAD	ENTERO	2	Clave de la SUBAD	Llave de la Tabla
2	DES_SUBAD	TEXTO	20	Descripción de la SUBAD	

Nombre de la Tabla		CAT_DEPTO			
Contenido de Información		Catálogo de Estatus de la Suspensión			
mpo	Nombre	Tipo	Tamaño	Descripción	Comentarios
1	CVE_DEPTO	ENTERO	2	Clave del DEPTO	Llave de la Tabla
2	DES_DEPTO	TEXTO	20	Descripción del DEPTO	

Nombre de la Tabla		CAT_TITULAR			
Contenido de Información		Catálogo de Contribuyentes			
Campo	Nombre	Tipo	Tamaño	Descripción	Comentarios
1	NOM_TITULAR	TEXTO	100	Nombre del Contribuyente	Llave de la Tabla
2	CVE_TITULAR	ENTERO	2	Clave del Revisor que tiene el Asunto	

Nombre de la Tabla		CAT_ARTICULO			
Contenido de Información		Catálogo de Personal Revisor			
Campo	Nombre	Tipo	Tamaño	Descripción	Comentarios
1	CVE_ARTICULO	ENTERO	2	Clave del ARTICULO	Llave de la Tabla
2	NOM_ARTICULO	TEXTO	30	Nombre del ARTICULO	

Nombre de la Tabla		CAT_USU			
Contenido de Información		Catálogo de Usuarios del Sistema			
Campo	Nombre	Tipo	Tamaño	Descripción	Comentarios
1	CVE_USU	ENTERO	2	No. de Usuario	Llave de la Tabla  1 = ADMINISTRADOR 2 = USUARIO 3 = OPERATIVO 4 = CONSULTA
2	NOM_USU	TEXTO	40	Nombre del Usuario	
3	CLA_USU	TEXTO	8	Clave del Usuario	
4	PAS_USU	TEXTO	6	Password del Usuario	
5	NIV_USU	ENTERO	2	Nivel de Acceso del Usuario	
6	CVE_REV	ENTERO	2	Clave de Catalogo de Revisores	

Nombre de la Tabla		CAT_UNIDAD			
Contenido de Información		Catálogo de Resoluciones			
Campo	Nombre	Tipo	Tamaño	Descripción	Comentarios
2	CVE_ADMON	ENTERO	2	Clave del Tipo de ADMON	Llave de la Tabla
	CVE_UNIDAD	ENTERO	2	Clave de la UNIDAD	Llave de la Tabla
	DES_UNIDAD	TEXTO	25	Descripción de la UNIDAD	

nombre de la Tabla		CAT_INSTRUMENTAL			
Contenido de Información		Catálogo de Areas Operativas			
Campo	Nombre	Tipo	Tamaño	Descripción	Comentarios
1	CVE_USUARIOI	ENTERO	2	Clave del Area USUARIOI	Llave de la Tabla
2	DES_INSTRUMENTAL	TEXTO	25	Descripción del Area INSTRUMENTAL	
3	TIT_INSTRUMENTAL	TEXTO	50	Titular del Area INSTRUMENTAL	

Nombre de la Tabla		PAS_ASI			
Contenido de Información		Tabla de Paso para Reporte de Asignados			
Campo	Nombre	Tipo	Tamaño	Descripción	Comentarios
1	ASUNTO	TEXTO	10	Descripción del Asunto	
2	NUMERO	TEXTO	25	Número de Asunto	
3	CONTRIBUYENTE	TEXTO	100	Nombre del Contribuyente	
4	FECHA	FECHA	10	Fecha de Asignación	
5	USUARIO	TEXTO	50	Usuario	
6	NOMBRE	TEXTO	100	Nombre del Revisor	

Nombre de la Tabla		CONT_FOLIOS			
Contenido de Información		INVENTARIO DE RECURSOS DE INSTRUMENTAL			
Campo	Nombre	Tipo	Tamaño	Descripción	Comentarios
1	NUM_MED	TEXTO	25	Número de FOLIO	Llave de la Tabla
	NOM_CON	TEXTO	100	Nombre del SOLICITANTE	
3	CVE_REV	ENTERO	2	Clave de Revisor	SI, NO
4	FEC_ASI	DATE	10	Fecha de ENTRADA	
5	CVE_RES	ENTERO	2	Clave de la ASIGNACION	
6	TRA_FIR	TEXTO	2	Firma	
7	OFL_FIR	TEXTO	25	No. de Oficio de FIRMA	
8	FEC_FIR	DATE	10	Fecha salida	
11	CVE_AOP	ENTERO	2	Clave del Area Operativa a que se Asigna	
13	FEC_AOP	DATE	10	Fecha de Asignación	
17	FEC_CAP	DATE	10	Fecha de Captura	

Nombre de la Tabla		INV_ENTRADAS INVENTARIO DE ENTRADAS DE CAMBIOS			
Contenido de Información					
Campo	Nombre	Tipo	Tamaño	Descripción	Comentarios
1	NUM_FOLIO	TEXTO	10	Número de Recurso de entrada	Llave de la Tabla
2	CVE_ADMON	ENTERO	10	Clave de la Administración	
3	NOM_ADMON	TEXTO	100	nombre de la administracion	
4	CVE_LOCAL	ENTERO	10	clave de la local	
5	NOM_LOCAL	TEXTO	100	nombre de la local	
6	CVE_AREA	ENTERO	10	clave de la area	
7	NOM_AREA	TEXTO	100	nombre del area	
8	CVE_DEPTO	ENTERO	10	clave del depto	
9	NOM_DEPTO	TEXTO	100	nombre del depto	
10	CVE_PEDIDO	ENTERO	10	clave del pedido	
11	NOM_PEDIDO	TEXTO	100	nombre del pedido	
12	CVE_INSUMO	ENTERO	10	clave del insumo	
13	NOM_INSUMO	TEXTO	100	nombre del insumo	
14	CVE_INSTRUMENTAL	ENTERO	10	clave del instrumental	
15	NOM_INSTRUMENTAL	TEXTO	100	nombre del instrumental	
16	CVE_TITULO	ENTERO	10	clave del titulo del servicio	
17	NOM_TITULO	TEXTO	100	nombre del titulo del servicio	
18	CVE_SURTET	ENTERO	10	clave de surte	
19	NOM_SURTE	TEXTO	100	nombre de surte	
20	FECHA_ENTRADA	DATE		fecha de cambio	
21	CAT_ENTRA	ENTERO	10	cantidad de instrumental que se cambia	
22	CVE_USUARIO	ENTERO	10	clave del usuario	
23	NOM_USUARIO	TEXTO	100	nombre del usuario	

Nombre de la Tabla		INV_SALIDAS INVENTARIO DE ENTRADAS DE INSTRUMENTAL			
Contenido de Información					
Campo	Nombre	Tipo	Tamaño	Descripción	Comentarios
1	NUM_FOLIO	TEXTO	10	Número de Recurso de entrada	Llave de la Tabla
2	CVE_ADMON	ENTERO	10	Clave de la Administración	
3	NOM_ADMON	TEXTO	100	nombre de la administraciion	
4	CVE_LOCAL	ENTERO	10	clave de la local	
5	NOM_LOCAL	TEXTO	100	nombre de la local	
6	CVE_AREA	ENTERO	10	clave de la area	
7	NOM_AREA	TEXTO	100	nombre del area	
8	CVE_DEPTO	ENTERO	10	clave del depto	
9	NOM_DEPTO	TEXTO	100	nombre del depto	
10	CVE_PEDIDO	ENTERO	10	clave del pedido	
11	NOM_PEDIDO	TEXTO	100	nombre del pedido	
12	CVE_INSUMO	ENTERO	10	clave del insumo	
13	NOM_INSUMO	TEXTO	100	nombre del insumo	
14	CVE_INSTRUMENTAL	ENTERO	10	clave del instrumental	
15	NOM_INSTRUMENTAL	TEXTO	100	nombre del instrumental	
16	CVE_TITULO	ENTERO	10	clave del titulo del servicio	
17	NOM_TITULO	TEXTO	100	nombre del titulo del servicio	
18	CVE_SURTET	ENTERO	10	clave de surte	
19	NOM_SURTE	TEXTO	100	nombre de surte	
20	FECHA_ENTRADA	DATE		fecha de salida del instrumental	
21	CAT_ENTRA	ENTERO	10	cantidad de instrumental que sale	
22	CVE_USUARIO	ENTERO	10	clave del usuario	
23	NOM_USUARIO	TEXTO	100	nombre del usuario	

Base de datos: creación de la base de datos.

```
/* CREAMER_BASE.SQL=====
*/
/*
/* ARCHIVO: CREAMER_BASE.SQL VERSION 1.0 */
/* DATABASE NAME: INSTRUMENTAL PROTECCION CIVIL
*/
/* DBMS NAME: MICROSOFT SQL SERVER 6.X */
/* CREATED ON: 25/06/2010 3:25 PM */
/* SISTEMA: CONTROL DE INSTRUMENTAL Y SERVICIOS
*/
/*
/*
/* =====
*/
```

DUMP TRAN MASTER WITH NO\_LOG

USE MASTER

DECLARE @VDEVNO INT

```
IF NOT EXISTS (SELECT * FROM SYSDEVICES WHERE
NAME='CONTROL_INSTRUMENTAL_DAT')
BEGIN
SELECT @VDEVNO = (MIN (LOW)/0X01000000) +1 FROM SYSDEVICES D1
WHERE LOW/0X01000000 BETWEEN 0 AND 254 AND
NOT EXISTS (SELECT * FROM SYSDEVICES D2 WHERE
D2.LOW/0X01000000 = (D1.LOW/0X01000000) +1)
DISK INIT
NAME = 'CONTROL_INSTRUMENTAL_DAT',
PHYSNAME = 'C:\MSSQL\DATA\CONTROL_INSTRUMENTAL_DAT.DAT',
VDEVNO = @VDEVNO,
SIZE = 20000
END
```

```
IF NOT EXISTS (SELECT * FROM SYSDEVICES WHERE
NAME='CONTROL_INSTRUMENTAL_LOG')
BEGIN
SELECT @VDEVNO = (MIN(LOW)/0X01000000)+1 FROM SYSDEVICES D1
WHERE LOW/0X01000000 BETWEEN 0 AND 254 AND
NOT EXISTS (SELECT * FROM SYSDEVICES D2 WHERE
D2.LOW/0X01000000 =(D1.LOW/0X01000000)+1)
```

```
DISK INIT
NAME = 'CONTROL_INSTRUMENTAL_LOG',
```

```

PHYSNAME = 'C:\MSSQL\DATA\CONTROL_INSTRUMENTAL_LOG.DAT',

VDEVNO = @VDEVNO, SIZE = 5000
END
/* =====*/
/* */
/* CREANDO LA BASE DE DATOS CONTROL DE INSTRUMENTAL */
/* */
/* =====*/

USE MASTER

DECLARE
@MESSAGE CHAR(255),
@BEGAN DATETIME

SELECT @BEGAN = GETDATE()

IF EXISTS ( SELECT NAME
            FROM SYSDATABASES
            WHERE NAME IN ('INSTRUMENTAL') )
BEGIN
    DROP DATABASE INSTRUMENTAL
END

CREATE DATABASE INSTRUMENTAL
ON CONTROL_INSTRUMENTAL_DAT = 20000
LOG ON CONTROL_INSTRUMENTAL_LOG = 5000

SELECT @MESSAGE =
'TIME (IN MINUTES:SECONDS) TO CREATE DATABASE ' +
CONVERT( CHAR(2), DATEDIFF(SS,@BEGAN,GETDATE())/60 ) +
':' +
CONVERT( CHAR(2), DATEDIFF(SS,@BEGAN,GETDATE())%60 )

RAISERROR(@MESSAGE, 10, 1)

GO

EXEC SP_DISKDEFAULT MASTER, DEFAULTOFF
GO

```

```

/* ===== */
/*          */
/*  CREAT LOG          */
/*          */
/* ===== */

```

```

USE MASTER
IF EXISTS (SELECT NAME FROM SYSUSERS WHERE
NAME='INSTRUMENTAL') BEGIN
EXEC SP_DROPUSER USU_INSTRUMENTAL
END
IF EXISTS (SELECT NAME FROM SYSLOGINS WHERE
NAME='USU_INSTRUMENTAL')
BEGIN
USE INSTRUMENTAL
EXEC SP_DROPALIAS USU_INSTRUMENTAL
EXEC SP_DROPLOGIN USU_INSTRUMENTAL
END

```

```

USE MASTER
EXEC SP_ADDLOGIN USU_INSTRUMENTAL,SURDF13,INSTRUMENTAL
GO

```

```

USE INSTRUMENTAL
EXEC SP_ADDALIAS USU_INSTRUMENTAL,DBO
GRANT ALL TO DBO
GO
SP_DBOPTION INSTRUMENTAL, 'SELECT INTO/BULKCOPY', TRUE
GO

```

```

SP_DBOPTION TEMPDB, 'SELECT INTO/BULKCOPY', TRUE
GO
SP_DBOPTION TEMPDB, 'TRUNC. LOG ON CHKPT.', TRUE

```

```

GO

```

```

/* BORRA USER Y LOGIN PARA BD TEMPDB */
USE TEMPDB
IF EXISTS (SELECT NAME FROM TEMPDB..SYSUSERS WHERE
NAME='TEMPDB')
BEGIN

```

```

EXEC SP_DROPUSER TEMPDB
END
GO
IF EXISTS (SELECT NAME FROM MASTER..SYSLOGINS WHERE
NAME='TEMPDB')
BEGIN
EXEC SP_DROPLOGIN TEMPDB
END
GO

/* CREA USER Y LOGIN PARA BD TEMPDB */
EXEC SP_ADDLOGIN TEMPDB, TEMPDB, TEMPDB
GO
USE TEMPDB
EXEC SP_ADDALIAS TEMPDB,DBO
GRANT ALL TO DBO
GO

/* ASOCIA TAMBI,N TEMPDB A INSTRUMENTAL */
USE INSTRUMENTAL
EXEC SP_ADDALIAS TEMPDB,DBO
40
GRANT ALL TO DBO
40
GO

```

Creación de la estructura de la base de datos:

```

/* SISTEMA: CONTROL DE INSTRUMENTALES (CONTROL_INSTRUMENTALS)
*/
/* BASE DE DATOS: INSTRUMENTALES */
/* FECHA DE CREACION:26/JUNIO/2010 */
/* AUTORES: JUAN TOMAS MEDINA ESPINOZA */
USE INSTRUMENTALS
CREATE TABLE CAT_AGTI
(CVE_AGTI INTEGER NOT NULL,
NOM_AGTI VARCHAR(100) NOT NULL,
CONSTRAINT PK_CAT_AGTI PRIMARY KEY CLUSTERED (CVE_AGTI)
)
GO

CREATE TABLE IDE_AGTI
(CVE_AGTI INTEGER NOT NULL,
TIT_AGTI VARCHAR(50) NOT NULL,
CONSTRAINT PK_IDE_AGTI PRIMARY KEY CLUSTERED (CVE_AGTI)
)
GO

```



```

CREATE TABLE CAT_LOCAL
    (CVE_LOCAL          INTEGER NOT NULL,
    CVE_AGTI           INTEGER NOT NULL,
    NOM_LOCAL          VARCHAR(60) NOT NULL,
    CONSTRAINT PK_CAT_LOCAL PRIMARY KEY CLUSTERED
(CVE_LOCAL)
)
GO

```

```

ALTER TABLE CAT_LOCAL ADD
    CONSTRAINT CVE_LOCAL FOREIGN KEY
    (IDE_AGTI)
    REFERENCES IDE_AGTI(CVE_AGTI)
GO

```

```

CREATE TABLE CAT_ADMON
    (CVE_ADMON          INTEGER NOT NULL,
    CVE_LOCAL           INTEGER NOT NULL,

    CVE_AGTI           INTEGER NOT NULL,
    NOM_ADMON          VARCHAR(100) NOT NULL,
    CONSTRAINT PK_CAT_ADMON PRIMARY KEY CLUSTERED
(CVE_ADMON)
)
GO

```

```

CREATE TABLE CAT_SUBAD
    (CVE_SUBAD          INTEGER NOT NULL,
    CVE_ADMON           INTEGER NOT NULL,
    CVE_LOCAL           INTEGER NOT NULL,
    CVE_AGTI           INTEGER NOT NULL,
    NOM_SUBAD          VARCHAR(50) NOT NULL,
    CONSTRAINT PK_CAT_SUBAD PRIMARY KEY CLUSTERED
(CVE_SUBAD)
)
GO

```

```

ALTER TABLE CAT_SUBAD ADD

    CONSTRAINT CVE_SUBAD FOREIGN KEY
    (CAT_ADMON)
    REFERENCES CAT_ADMON(CVE_ADMON)
GO

```

```

CREATE TABLE CAT_DEPTO
    (CVE_DEPTO          INTEGER NOT FULL,
    CVE_SUBAD           INTEGER NOT NULL,

```

```

        CVE_ADMON            INTEGER NOT NULL,
        CVE_LOCAL            INTEGER NOT NULL,
        CVE_AGTI             INTEGER NOT NULL,
        NOM_DEPTO            VARCHAR(50) NOT NULL,
        CONSTRAINT PK_CAT_DEPTO PRIMARY KEY CLUSTERED
(CVE_DEPTO)
)
GO

ALTER TABLE CAT_DEPTO ADD
    CONSTRAINT CVE_DEPTO FOREIGN KEY
    (CAT_SUBAD)
    REFERENCES CAT_SUBAD(CVE_SUBAD)
GO

CREATE TABLE CAT_TITULAR
    (CVE_TITULAR            INTEGER NOT FULL,
    CVE_DEPTO              INTEGER NOT FULL,
    CVE_SUBAD              INTEGER NOT NULL,
    CVE_ADMON              INTEGER NOT NULL,
    CVE_LOCAL              INTEGER NOT NULL,
    CVE_AGTI               INTEGER NOT NULL,
    NOM_TITULAR            VARCHAR(60) NOT NULL,
    CONSTRAINT PK_CAT_TITULAR PRIMARY KEY CLUSTERED
(CVE_TITULAR)
)
GO

ALTER TABLE CAT_TITULAR ADD
    CONSTRAINT CVE_TITULAR FOREIGN KEY
    (CAT_DEPTO)
    REFERENCES CAT_DEPTO(CVE_DEPTO)
GO

CREATE TABLE CAT_ARTICULO
    (CVE_ARTICULO          INTEGER NOT NULL,
    NOM_ARTICULO           VARCHAR(15) NOT NULL,
    CONSTRAINT PK_CAT_INSTRUMENTAL PRIMARY KEY CLUSTERED
(CVE_ARTICULO)
)
GO

CREATE TABLE CAT_UNIDAD
    (CVE_UNIDAD            INTEGER NOT NULL,
    NOM_UNIDAD             VARCHAR(5) NOT NULL,
    CONSTRAINT PK_CAT_UNIDAD PRIMARY KEY CLUSTERED
(CVE_UNIDAD)
)

```

```

)
GO

CREATE TABLE CAT_INSTRUMENTAL
    (CVE_INSTRUMENTAL      INTEGER NOT NULL,
    CVE_ARTICULO           INTEGER NOT NULL,
    CVE_UNIDAD             INTEGER NOT NULL,
    NOM_INSTRUMENTAL       VARCHAR(30) NOT NULL,
    MOD_INSTRUMENTAL       VARCHAR(30) NOT NULL,
    CON_FOLIOS             INTEGER NOT NULL,
    CONSTRAINT PK_CAT_INSTRUMENTAL PRIMARY KEY CLUSTERED
(CVE_INSTRUMENTAL)
)
GO
ALTER TABLE CAT_INSTRUMENTAL ADD
    CONSTRAINT CVE_INSTRUMENTAL FOREIGN KEY
    (CAT_ARTICULO)
    REFERENCES CAT_ARTICULO(CVE_ARTICULO)
GO

CREATE TABLE CON_FOLIOS
    (NUM_CAJA              INTEGER NOT NULL,
    CVE_INSTRUMENTAL       INTEGER NOT NULL,
    CVE_ARTICULO           INTEGER NOT NULL,
    FOL_INICIO             VARCHAR(8) NOT NULL,
    FOL_FINAL              VARCHAR(8) NOT NULL,
    CONSTRAINT PK_CON_FOLIOS PRIMARY KEY CLUSTERED
(NUM_CAJA)
)
GO

ALTER TABLE CON_FOLIOS
    CONSTRAINT NUM_CAJA FOREIGN KEY
    (CAT_INSTRUMENTAL)
    REFERENCES CAT_INSTRUMENTAL(CVE_INSTRUMENTAL)
GO

CREATE TABLE CAN_FOLIOS
    (NUM_CAJA              INTEGER NOT NULL,
    CVE_INSTRUMENTAL       INTEGER NOT NULL,
    CVE_ARTICULO           INTEGER NOT NULL,
    FOL_INICIO             VARCHAR(8) NOT NULL,
    FOL_FINAL              VARCHAR(8) NOT NULL,
    MOT_CANCELA            LONG NOT NULL,
    FEC_CANCELA            DATETIME NOT NULL,

```

```
        CONSTRAINT PK_CON_FOLIOS PRIMARY KEY CLUSTERED
(NUM_CAJA)
)
GO
```

```
ALTER TABLE CAN_FOLIOS ADD
    CONSTRAINT NUM_CAJA FOREIGN KEY
    (CAT_INSTRUMENTAL)
    REFERENCES CAT_INSTRUMENTAL(CVE_INSTRUMENTAL)
GO
```

```
CREATE TABLE INV_AGTI
    (NUM_REP          INTEGER NOT NULL,

    FEC_INICIO       DATETIME NOT NULL,
    FEC_FINAL        DATETIME NOT NULL,
    CVE_INSTRUMENTAL INTEGER NOT NULL,
    CVE_ARTICULO     INTEGER NOT NULL,
    CVE_UNIDAD       INTEGER NOT NULL,
    INV_INICIAL      INTEGER NOT NULL,
    INV_FINAL        INTEGER NOT NULL,
    CONSTRAINT PK_INV_AGTI PRIMARY KEY CLUSTERED (NUM_REP)
)
GO
```

```
ALTER TABLE INV_AGTI ADD
    CONSTRAINT NUM_REP FOREIGN KEY
    (CAT_INSTRUMENTAL)
    REFERENCES CAT_INSTRUMENTAL(CVE_INSTRUMENTAL)
GO
```

```
CREATE TABLE INV_ALMACEN
    (CVE_INSTRUMENTAL INTEGER NOT NULL,
    CVE_ARTICULO      INTEGER NOT NULL,

    CVE_UNIDAD        INTEGER NOT NULL,
    CAN_ALMACEN       INTEGER NOT NULL    CONSTRAINT
    PK_INV_ALMACEN PRIMARY KEY CLUSTERED (CVE_INSTRUMENTAL)
)
GO
```

```
ALTER TABLE INV_ALMACEN ADD
    CONSTRAINT CVE_INSTRUMENTAL FOREIGN KEY
    (CAT_ARTICULO)
    REFERENCES CAT_ARTICULO(CVE_ARTICULO)
GO
```

```

CREATE TABLE INV_ENTRADA
    (CVE_INSTRUMENTAL          INTEGER NOT NULL,
    CVE_ARTICULO              INTEGER NOT NULL,
    CVE_UNIDAD                INTEGER NOT NULL,
    CAN_ENTRADA              INTEGER NOT NULL,
    FEC_ENTRADA              DATETIME NOT NULL,
    CONSTRAINT PK_INV_ENTRADA PRIMARY KEY CLUSTERED
(CVE_INSTRUMENTAL)
)
GO

```

```

ALTER TABLE INV_ENTRADA ADD
    CONSTRAINT CVE_INSTRUMENTAL FOREIGN KEY
    (CAT_ARTICULO)
    REFERENCES CAT_ARTICULO(CVE_ARTICULO)
GO

```

```

CREATE TABLE INV_SALIDA
    (CVE_INSTRUMENTAL          INTEGER NOT NULL,
    CVE_ARTICULO              INTEGER NOT NULL,
    CVE_UNIDAD                INTEGER NOT NULL,
    CVE_DEPTO                INTEGER NOT NULL,
    CVE_SUBAD                INTEGER NOT NULL,
    CVE_ADMON                INTEGER NOT NULL,
    CVE_LOCAL                INTEGER NOT NULL,
    CVE_AGTI                 INTEGER NOT NULL,
    CAN_SALIDA              INTEGER NOT NULL,

    NOM_RECIBE              VARCHAR(30) NOT NULL,
    FEC_SALIDA              DATETIME NOT NULL,
    CONSTRAINT PK_INV_SALIDA PRIMARY KEY CLUSTERED
(CVE_INSTRUMENTAL)
)
GO

```

```

ALTER TABLE INV_SALIDA ADD
    CONSTRAINT CVE_INSTRUMENTAL FOREIGN KEY
    (CAT_ARTICULO)
    REFERENCES CAT_ARTICULO(CVE_ARTICULO)GO

    (CVE_INSTRUMENTAL          INTEGER NOT NULL,
    CVE_ARTICULO              INTEGER NOT NULL,
    CVE_UNIDAD                INTEGER NOT NULL,
    CAN_ENTRADA              INTEGER NOT NULL,
    FEC_ENTRADA              DATETIME NOT NULL,
    CVE_AGTI                 INTEGER NOT NULL,
    OFI_AGTI                 VARCHAR(25) NOT NULL,

```

```

        CONSTRAINT PK_ENT_APOYO PRIMARY KEY CLUSTERED
(CVE_INSTRUMENTAL)
)
GO

```

```

ALTER TABLE ENT_APOYO ADD
    CONSTRAINT CVE_INSTRUMENTAL FOREIGN KEY
(CAT_ARTICULO)
REFERENCES CAT_ARTICULO(CVE_ARTICULO)
GO

```

```

CREATE TABLE SAL_APOYO
(CVE_INSTRUMENTAL          INTEGER NOT NULL,
CVE_ARTICULO              INTEGER NOT NULL,
CVE_UNIDAD                INTEGER NOT NULL,
CAN_SALIDA                INTEGER NOT NULL,
FEC_SALIDA                DATETIME NOT NULL,
CVE_AGTI                  INTEGER NOT NULL,
OFI_AGTI                  VARCHAR(25) NOT NULL,

```

```

        CONSTRAINT PK_SAL_APOYO PRIMARY KEY CLUSTERED
(CVE_INSTRUMENTAL)
)
GO

```

```

ALTER TABLE SAL_APOYO ADD
    CONSTRAINT CVE_INSTRUMENTAL FOREIGN KEY
(CAT_ARTICULO)
REFERENCES CAT_ARTICULO(CVE_ARTICULO)
GO

```

```

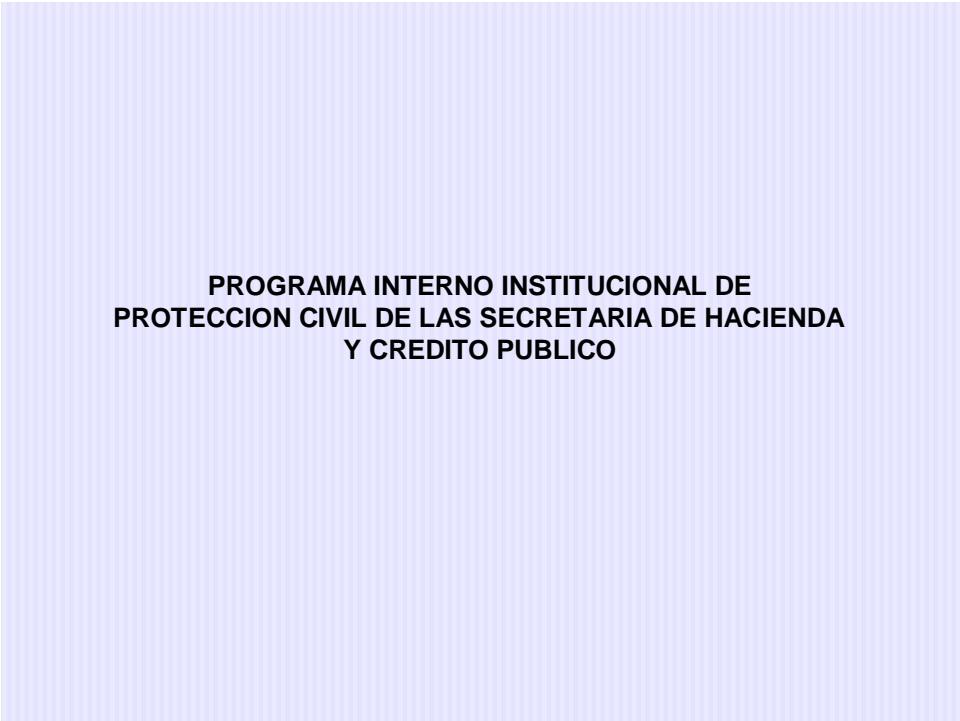
CREATE TABLE CAT_USUARIO
(CVE_AGTI                  INTEGER NOT NULL,
CVE_USUARIO               INTEGER NOT NULL,
RFC_USUARIO               VARCHAR(13) NOT NULL,
NOM_USUARIO               VARCHAR(50) NOT NULL,
FEC_ALTA                  DATETIME NOT NULL,
FEC_BAJA                  DATETIME NOT NULL,
UID_USUARIO               VARCHAR(8) NOT NULL,
PAS_USUARIO               VARCHAR(6) NOT NULL,
NIV_USUARIO               INTEGER NOT NULL,
    CONSTRAINT PK_CAT_USUARIO PRIMARY KEY CLUSTERED
(PAS_USUARIO)
)
GO

```

### **2.3.5.- PROGRAMACIÓN Y DESARROLLO**

La tecnología utilizada el tratamiento automático de la información en los procesos internos del de la función del Leguaje de Programación se encuentra construido Visual Basic estudio 6, con plataforma en base de datos SQL Server y acces, así como Erwin en la construcción de tablas.

**A continuación se da a conocer la estructura del programa a través de una serie de formularios, los cuales describen en forma evolutiva el desarrollo del presente sistema.**

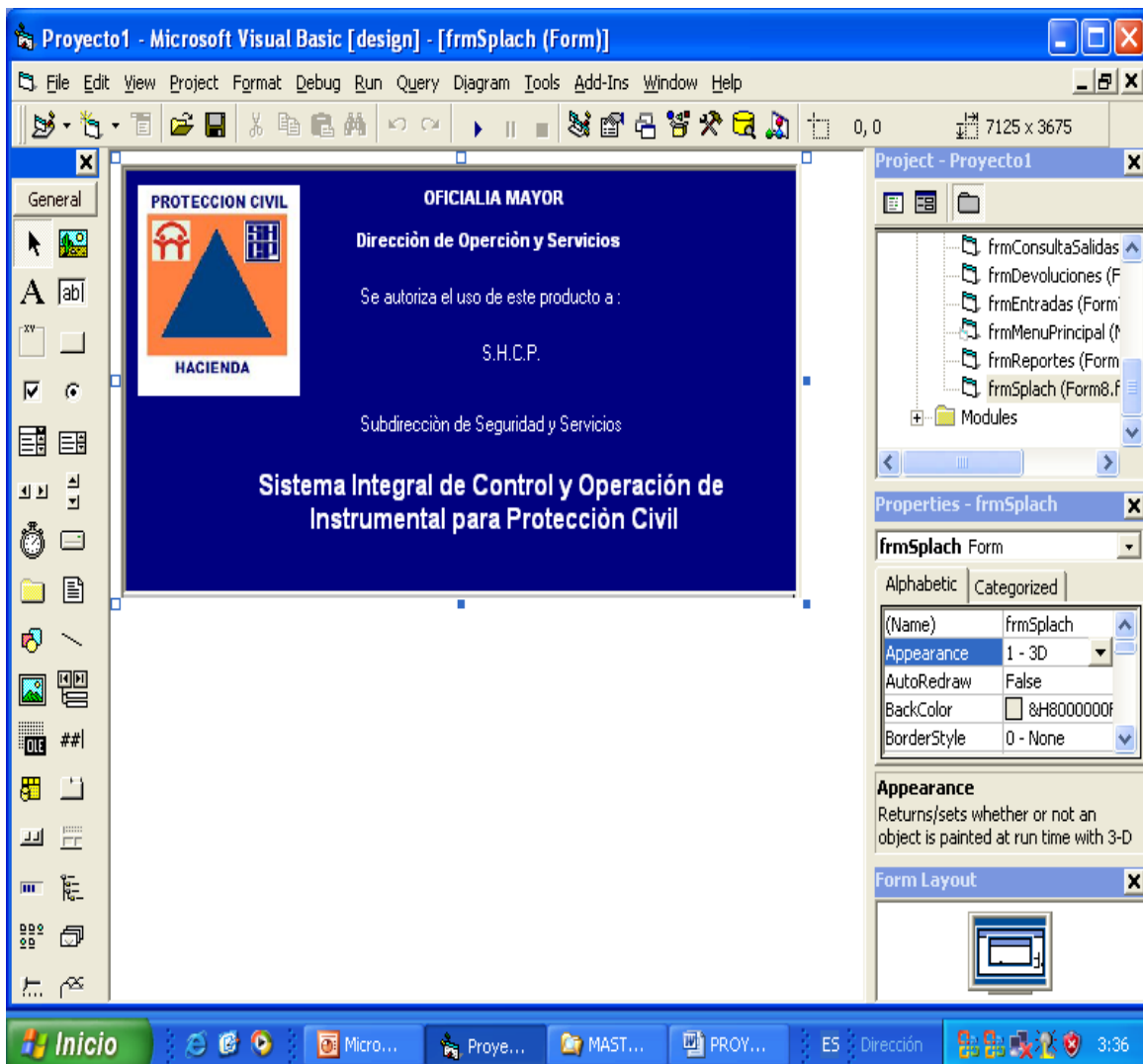


**PROGRAMA INTERNO INSTITUCIONAL DE  
PROTECCION CIVIL DE LAS SECRETARIA DE HACIENDA  
Y CREDITO PUBLICO**

**FASES DE DESARROLLO  
CARACTERISTICAS DEL  
SISTEMA**

- **SPLACH Y CODIGO**

El sistema tiene como inicio de presentación un splash, en donde se identifica el nombre del sistema y el logotipo de la empresa, así como el código fuente correspondiente a la presente pantalla.





Código splash.  
Option Explicit  
Sistema: Control de instrumental de proteccion civil

Subsistema:

Descripción: Pantalla de Presentación del Sistema  
Autor: juan tomas  
Fecha: febrero/2010

```
Formas: frmSplash.Frm
Private Sub Degradado (vForm As PictureBox)
    Dim iContador As Integer
    vForm.DrawStyle = vbInsideSolid
    vForm.DrawMode = vbCopyPen
    vForm.ScaleMode = vbPixels
    vForm.DrawWidth = 2
    vForm.ScaleHeight = 256
    For iContador = 0 To 255
vForm.Line (0, iContador)-(Screen.Width, _
    iContador - 1), RGB(0, 0, 255 - iContador), B
    Next iContador
End Sub

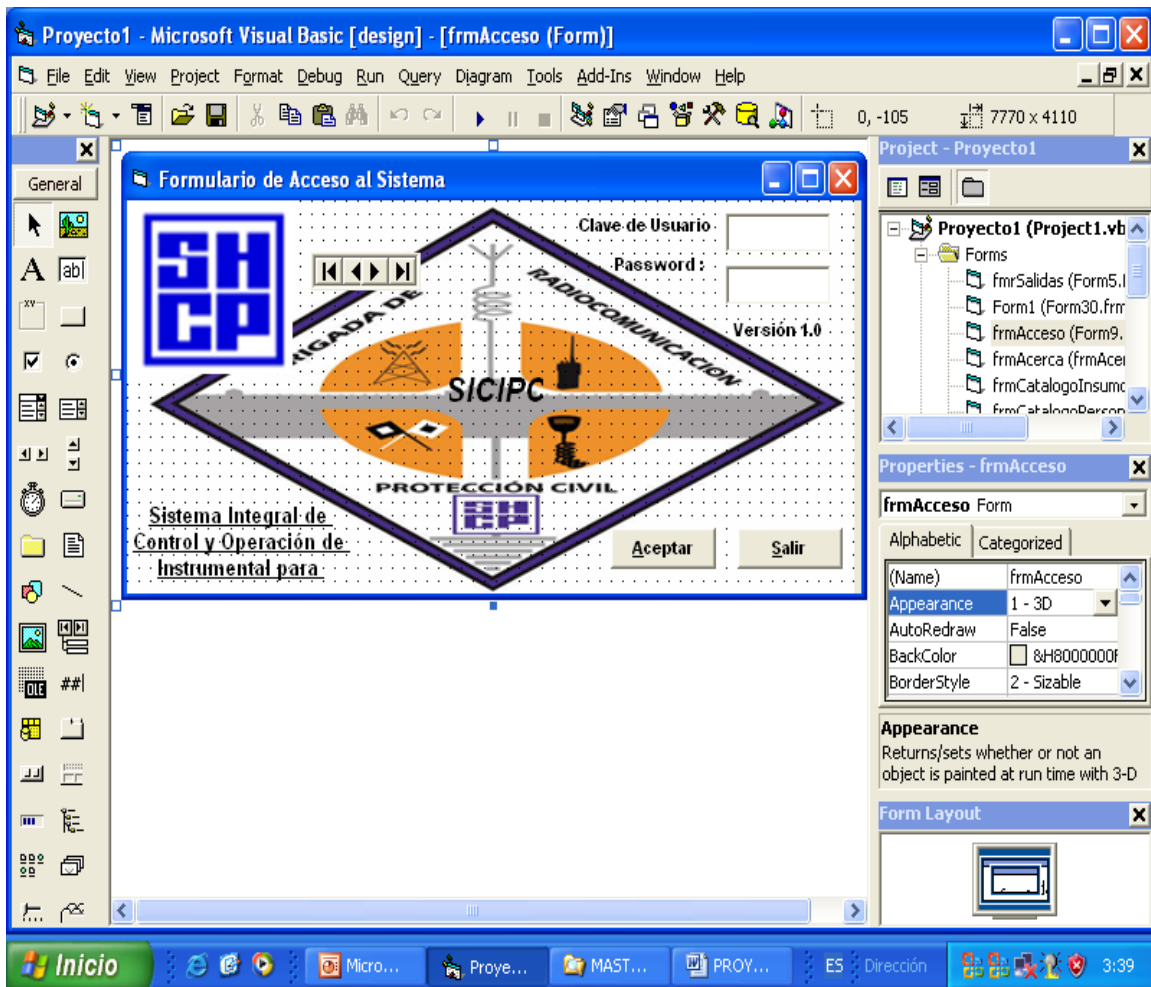
Private Sub Picture1_Paint()
    Call Degradado(Picture1)
End Sub

Private Sub Timer1_Timer()
    frmAcceso.Show
Unload Me
End Sub
```

## **FASES DE DESARROLLO CARACTERISTICAS DEL SISTEMA**

- **VENTANA DE ACCESO  
CONTRASEÑA Y PASSWORD Y  
CODIGO**

En esta pantalla de muestra el acceso al sistema, a traves de una clave y password de usuario.



Código a utilizar para el formulario de acceso al sistema:

Option Explicit

```
Private Sub cmdAceptar_Click()
On Error GoTo Error_Grabando
If Clave_Usuario <> "" And Pass_Usuario <> "" Then
    dbsql = "SELECT * FROM Cat_Usuarios WHERE CVE_Usuario=" &
Clave_Usuario & " And Pass_Usuario=" & _
    Pass_Usuario & ""
    Set rsAux = dbData.OpenRecordset(dbsql)
    If rsAux.RecordCount > 0 Then
        frmMenuPrincipal.Show
    Unload frmAcceso
Else
    MsgBox "Password de Usuario no Valido", 64, "Atención"
    Clave_Usuario = Empty
    Pass_Usuario = Empty
Clave_Usuario.SetFocus
End If
End If
On Error GoTo 0
Exit Sub
Error_Grabando:
If Err.Number <> 0 Then

    Call Mensajes_de_Error("EL PASSWORD QUE DIGITO USTED NO ES
VALIDO")
End If
End Sub
Private Sub Form_Load()
Set dbData = Workspaces(0).OpenDatabase("D:\P_CIVIL\INSUMO.mdb")
End Sub
Private Sub cmdSalir_Click()
    dbData.Close
End
End Sub
Private Sub CLAVE_USUARIO_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then Pass_Usuario.SetFocus
If KeyAscii = 8 Then Exit Sub

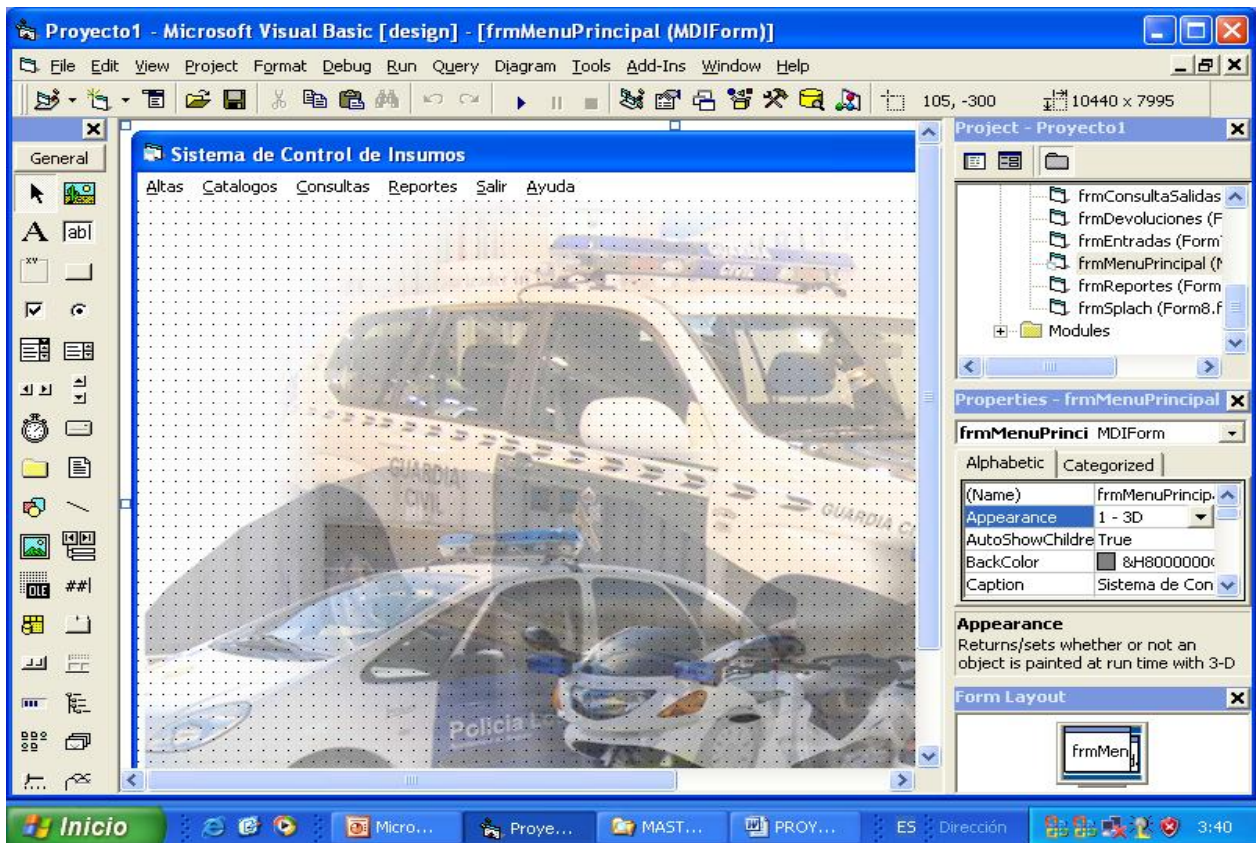
Select Case Chr(KeyAscii)
Case "{", "}", "[", "]", "|", "", "¿", "?", "¡", "!", "=", "/", "\", ""
    KeyAscii = 0
Case Else
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
```

```
End Select
End Sub
Private Sub PASS_USUARIO_Keypress(KeyAscii As Integer)
If KeyAscii = 13 Then cmdAceptar.SetFocus
If KeyAscii = 8 Then Exit Sub
Select Case Chr(KeyAscii)
Case "{", "}", "[", "]", "|", " ", "¿", "?", "¡", "!", "=", "/", "\", ""
KeyAscii = 0
Case Else
KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Select
End Sub
```

## **FASES DE DESARROLLO CARACTERISTICAS DEL SISTEMA**

- **PANTALLA DEL MENU PRINCIPAL Y CODIGO**

Pantalla del menú principal del sistema:



Código en el formulario del menú principal:

```

Private Sub ayuda_Click()
frmAcerca.Show
End Sub
Private Sub cambio_Click()
frmDevoluciones.Show
End Sub
Private Sub cambios_Click()
frmConsultaDevoluciones.Show
End Sub
Private Sub entradas_Click()
frmEntradas.Show

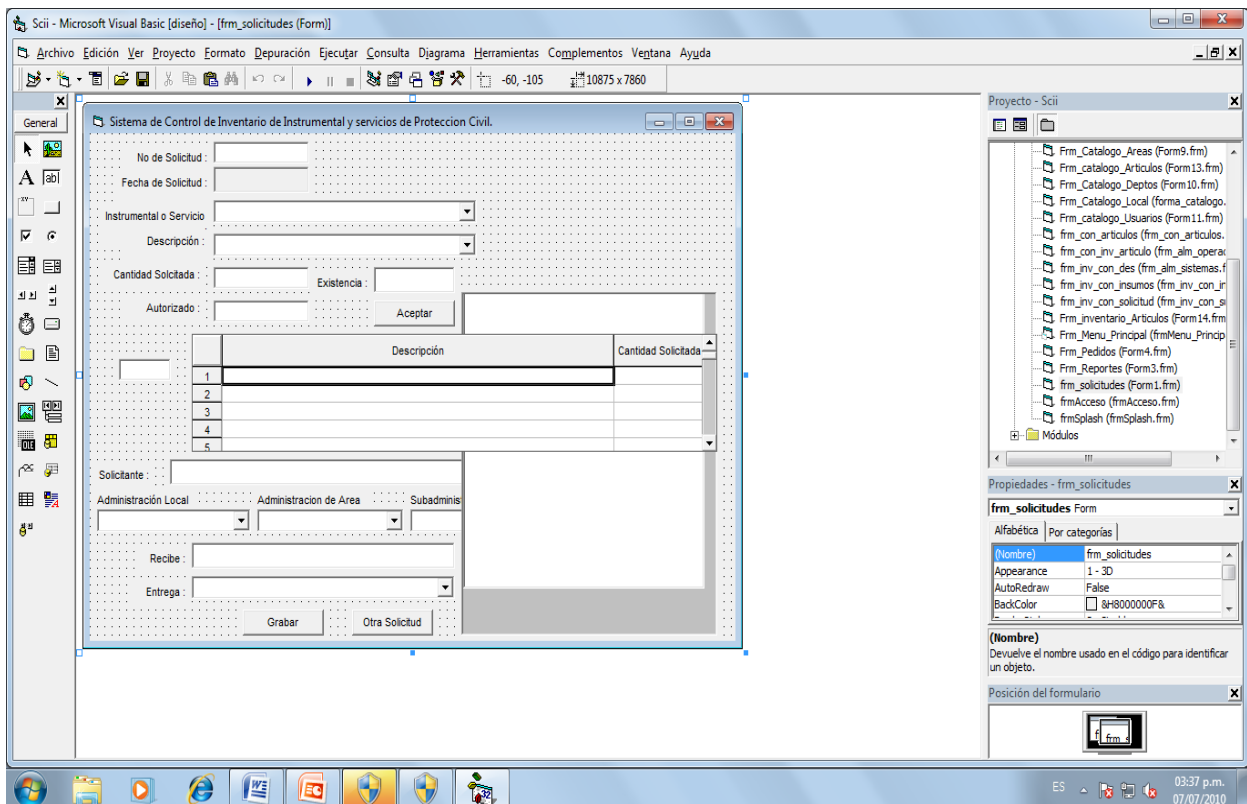
```

```
End Sub
Private Sub entradas1_Click()
frmConsultaEntrada.Show
End Sub
Private Sub entradas2_Click()
Form1.Show
End Sub
Private Sub insumos_Click()
frmCatalogoInsumos.Show
End Sub
Private Sub MDIForm_Load()
    Set dbData = OpenDatabase("D:\P_CIVIL\INSUMO.mdb")
End Sub
Private Sub personal_Click()
frmCatalogoPersonal.Show
End Sub
Private Sub reportes_Click()
frmReportes.Show
End Sub
Private Sub salida_Click()
fmrSalidas.Show
End Sub
Private Sub salidas1_Click()
frmConsultaSalidas.Show
End Sub
Private Sub salir_Click()
Unload Me
End Sub
Private Sub usuarios_Click()
frmCatalogoUsuario.Show
End Sub
```

## FASES DE DESARROLLO CARACTERISTICAS DEL SISTEMA

- **PANTALLA PRINCIPAL DE CAPTURA DE DATOS Y CODIGO**

En esta pantalla se muestran las opciones que se utilizan para ingresar Equipo, Instrumental y Servicios Al sistema.





Codigo Autilizar en la pantalla de ingreso:

Option Explicit

```
Private Sub Cbo_admon_Click()  
If Cbo_local <> "" And Cbo_admon <> "" Then  
    sSQL = "SELECT * FROM cat_area WHERE c_cve_local=" &  
Val(Mid(Cbo_local, 101)) & _  
        " AND c_cve_admon=" & Val(Mid(Cbo_admon, 101))  
    Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,  
dbSQLPassThrough)  
    Cbo_area.Clear  
    If Not recData.BOF And Not recData.EOF Then  
        Do While Not recData.EOF  
            Cbo_area.AddItem recData("d_nom_area") & String(100 -  
Len(recData("d_nom_area")), " ") & recData("c_cve_area")  
  
            recData.MoveNext  
        Loop  
    End If  
    Cbo_area.SetFocus  
End If  
End Sub
```

```
Private Sub cbo_area_Click()  
If Cbo_local <> "" And Cbo_area <> "" Then  
    sSQL = "SELECT * FROM cat_depto WHERE c_cve_local=" &  
Val(Mid(Cbo_local, 101)) & _  
        " AND c_cve_area=" & Val(Mid(Cbo_area, 101))  
    Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,  
dbSQLPassThrough)  
    Cbo_depto.Clear  
    If Not recData.BOF And Not recData.EOF Then  
        Do While Not recData.EOF  
            Cbo_depto.AddItem recData("d_nom_depto") & String(100 -  
Len(recData("d_nom_depto")), " ") & recData("c_cve_depto")  
            recData.MoveNext  
        Loop  
    End If  
    Cbo_depto.SetFocus  
End If  
End Sub
```

```
Private Sub Cbo_descripcion_Click()  
If Cbo_descripcion <> "" Then  
    txt_cantidad_Sol.SetFocus
```

```
End If
End Sub
```

```
Private Sub cbo_entrega_Click()
    cmd_grabar.SetFocus
End Sub
```

```
Private Sub cbo_Local_Click()
If Cbo_local <> "" Then
    sSQL = "SELECT * FROM cat_admon WHERE c_cve_local=" &
Val(Mid(Cbo_local, 101))
```

```
    Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
```

```
    Cbo_admon.Clear
    If Not recData.BOF And Not recData.EOF Then
        Do While Not recData.EOF
            Cbo_admon.AddItem recData("d_nom_admon") & String(100 -
Len(recData("d_nom_admon")), " ") & recData("c_cve_admon")
            recData.MoveNext
        Loop
    End If
    Cbo_admon.SetFocus
End If
End Sub
```

```
'Este proceso llena el spread de aceptar'
Private Sub cmd_aceptar_Click()
```

```
    On Error GoTo Error_Aceptar
    If iRen = 1 Then
        sSQL = "DELETE FROM tt_inv_temporal WHERE c_cve_almacen=" &
iAlmacen & " AND c_cve_personal=" & iPersona
        wspEspacio.BeginTrans
        dbData.Execute sSQL, dbSQLPassThrough
        wspEspacio.CommitTrans
    End If
    sSQL = "INSERT INTO
tt_inv_temporal(c_cve_almacen,c_cve_personal,c_cve_articulo,c_num_articulo,n_
cantidad) VALUES(" & iAlmacen & "," & iPersona & ","
    vsp_sol.Row = iRen
    vsp_sol.Col = 1
    vsp_sol.Text = txt_num_sol
    vsp_sol.Col = 2
    vsp_sol.Text = msd_fecha_sol
    vsp_sol.Col = 3
```

```

vsp_sol.Text = Cbo_Articulo
sSQL = sSQL & Val(Mid(Cbo_Articulo, 100)) & ","
vsp_sol.Col = 4
vsp_sol.Text = Cbo_descripcion
sSQL = sSQL & Val(Mid(Cbo_descripcion, 100)) & ","
vsp_sol.Col = 5
vsp_sol.Text = txt_cantidad_Sol
vsp_sol.Col = 6
vsp_sol.Text = txt_existencia
vsp_sol.Col = 7
vsp_sol.Text = txt_Autorizo
sSQL = sSQL & txt_Autorizo & ")"
wspEspacio.BeginTrans
    dbData.Execute sSQL, dbSQLPassThrough
wspEspacio.CommitTrans
    iRen = iRen + 1
Cbo_Articulo = Empty
    Cbo_descripcion = Empty
    txt_cantidad_Sol = Empty
    txt_existencia = Empty
    txt_Autorizo = Empty
    Cbo_Articulo.SetFocus
Exit Sub
Error_Aceptar:
    If Err.Number <> 0 Then
Call Mensajes_de_Error("Grabando en Temporal")
    End If
End Sub

' *****
' Este proceso graba las solicitudes diarias.
' *****

Private Sub cmd_grabar_Click()
    Dim iSol As Integer, iArti As Integer, iDes As Integer, iCanti As Integer
    If iRen >= 1 Then
        On Error GoTo Error_Grabando
        sSQL = "SELECT * FROM mm_inv_solicitud WHERE c_cve_almacen=" &
iAlmacen
        Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot)
        If Not recData.BOF And Not recData.EOF Then
            recData.MoveLast
            iSol = recData.RecordCount + 1
        Else
            iSol = 1
        End If
        wspEspacio.BeginTrans

```

```

sSQL = "INSERT INTO
mm_inv_solicitud(c_cve_almacen,c_num_folio,f_fec_sol," & _
"c_cve_local,c_cve_admon,c_cve_area,c_cve_depto,c_cve_usuario,d_nom_solicit
a," & _
"d_nom_recibe,c_cve_personal) VALUES(" & iAlmacen & "," & iSol & _
"," & Fecha_Ingles(msd_fecha_sol) & "," & Val(Mid(Cbo_local, 100)) & _
"," & Val(Mid(Cbo_admon, 100)) & "," & Val(Mid(Cbo_area, 100)) & _
"," & Val(Mid(Cbo_depto, 100)) & "," & Text1 & "," & txt_solicitante & "," & _
txt_recibe & "," & Val(Mid(cbo_entrega, 100)) & ")
dbData.Execute sSQL, dbSQLPassThrough
wspEspacio.CommitTrans
wspEspacio.BeginTrans
For iPaso = 1 To iRen - 1
vsp_sol.Row = iPaso
sSQL = "INSERT INTO mm_inv_salidas(c_cve_almacen,c_num_folio," & _
"c_cve_articulo,c_num_articulo,f_fec_salida,n_cantidad) " & _
"VALUES (" & iAlmacen & "," & iSol & ","
vsp_sol.Col = 3
sSQL = sSQL & Val(Mid(vsp_sol.Text, 100)) & ","
iArti = Val(Mid(vsp_sol.Text, 100))
vsp_sol.Col = 4
sSQL = sSQL & Val(Mid(vsp_sol.Text, 100)) & ","
iDes = Val(Mid(vsp_sol.Text, 100))
vsp_sol.Col = 2
sSQL = sSQL & Fecha_Ingles(vsp_sol.Text) & ","
vsp_sol.Col = 5
sSQL = sSQL & Val(vsp_sol.Text) & ")"
iCanti = Val(vsp_sol.Text)
dbData.Execute sSQL, dbSQLPassThrough
sSQL = "UPDATE mm_inv_gral SET n_cantidad=n_cantidad-" & iCanti & _
" WHERE c_cve_almacen=" & iAlmacen & " AND c_cve_articulo=" & _
iArti & " AND c_num_articulo=" & iDes
dbData.Execute sSQL, dbSQLPassThrough
Next
wspEspacio.CommitTrans

sSQL = "DELETE FROM tt_inv_temporal WHERE c_cve_almacen=" &
iAlmacen & " AND c_cve_personal=" & iPersona
wspEspacio.BeginTrans
dbData.Execute sSQL, dbSQLPassThrough
wspEspacio.CommitTrans
cmd_grabar.Enabled = False
End If
If iSol > 0 Then
txt_num_sol = iSol
MsgBox "No. de Solicitud asignado al pedido [ " & iSol & " ]", vbOKOnly +
vbInformation, "Registro de Solicitudes"

```

```

End If
    Screen.MousePointer = vbdefault
    On Error GoTo 0
    Exit Sub
Error_Grabando:
    If Err.Number <> 0 Then
    Call Mensajes_de_Error("Grabando solicitud")
    End If
End Sub

```

```

Private Sub cmd_otra_sol_Click()
    iRen = 1
    sSQL = "DELETE FROM tt_inv_temporal WHERE c_cve_almacen=" &
iAlmacen & " AND c_cve_personal=" & iPersona
    wspEspacio.BeginTrans
    dbData.Execute sSQL, dbSQLPassThrough
    wspEspacio.CommitTrans
    'otra
    vsp_sol.MaxRows = 0
    vsp_sol.MaxRows = 100
    ' msd_fecha_sol = " / / "
    txt_num_sol = Empty
    Cbo_Articulo = Empty
    Cbo_descripcion = Empty
    txt_cantidad_Sol = Empty
    txt_existencia = Empty
    txt_Autorizo = Empty
    txt_solicitante = Empty
    Cbo_local = Empty
    Cbo_admon = Empty

```

```

Cbo_area = Empty
    Cbo_depto = Empty
    txt_recibe = Empty
    cbo_entrega = Empty
    msd_fecha_sol.SetFocus
    cmd_grabar.Enabled = True
    iRen = 1
End Sub

```

"proceso de aceptar espread'

```

Private Sub cmd_salir_Click()
Unload Me

```

End Sub

```
Private Sub Cbo_articulo_Click()
    If Cbo_Articulo <> "" Then
        sSQL = "SELECT * FROM mm_inv_articulos WHERE c_cve_articulo=" &
Val(Mid(Cbo_Articulo, 101))
        Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
        Cbo_descripcion.Clear
        If Not recData.BOF And Not recData.EOF Then
            Do While Not recData.EOF
                Cbo_descripcion.AddItem recData("d_nom_articulo") & String(100 -
Len(recData("d_nom_articulo")), " ") & recData("c_num_articulo")
                recData.MoveNext
            Loop
        End If
        Cbo_descripcion.SetFocus
    End If
End Sub
```

```
Private Sub Form_Activate()
    Cbo_Articulo.SetFocus
End Sub
```

```
Private Sub Form_Load()
```

```
Screen.MousePointer = vbHourglass
! *****
! * Esta rutina busca los articulos para mostrarlos en el combo Cbo_articulos *
! *****

    sSQL = "SELECT * FROM cat_articulos ORDER BY c_cve_articulo"
    Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
    If Not recData.BOF And Not recData.BOF Then
        Do While Not recData.EOF
            Cbo_Articulo.AddItem recData("d_des_articulo") & String(100 -
Len(recData("d_des_articulo")), " ") & recData("c_cve_articulo")
            recData.MoveNext
        Loop
    End If
    sSQL = "SELECT * FROM cat_personal WHERE c_cve_personal < 99 ORDER
BY c_cve_personal"
```

```

Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
If Not recData.EOF And Not recData.EOF Then
    Do While Not recData.EOF
        cbo_entrega.AddItem recData("d_nom_personal") & String(100 -
Len(recData("d_nom_personal")), " ") & recData("c_cve_personal")
        recData.MoveNext
    Loop
End If
msd_fecha_sol = Format(Now, "dd/mm/yyyy")
Screen.MousePointer = vbdefault
iRen = 1
End Sub

Private Sub List1_Click()
    If List1 <> "" Then
        sSQL = "SELECT
cat_usuarios.c_cve_usuario,cat_usuarios.d_nom_usuario,cat_local.c_cve_local,ca
t_local.d_nom_local," & _

"cat_admon.c_cve_admon,cat_admon.d_nom_admon,cat_area.c_cve_area,cat_ar
ea.d_nom_area,cat_depto.c_cve_depto,cat_depto.d_nom_depto " & _
        "FROM (((cat_usuarios INNER JOIN cat_local ON
cat_usuarios.c_cve_local=" & _
        "cat_local.c_cve_local) INNER JOIN cat_admon ON
(cat_usuarios.c_cve_local=" & _
        "cat_admon.c_cve_local) AND
(cat_usuarios.c_cve_admon=cat_admon.c_cve_admon)) " & _
        "INNER JOIN cat_area ON
(cat_usuarios.c_cve_area=cat_area.c_cve_area) AND " & _
"(cat_usuarios.c_cve_admon = cat_area.c_cve_admon) and
(cat_usuarios.c_cve_local=" & _
"cat_area.c_cve_local)) INNER JOIN cat_depto ON (cat_usuarios.c_cve_depto="
& _
        "cat_depto.c_cve_depto) AND
(cat_usuarios.c_cve_area=cat_depto.c_cve_area) AND " & _
        "(cat_usuarios.c_cve_admon=cat_depto.c_cve_admon) AND
(cat_usuarios.c_cve_local=" & _
        "cat_depto.c_cve_local) WHERE (((cat_usuarios.c_cve_usuario)=" &
Val(Mid(List1, 101, 3)) & ")))"
        Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot)
        If Not recData.EOF And Not recData.EOF Then
            txt_solicitante = Trim(Mid(List1, 1, 100))
            Text1 = recData("c_cve_usuario")
            Cbo_local = recData("d_nom_local") & String(100 -
Len(recData("d_nom_local")), " ") & recData("c_cve_local")

```

```

        Cbo_admon = recData("d_nom_admon") & String(100 -
Len(recData("d_nom_admon")), " ") & recData("c_cve_admon")
        Cbo_area = recData("d_nom_area") & String(100 -
Len(recData("d_nom_area")), " ") & recData("c_cve_area")
        Cbo_depto = recData("d_nom_depto") & String(100 -
Len(recData("d_nom_depto")), " ") & recData("c_cve_depto")
        Nombres.Visible = False
        txt_recibe.SetFocus
    End If
        Screen.MousePointer = vbdefault
    End If
End Sub

Private Sub msd_fecha_sol_KeyPress(KeyAscii As Integer)
    If KeyAscii = 8 Then Exit Sub
    If KeyAscii = 13 Then
        msd_fecha_sol_LostFocus
    End If
End Sub

Private Sub msd_fecha_sol_LostFocus()
    If msd_fecha_sol <> " / / " Then
        If Format(msd_fecha_sol, "mm/dd/yyyy") > Format(Now, "mm/dd/yyyy") Or
InStr(1, Format(msd_fecha_sol, "mm/dd/yyyy"), " ") > 0 Or Not
IsDate(msd_fecha_sol) Or Mid(msd_fecha_sol, 4, 2) > 12 Or Mid(msd_fecha_sol,
7, 4) < 2001 Then
'        MsgBox "La Fecha es Incorrecta ", 48, At
MsgBox "La Fecha no es Valida, Escribala Correctamente", vbOKOnly +
vblInformation, "Solicitud de Insumos"
            msd_fecha_sol.SetFocus
            msd_fecha_sol = limpia
        Else
            Cbo_Articulo.SetFocus
        End If
    Else
        Cbo_Articulo.SetFocus
    End If
    Exit Sub
End Sub

Private Sub txt_Autorizo_KeyPress(KeyAscii As Integer)

    If KeyAscii = 13 Then
        If Cbo_local = "" Then
            cmd_aceptar.SetFocus
        Else
            cmd_aceptar.SetFocus
        End If
    End If
End Sub

```



```

        End If
    End If
    If KeyAscii = 8 Then Exit Sub
    If Chr(KeyAscii) < "0" Or Chr(KeyAscii) > "9" Then
        KeyAscii = 0
    End If
End Sub

Private Sub txt_cantidad_Sol_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then txt_cantidad_Sol_LostFocus
    If KeyAscii = 8 Then Exit Sub
    If Chr(KeyAscii) < "0" Or Chr(KeyAscii) > "9" Then
        KeyAscii = 0
    End If
End Sub

Private Sub txt_cantidad_Sol_LostFocus()
    Dim iExiste_Almacen As Integer
    Dim iExiste_Temporal As Integer
    If Val(txt_cantidad_Sol) > 0 Then
        On Error GoTo Error_Buscando_Existencia
        Screen.MousePointer = vbHourglass
        sSQL = "SELECT n_cantidad as Existe FROM mm_inv_gral WHERE
c_cve_almacen=" & iAlmacen & " AND " & _
"c_cve_articulo=" & Val(Mid(Cbo_Articulo, 101)) & " AND " & _
        "c_num_articulo=" & Val(Mid(Cbo_descripcion, 101))
        Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot)

        If Not recData.BOF And Not recData.EOF Then
            iExiste_Almacen = recData("Existe")
        Else
            iExiste_Almacen = 0
        End If
        sSQL = "SELECT n_cantidad as Existe FROM tt_inv_temporal WHERE
c_cve_almacen=" & iAlmacen & " AND " & _
"c_cve_articulo=" & Val(Mid(Cbo_Articulo, 101)) & " AND " & _
        "c_num_articulo=" & Val(Mid(Cbo_descripcion, 101))
        Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot)
        If Not recData.BOF And Not recData.EOF Then
            iExiste_Temporal = recData("Existe")
        Else
            iExiste_Temporal = 0
        End If
        If iExiste_Almacen = 0 Then
            txt_existencia = Empty
            txt_cantidad_Sol = Empty
            txt_cantidad_Sol.SetFocus
        End If
    End If
End Sub

```

```

        Screen.MousePointer = vbdefault
        On Error GoTo 0
        Exit Sub
    End If
    If iExiste_Temporal > 0 Then
        If iExiste_Almacen > iExiste_Temporal Then
            txt_existencia = iExiste_Almacen - iExiste_Temporal
            txt_Autorizo.SetFocus
            Screen.MousePointer = vbdefault
            On Error GoTo 0
            Exit Sub
        End If
    Else
        txt_existencia = iExiste_Almacen
        txt_Autorizo.SetFocus
        Screen.MousePointer = vbdefault
        On Error GoTo 0
        Exit Sub
    End If
End If

Exit Sub
Error_Buscando_Existencia:
    If Err.Number <> 0 Then
        Call Mensajes_de_Error("Buscando Existencia en Almacen")
    End If
End Sub

Private Sub txt_num_sol_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then msd_fecha_sol.SetFocus
    If KeyAscii = 8 Then Exit Sub
    If Chr(KeyAscii) < "0" Or Chr(KeyAscii) > "9" Then
        KeyAscii = 0
        If Not IsNumeric(txt_num_sol.Text) Then
            MsgBox "La Clave Debe Ser Numerica", 64, At
            txt_num_sol = ""
            txt_num_sol.SetFocus
        Else
            End If
        End If
    End If
End Sub

Private Sub txt_recibe_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then cbo_entrega.SetFocus
    If KeyAscii = 8 Then Exit Sub

```

```

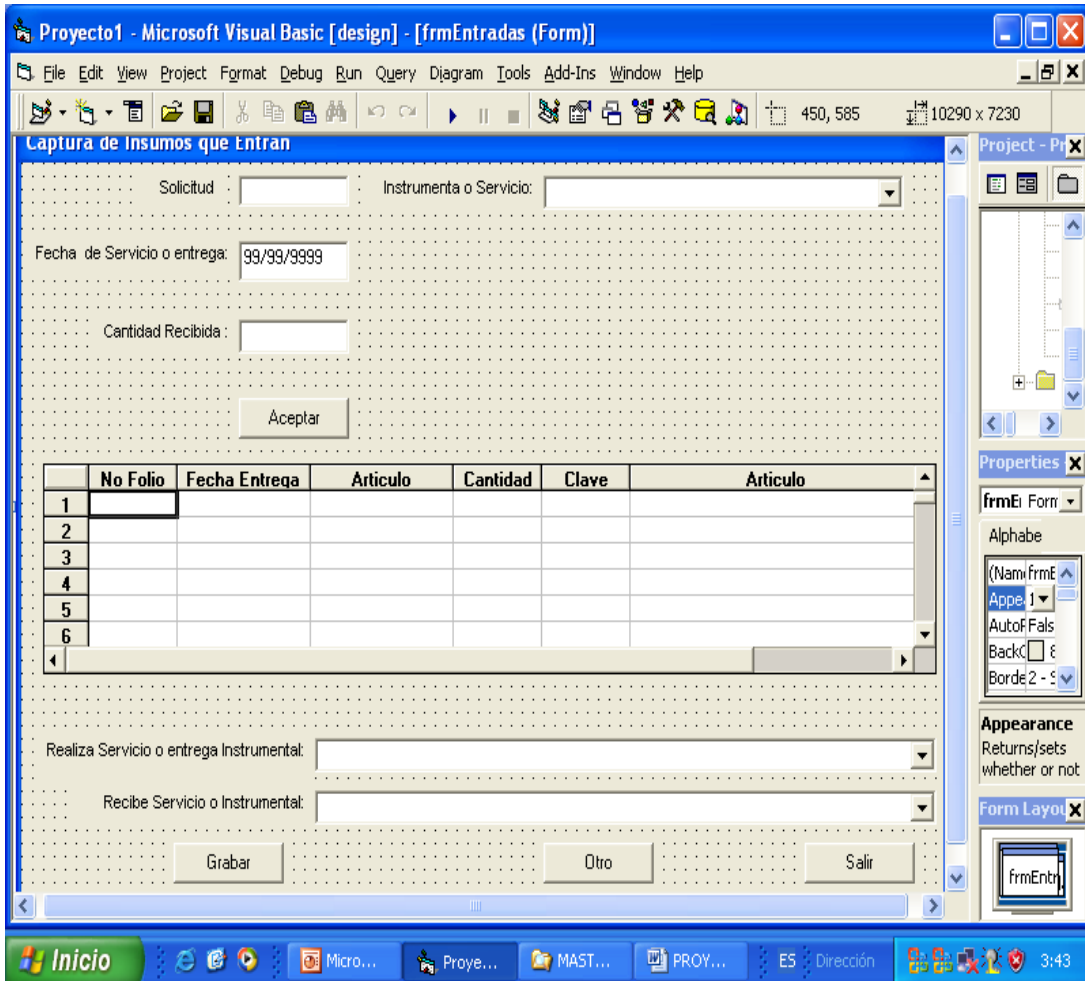
Select Case Chr(KeyAscii)
  Case "{", "}", "[", "]", "|", "`", "¿", "?", "¡", "!", "=", "/", "\", ""
    KeyAscii = 0
  Case Else
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Select
End Sub

Private Sub txt_solicitante_Change()
  If txt_solicitante <> "" Then
    sSQL = "SELECT * FROM cat_usuarios WHERE d_nom_usuario LIKE '" & _
      txt_solicitante & "' ORDER BY d_nom_usuario"
    Set rsAux = dbData.OpenRecordset(sSQL, dbOpenSnapshot)

    If Not rsAux.EOF And Not rsAux.EOF Then
      List1.Clear
      Do While Not rsAux.EOF
        List1.AddItem Trim(rsAux("d_nom_usuario")) & String(100 -
          Len(rsAux("d_nom_usuario")), " ") & Format(rsAux("c_cve_usuario"), "000") &
          Format(rsAux("c_cve_local"), "000") & Format(rsAux("c_cve_admon"), "000") &
          Format(rsAux("c_cve_area"), "000") & Format(rsAux("c_cve_depto"), "000")
        rsAux.MoveNext
      Loop
    End If
    Nombres.Visible = True
    List1.SetFocus
  End If
End Sub

```

En esta pantalla se muestra las opciones de solicitudes de equipo, instrumental o servicios por los usuarios:



Código a utilizar en la pantalla:

Option Explicit

```
Private Sub cmd_aceptar_Click()
    vsp_aceptar.Row = iRen + 1
    vsp_aceptar.Col = 1
    vsp_aceptar.Text = txt_pedido
    vsp_aceptar.Col = 2
    vsp_aceptar.Text = fecha_entrega
    vsp_aceptar.Col = 3
    vsp_aceptar.Text = Cbo_insumo
    vsp_aceptar.Col = 4
    vsp_aceptar.Text = txt_cantidad
    vsp_aceptar.Col = 5
    vsp_aceptar.Text = Val(Mid(Cbo_insumo, 51))
    vsp_aceptar.Col = 6
End Sub
```

```

    vsp_aceptar.Text = Val(Mid(Cbo_insumo, 15))
    iRen = iRen + 1
    Cbo_insumo = Empty
    entrego = Empty
    txt_cantidad = Empty
    Cbo_insumo.SetFocus
End Sub

Private Sub cmd_grabar_Click()
If iRen >= 1 Then
    For iPaso = 1 To iRen
        vsp_aceptar.Row = iPaso
        dbsql = "INSERT INTO
cat_inv_entradas(cve_pedido,cve_insumo,nom_insumo," & _
"fecha_entra,can_entra,nom_surte,nom_recibe) VALUES("
        vsp_aceptar.Col = 1
        dbsql = dbsql & Val(vsp_aceptar.Text) & ","
        vsp_aceptar.Col = 5
        dbsql = dbsql & Val(vsp_aceptar.Text) & ","
        vsp_aceptar.Col = 3
        dbsql = dbsql & vsp_aceptar.Text & ","
        vsp_aceptar.Col = 2
        dbsql = dbsql & Format(vsp_aceptar.Text, "dd/mm/yyyy") & ","
        vsp_aceptar.Col = 4
        dbsql = dbsql & Val(vsp_aceptar.Text) & ","
        dbsql = dbsql & entrego & ","
        dbsql = dbsql & recibe & ")")

        dbData.Execute dbsql
        iRen = iRen + 1

    Next
    cmd_grabar.Enabled = False
    cmd_Otro.SetFocus
End If
End Sub

Private Sub cmd_Otro_Click()
    dbsql = " select * From cat_inv_entradas "
    Set rsAux = dbData.OpenRecordset(dbsql)
    rsAux.MoveLast
    If rsAux.RecordCount = 0 Then
        txt_pedido = 1
        txt_pedido.Refresh
    Else
        txt_pedido = rsAux.RecordCount + 1
    End If
End Sub

```

```

        txt_pedido.Refresh
    End If
    vsp_aceptar.MaxRows = 0
    vsp_aceptar.MaxRows = 100
    fecha_entrega = " / / "
    Cbo_insumo = Empty
    entrega = Empty
    txt_cantidad = Empty
    entrega = Empty
    recibe = Empty
    Cbo_insumo.SetFocus
    cmd_grabar.Enabled = True
    iRen = 0
End Sub

Private Sub cmd_salir_Click()
Unload Me
End Sub
Private Sub fecha_entrega_KeyPress(KeyAscii As Integer)

    If KeyAscii = 8 Then Exit Sub
    If KeyAscii = 13 Then
        fecha_entrega_LostFocus
    
```

```

    End If
End Sub
Private Sub fecha_entrega_LostFocus()
On Error GoTo q

    If fecha_entrega <> limpia Then
        If Format(fecha_entrega, "mm/dd/yyyy") > Format(Now, "mm/dd/yyyy") Or
InStr(1, Format(fecha_entrega, "mm/dd/yyyy"), " ") > 0 Or Not
IsDate(fecha_entrega) Or Mid(fecha_entrega, 4, 2) > 12 Or Mid(fecha_entrega, 7,
4) < 2003 Then
            MsgBox "La Fecha es Incorrecta ", 48, At
            fecha_entrega.SetFocus
            fecha_entrega = limpia
        Else
            txt_cantidad.SetFocus
        End If
        Else
            cmd_aceptar.SetFocus

    End If
Exit Sub

q:
If Err.Number = 13 Then MsgBox " fecha incorrecta", , "Fecha":
fecha_entrega.SetFocus: fecha_entrega = limpia: Exit Sub
End Sub

Private Sub Form_Load()
'   Screen.MousePointer = vbHourglass
'   *****
' * Esta rutina busca los articulos para mostrarlos en el combo Cbo_articulos *
Set dbData = Workspaces(0).OpenDatabase("D:\P_CIVIL\INSUMO.mdb")
dbsql = "select * from cat_insumos order by cve_insumo"
Set rsAux = dbData.OpenRecordset(dbsql)
If rsAux.RecordCount > 0 Then
Do While Not rsAux.EOF
Cbo_insumo.AddItem rsAux("nom_insumo") & String(50 -
Len(rsAux("nom_insumo")), ".") & rsAux("cve_insumo")
rsAux.MoveNext
Loop
End If

```

```

dbsql = "select * from cat_personal order by cve_persona"
Set rsAux = dbData.OpenRecordset(dbsql)
If rsAux.RecordCount > 0 Then
Do While Not rsAux.EOF
entrego.AddItem rsAux("nom_persona") & String(50 -
Len(rsAux("nom_persona")), ".") & rsAux("cve_persona")
rsAux.MoveNext
Loop
End If
dbsql = "select * from cat_personal order by cve_persona"
Set rsAux = dbData.OpenRecordset(dbsql)
If rsAux.RecordCount > 0 Then
Do While Not rsAux.EOF
recibe.AddItem rsAux("nom_persona") & String(50 -
Len(rsAux("nom_persona")), ".") & rsAux("cve_persona")
rsAux.MoveNext
Loop
End If

```

```

dbsql = " select * From cat_inv_entradas  "
Set rsAux = dbData.OpenRecordset(dbsql)

```

```

If rsAux.RecordCount > 0 Then
rsAux.MoveLast

```

```

txt_pedido = rsAux.RecordCount + 1
Else
txt_pedido = 1
End If

```

```

' txt_clave = rsAux.RecordCount + 1
' Else
' txt_clave = 1

```

```

If rsAux.RecordCount = 0 Then
txt_pedido = 1
txt_pedido.Refresh
Else
txt_pedido = rsAux.RecordCount + 1

```

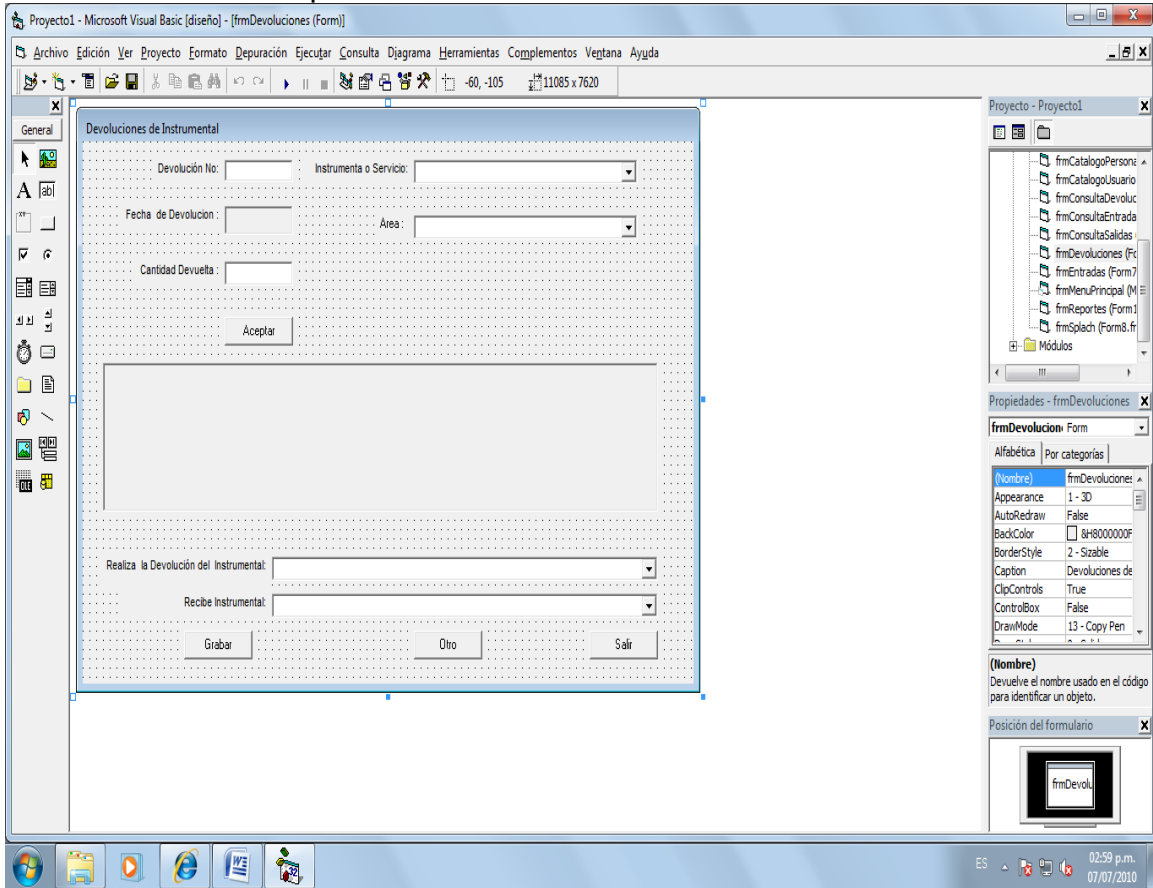


```
        txt_pedido.Refresh
    End If
    Exit Sub
End Sub
```

```
Private Sub txt_cantidad_recibida_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then cmd_aceptar.SetFocus
    If KeyAscii = 8 Then Exit Sub
    If Chr(KeyAscii) < "0" Or Chr(KeyAscii) > "9" Then
        KeyAscii = 0
    End If
End Sub
```

```
Private Sub txt_pedido_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then fecha_entrega.SetFocus
    If KeyAscii = 8 Then Exit Sub
    If Chr(KeyAscii) < "0" Or Chr(KeyAscii) > "9" Then
        KeyAscii = 0
        If Not IsNumeric(fecha_entrega.Text) Then
            MsgBox "La Clave Debe Ser Numerica", 64, At
            fecha_entrega = ""
            fecha_entrega.SetFocus
        End If
    Else
        End If
    End If
End Sub
```

Esta pantalla muestra las opciones en la devolucion de un instrumental o equipos defectuosos o no.Requerido:



Codigo a realizar para la funcion de la pantalla:

Option Explicit

```
Private Sub area_KeyPress(KeyAscii As Integer)
```

```
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
```

```
    If KeyAscii = 8 Then Exit Sub
```

```
    If KeyAscii = 13 Then
```

```
        recibe.SetFocus
```

```
    End If
```

End Sub

```
Private Sub cmd_aceptar_Click()  
    vsp_aceptar.Row = iRen + 1  
    vsp_aceptar.Col = 1  
    vsp_aceptar.Text = txt_pedido  
    vsp_aceptar.Col = 2  
    vsp_aceptar.Text = fecha_entrega  
    vsp_aceptar.Col = 3  
    vsp_aceptar.Text = Cbo_insumo  
    vsp_aceptar.Col = 4  
    vsp_aceptar.Text = txt_cantidad  
    vsp_aceptar.Col = 5  
    vsp_aceptar.Text = Val(Mid(Cbo_insumo, 51))  
    iRen = iRen + 1  
    Cbo_insumo = Empty  
    entrego = Empty  
    txt_cantidad = Empty  
    Cbo_insumo.SetFocus  
End Sub
```

```
Private Sub cmd_grabar_Click()  
    If iRen >= 1 Then  
        For iPaso = 1 To iRen  
            vsp_aceptar.Row = iPaso  
            dbsql = "INSERT INTO  
cat_inv_devoluciones(cve_pedido,cve_insumo,nom_insumo," & _  
                "fecha_entra,can_entra,nom_devuelve,nom_area,nom_recibe)  
VALUES("  
                vsp_aceptar.Col = 1  
                dbsql = dbsql & Val(vsp_aceptar.Text) & ","  
                vsp_aceptar.Col = 5  
                dbsql = dbsql & Val(vsp_aceptar.Text) & ","  
                vsp_aceptar.Col = 3  
                dbsql = dbsql & vsp_aceptar.Text & ","  
                vsp_aceptar.Col = 2  
                dbsql = dbsql & Format(vsp_aceptar.Text, "dd/mm/yyyy") & ","  
                vsp_aceptar.Col = 4  
                dbsql = dbsql & Val(vsp_aceptar.Text) & ","  
                dbsql = dbsql & entrego & ","  
            
```

```

dbsql = dbsql & area & ", "
      dbsql = dbsql & recibe & ")"

      dbData.Execute dbsql
      iRen = iRen + 1
    Next
    cmd_grabar.Enabled = False
    cmd_Otro.SetFocus
  End If
End Sub

Private Sub cmd_Otro_Click()
  dbsql = " select * From cat_inv_devoluciones  "
  Set rsAux = dbData.OpenRecordset(dbsql)
  rsAux.MoveLast
  If rsAux.RecordCount = 0 Then
    txt_pedido = 1
    txt_pedido.Refresh
  Else
    txt_pedido = rsAux.RecordCount + 1
    txt_pedido.Refresh
  End If
  vsp_aceptar.MaxRows = 0
  vsp_aceptar.MaxRows = 100
  fecha_entrega = " / / "
  Cbo_insumo = Empty
  entrega = Empty
  txt_cantidad = Empty
  Cbo_insumo.SetFocus
  cmd_grabar.Enabled = True
  iRen = 0
End Sub

Private Sub cmd_salir_Click()
Unload Me
End Sub

```

```

Private Sub entrego_KeyPress(KeyAscii As Integer)
KeyAscii = Asc(UCase(Chr(KeyAscii)))
If KeyAscii = 8 Then Exit Sub
If KeyAscii = 13 Then
area.SetFocus
End If
End Sub

```

```

Private Sub fecha_entrega_KeyPress(KeyAscii As Integer)

```

```

    If KeyAscii = 8 Then Exit Sub
If KeyAscii = 13 Then
    fecha_entrega_LostFocus
End If
End Sub

```

```

Private Sub fecha_entrega_LostFocus()
On Error GoTo q

```

```

    If fecha_entrega <> limpia Then
        If Format(fecha_entrega, "mm/dd/yyyy") > Format(Now, "mm/dd/yyyy") Or
InStr(1, Format(fecha_entrega, "mm/dd/yyyy"), " ") > 0 Or Not
IsDate(fecha_entrega) Or Mid(fecha_entrega, 4, 2) > 12 Or Mid(fecha_entrega, 7,
4) < 2003 Then
            MsgBox "La Fecha es Incorrecta ", 48, At
            fecha_entrega.SetFocus
            fecha_entrega = limpia

```

```

Else
    txt_cantidad.SetFocus
End If
Else
    cmd_aceptar.SetFocus

```

```

    End If
Exit Sub

```

```

q:
If Err.Number = 13 Then MsgBox " fecha incorrecta", , "Fecha":
fecha_entrega.SetFocus: fecha_entrega = limpia: Exit Sub
End Sub

```

```

Private Sub Form_Load()
'   Screen.MousePointer = vbHourglass
'   ! *****
' * Esta rutina busca los artículos para mostrarlos en el combo Cbo_articulos *

Set dbData = Workspaces(0).OpenDatabase("D:\P_CIVIL\INSUMO.mdb")
  dbsql = "select * from cat_insumos order by cve_insumo"
  Set rsAux = dbData.OpenRecordset(dbsql)
  If rsAux.RecordCount > 0 Then
    Do While Not rsAux.EOF
      Cbo_insumo.AddItem rsAux("nom_insumo") & String(50 -
Len(rsAux("nom_insumo")), ".") & rsAux("cve_insumo")
      rsAux.MoveNext
    Loop
  End If

  dbsql = "select * from cat_personal order by cve_persona"
  Set rsAux = dbData.OpenRecordset(dbsql)
  If rsAux.RecordCount > 0 Then
    Do While Not rsAux.EOF
      recibe.AddItem rsAux("nom_persona") & String(50 -
Len(rsAux("nom_persona")), ".") & rsAux("cve_persona")
      rsAux.MoveNext
    Loop
  End If

  dbsql = "select * from cat_personal order by cve_persona"
  Set rsAux = dbData.OpenRecordset(dbsql)
  If rsAux.RecordCount > 0 Then
    Do While Not rsAux.EOF
      entrego.AddItem rsAux("nom_persona") & String(50 -
Len(rsAux("nom_persona")), ".") & rsAux("cve_persona")
      rsAux.MoveNext
    Loop
  End If

  dbsql = " select * From cat_inv_devoluciones  "
  Set rsAux = dbData.OpenRecordset(dbsql)

  If rsAux.RecordCount > 0 Then
    rsAux.MoveLast

```

```

txt_pedido = rsAux.RecordCount + 1

Else
txt_pedido = 1
End If
If rsAux.RecordCount = 0 Then
txt_pedido = 1
txt_pedido.Refresh
Else
txt_pedido = rsAux.RecordCount + 1
txt_pedido.Refresh
End If
End Sub

```

```

Private Sub txt_cantidad_recibida_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then cmd_aceptar.SetFocus
If KeyAscii = 8 Then Exit Sub
If Chr(KeyAscii) < "0" Or Chr(KeyAscii) > "9" Then
KeyAscii = 0
End If
End Sub

```

```

Private Sub txt_pedido_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then fecha_entrega.SetFocus
If KeyAscii = 8 Then Exit Sub
If Chr(KeyAscii) < "0" Or Chr(KeyAscii) > "9" Then
KeyAscii = 0
If Not IsNumeric(fecha_entrega.Text) Then
MsgBox "La Clave Debe Ser Numerica", 64, At
fecha_entrega = ""
fecha_entrega.SetFocus
Else
End If
End If
End Sub

```

Pantalla de salidas de Instrumental o equipo del almacén:

The screenshot shows a Microsoft Visual Basic form titled "Captura de Instrumental o Servicios Solicitados". The form is designed for data entry and includes the following elements:

- Form Fields:**
  - No de Solicitud:** A text input field.
  - Instrumental o Servicio:** A dropdown menu.
  - Fecha de Salida:** A date input field with the value "99/99/9999".
  - Area Solicitante:** A dropdown menu.
  - Cantidad Entregada:** A text input field.
  - Aceptar:** A button.
- Table:** A table with 6 columns: "Pedido No.", "Fecha de Solicitud", "Instrumental o Servicio", "Cantidad", and "Area Solicitante". The table has 7 rows, with the first row containing the number "1" in the first column.
- Form Fields (Bottom):**
  - Entrego Insumo:** A dropdown menu.
  - Grabar:** A button.
  - Otro:** A button.
  - Salir:** A button.

The form is displayed in a window titled "Proyecto1 - Microsoft Visual Basic [design] - [fmrSalidas (Form)]". The Windows taskbar at the bottom shows the "Inicio" button, several application icons, and the system tray with the time "3:50".



Option Explicit

```
Private Sub cmd_aceptar_Click()  
    vsp_aceptar.Row = iRen + 1  
    vsp_aceptar.Col = 1  
    vsp_aceptar.Text = txt_pedido  
    vsp_aceptar.Col = 2  
    vsp_aceptar.Text = fecha_entrega  
    vsp_aceptar.Col = 3  
    vsp_aceptar.Text = Cbo_insumo  
    vsp_aceptar.Col = 4  
    vsp_aceptar.Text = txt_cantidad  
    vsp_aceptar.Col = 5  
    vsp_aceptar.Text = Val(Mid(Cbo_insumo, 51))  
    iRen = iRen + 1  
    Cbo_insumo = Empty  
    entrego = Empty  
    txt_cantidad = Empty  
    Cbo_insumo.SetFocus  
End Sub
```

```
Private Sub cmd_grabar_Click()
```

```
    If iRen >= 1 Then  
        For iPaso = 1 To iRen  
            vsp_aceptar.Row = iPaso  
            dbsql = "INSERT INTO  
cat_inv_salidas(cve_pedido,cve_insumo,nom_insumo," & _  
"fecha_sale,can_sale,nom_entrega) VALUES("  
                vsp_aceptar.Col = 1  
                dbsql = dbsql & Val(vsp_aceptar.Text) & ","  
                vsp_aceptar.Col = 5  
                dbsql = dbsql & Val(vsp_aceptar.Text) & ","  
                vsp_aceptar.Col = 3  
                dbsql = dbsql & vsp_aceptar.Text & ","  
                vsp_aceptar.Col = 2  
                dbsql = dbsql & Format(vsp_aceptar.Text, "dd/mm/yyyy") & ","  
                vsp_aceptar.Col = 4  
                dbsql = dbsql & Val(vsp_aceptar.Text) & ","  
            dbsql = dbsql & entrego & " )"  
            dbData.Execute dbsql  
            iRen = iRen + 1
```

```

        Next
        cmd_grabar.Enabled = False
        cmd_Otro.SetFocus
    End If
End Sub

Private Sub cmd_Otro_Click()
    dbsql = " select * From cat_inv_salidas "
    Set rsAux = dbData.OpenRecordset(dbsql)
    rsAux.MoveLast
    If rsAux.RecordCount = 0 Then
        txt_pedido = 1
        txt_pedido.Refresh
    Else
        txt_pedido = rsAux.RecordCount + 1
        txt_pedido.Refresh
    End If
    vsp_aceptar.MaxRows = 0
    vsp_aceptar.MaxRows = 100
    fecha_entrega = " / / "
    Cbo_insumo = Empty
    entrega = Empty
    txt_cantidad = Empty
    Cbo_insumo.SetFocus

    cmd_grabar.Enabled = True
    iRen = 0
End Sub

Private Sub cmd_salir_Click()
    Unload Me
End Sub

Private Sub fecha_entrega_KeyPress(KeyAscii As Integer)

    If KeyAscii = 8 Then Exit Sub
    If KeyAscii = 13 Then
        fecha_entrega_LostFocus
    End If
End Sub

```

```
End If
End Sub
```

```
Private Sub fecha_entrega_LostFocus()
On Error GoTo q
```

```
    If fecha_entrega <> limpia Then
        If Format(fecha_entrega, "mm/dd/yyyy") > Format(Now, "mm/dd/yyyy") Or
InStr(1, Format(fecha_entrega, "mm/dd/yyyy"), " ") > 0 Or Not
IsDate(fecha_entrega) Or Mid(fecha_entrega, 4, 2) > 12 Or Mid(fecha_entrega, 7,
4) < 2003 Then
            MsgBox "La Fecha es Incorrecta ", 48, At
            fecha_entrega.SetFocus
            fecha_entrega = limpia
        Else
            txt_cantidad.SetFocus
        End If
    Else
        Else
            cmd_aceptar.SetFocus
        End If
    End Sub
```

```
End If
Exit Sub
```

```
q:
If Err.Number = 13 Then MsgBox " fecha incorrecta", , "Fecha":
fecha_entrega.SetFocus: fecha_entrega = limpia: Exit Sub
End Sub
```

```
Private Sub Form_Load()
```

```
' Screen.MousePointer = vbHourglass
```

```
' *****
```

```
' * Esta rutina busca los articulos para mostrarlos en el combo Cbo_articulos *
```

```
Set dbData = Workspaces(0).OpenDatabase("D:\P_CIVIL_JT\INSUMO.mdb")
dbsql = "select * from cat_insumos order by cve_insumo"
Set rsAux = dbData.OpenRecordset(dbsql)
```

```

If rsAux.RecordCount > 0 Then
  Do While Not rsAux.EOF
    Cbo_insumo.AddItem rsAux("nom_insumo") & String(50 -
Len(rsAux("nom_insumo")), ".") & rsAux("cve_insumo")
    rsAux.MoveNext
  Loop
End If
dbsql = " select * From cat_inv_salidas "
Set rsAux = dbData.OpenRecordset(dbsql)

If rsAux.RecordCount > 0 Then
  rsAux.MoveLast

  txt_pedido = rsAux.RecordCount + 1
Else
  txt_pedido = 1
End If

rsAux.MoveLast
If rsAux.RecordCount = 0 Then
  txt_pedido = 1
  txt_pedido.Refresh
Else
  txt_pedido = rsAux.RecordCount + 1
  txt_pedido.Refresh
End If

dbsql = "select * from cat_personal order by cve_persona"
Set rsAux = dbData.OpenRecordset(dbsql)
If rsAux.RecordCount > 0 Then
  Do While Not rsAux.EOF
    entrego.AddItem rsAux("nom_persona") & String(50 -
Len(rsAux("nom_persona")), ".") & rsAux("cve_persona")
    rsAux.MoveNext
  Loop
End If
End Sub

```

```

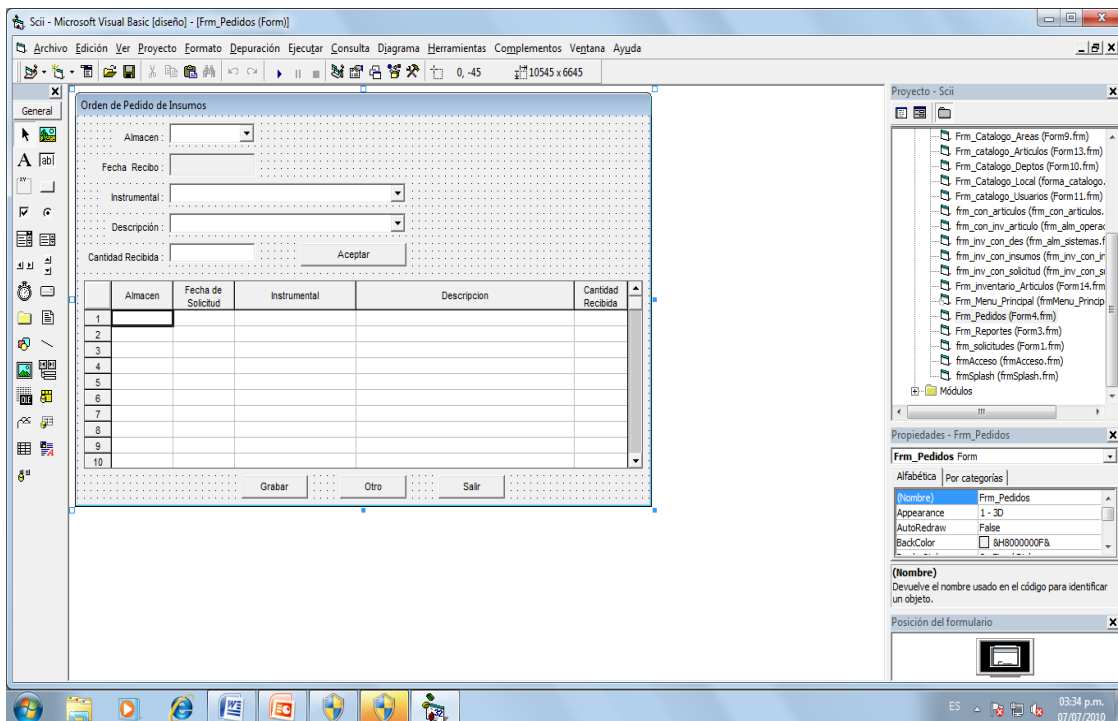
Private Sub txt_cantidad_recibida_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then cmd_aceptar.SetFocus
    If KeyAscii = 8 Then Exit Sub
    If Chr(KeyAscii) < "0" Or Chr(KeyAscii) > "9" Then
        KeyAscii = 0
    End If
End Sub

```

```

Private Sub txt_pedido_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then fecha_entrega.SetFocus
    If KeyAscii = 8 Then Exit Sub
    If Chr(KeyAscii) < "0" Or Chr(KeyAscii) > "9" Then
        KeyAscii = 0
        If Not IsNumeric(fecha_entrega.Text) Then
            MsgBox "La Clave Debe Ser Numerica", 64, At
            fecha_entrega = ""
            fecha_entrega.SetFocus
        End If
    End If
End Sub

```



Codigo de la pantalla:

Option Explicit

```
Private Sub cbo_almacen_Click()  
msd_fecha_sol.SetFocus  
End Sub
```

```
Private Sub cbo_almacen_KeyPress(KeyAscii As Integer)
```

```
    KeyAscii = 0  
    msd_fecha_sol.SetFocus  
End Sub
```

```
Private Sub Cbo_Articulo_KeyPress(KeyAscii As Integer)
```

```
KeyAscii = 0  
    Cbo_descripcion.SetFocus  
End Sub
```

```
Private Sub Cbo_descripcion_Click()
```

```
If Cbo_descripcion <> "" Then  
    For iPaso = 1 To iRen - 1  
vsp_aceptar.Row = iPaso  
        vsp_aceptar.Col = 4  
        If vsp_aceptar.Text = Cbo_descripcion Then  
            MsgBox "El Artículo ya fue seleccionado, Verifique", vbOKOnly +  
vblInformation, "Registro de Entradas"  
            Cbo_descripcion = Empty  
            Cbo_descripcion.SetFocus  
Exit Sub  
        End If  
    Next  
    txt_cantidad_recibida.SetFocus  
End If  
End Sub
```

```
Private Sub cmd_aceptar_Click()
```

```
vsp_aceptar.Row = iRen  
    vsp_aceptar.Col = 1  
    vsp_aceptar.Text = cbo_almacen  
    vsp_aceptar.Col = 2  
    vsp_aceptar.Text = msd_fecha_sol  
    vsp_aceptar.Col = 3  
    vsp_aceptar.Text = Cbo_Articulo  
    vsp_aceptar.Col = 4
```

```

vsp_aceptar.Text = Cbo_descripcion
vsp_aceptar.Col = 5
vsp_aceptar.Text = txt_cantidad_recibida
iRen = iRen + 1

Cbo_Articulo = Empty
Cbo_descripcion = Empty
txt_cantidad_recibida = Empty
Cbo_Articulo.SetFocus
End Sub

Private Sub cmd_Grabar_Click()
    Dim iAI As Integer, iAr As Integer, iDe As Integer, iCa As Double
    If iRen >= 1 Then
        On Error GoTo Error_Grabando
        For iPaso = 1 To iRen - 1
            vsp_aceptar.Row = iPaso
            sSQL = "INSERT INTO
mm_inv_entradas(c_cve_almacen,c_cve_articulo,c_num_articulo," & _
"f_fec_entrada,n_can_entrada) VALUES("
            vsp_aceptar.Col = 1
            sSQL = sSQL & Val(Right(vsp_aceptar.Text, 1)) & ","
            iAI = Val(Right(vsp_aceptar.Text, 1))
            vsp_aceptar.Col = 3
            sSQL = sSQL & Val(Right(vsp_aceptar.Text, 1)) & ","
            iAr = Val(Right(vsp_aceptar.Text, 1))
            vsp_aceptar.Col = 4
            sSQL = sSQL & Val(Right(vsp_aceptar.Text, 1)) & ","
            iDe = Val(Right(vsp_aceptar.Text, 1))
            vsp_aceptar.Col = 2
            sSQL = sSQL & Fecha_Ingles(vsp_aceptar.Text) & ","
            vsp_aceptar.Col = 5
            sSQL = sSQL & Val(vsp_aceptar.Text) & ")"
            iCa = Val(vsp_aceptar.Text)
            wspEspacio.BeginTrans
            dbData.Execute sSQL, dbSQLPassThrough
            wspEspacio.CommitTrans
            sSQL = "SELECT * FROM mm_inv_gral WHERE c_cve_almacen=" & iAI &
" AND c_cve_articulo=" & _
            iAr & " AND c_num_articulo=" & iDe
            Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot)
            If Not recData.BOF And Not recData.EOF Then
                sSQL = "UPDATE mm_inv_gral SET n_cantidad=n_cantidad + " & iCa & _
" WHERE c_cve_almacen=" & iAI & " AND c_cve_articulo=" & _
                iAr & " AND c_num_articulo=" & iDe

```

```

Else
    sSQL = "INSERT INTO
mm_inv_gral(c_cve_almacen,c_cve_articulo,c_num_articulo," & _
        "n_cantidad) VALUES(" & iAI & "," & iAr & "," & iDe & "," & iCa & ")"
End If
wspEspacio.BeginTrans
dbData.Execute sSQL, dbSQLPassThrough
wspEspacio.CommitTrans
Next
cmd_grabar.Enabled = False
End If
Screen.MousePointer = vbdefault
On Error GoTo 0
Exit Sub
Error_Grabando:
If Err.Number <> 0 Then
Call Mensajes_de_Error("Grabando usuarios")
End If
End Sub

```

```

Private Sub Cbo_articulo_Click()
If Cbo_Articulo <> "" Then
    sSQL = "SELECT * FROM mm_inv_articulos WHERE c_cve_articulo=" &
Val(Mid(Cbo_Articulo, 101))
Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot)
Cbo_descripcion.Clear
If Not recData.BOF And Not recData.EOF Then
    Do While Not recData.EOF
        Cbo_descripcion.AddItem recData("d_nom_articulo") & String(100 -
Len(recData("d_nom_articulo")), " ") & recData("c_num_articulo")
        recData.MoveNext
    Loop
End If
Cbo_descripcion.SetFocus
End If
End Sub

```

```

Private Sub cmd_Otro_Click()
    vsp_aceptar.MaxRows = 0

```



```

    vsp_aceptar.MaxRows = 100
msd_fecha_sol = " / / "
    Cbo_Articulo = Empty
    Cbo_descripcion = Empty
    txt_cantidad_recibida = Empty
msd_fecha_sol.SetFocus
    cmd_grabar.Enabled = True
    iRen = 1
End Sub

```

```

Private Sub cmd_salir_Click()
Unload Me
End Sub

```

```

Private Sub Form_Load()
'Screen.MousePointer = vbHourglass
    ! *****
    ' * Esta rutina busca los articulos para mostrarlos en el combo Cbo_articulos *
    ! *****
    sSQL = "SELECT * FROM cat_articulos ORDER BY c_cve_articulo"
    Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
    If Not recData.BOF And Not recData.EOF Then
        Do While Not recData.EOF
            Cbo_Articulo.AddItem recData("d_des_articulo") & String(100 -
Len(recData("d_des_articulo")), " ") & recData("c_cve_articulo")
            recData.MoveNext
        Loop
    End If
    ! *****
    ' * Esta rutina busca los articulos para mostrarlos en el combo cbo_almacen *
    ! *****
    If iAlmacen = 1 Then
        dbsql = "Select * From cat_almacen Order By c_cve_almacen"
        Set rsData = dbData.OpenRecordset(dbsql)
        If rsData.RecordCount > 0 Then
            Do While Not rsData.EOF
cbo_almacen.AddItem rsData("d_des_almacen") & String(30 -
Len(rsData("d_des_almacen")), ",") & rsData("c_cve_almacen")

```

```

rsData.MoveNext
    Loop
End If
Else
cbo_almacen = Busca_Almacen(iAlmacen)
cbo_almacen.Enabled = False
End If
iRen = 1
End Sub

Private Sub msd_fecha_sol_KeyPress(KeyAscii As Integer)
    If KeyAscii = 8 Then Exit Sub
    If KeyAscii = 13 Then
        msd_fecha_sol_LostFocus
    End If
End Sub

Private Sub msd_fecha_sol_LostFocus()
    If msd_fecha_sol <> " / / " Then
        If Format(msd_fecha_sol, "mm/dd/yyyy") > Format(Now, "mm/dd/yyyy") Or
        InStr(1, Format(msd_fecha_sol, "mm/dd/yyyy"), " ") > 0 Or Not
        IsDate(msd_fecha_sol) Or Mid(msd_fecha_sol, 4, 2) > 12 Or Mid(msd_fecha_sol,
        7, 4) < 2001 Then
            MsgBox "La Fecha es Incorrecta ", 48, At, "El Artículo ya fue seleccionado,
            Verifique", vbOKOnly + vbInformation, "Registro de Entradas"
            MsgBox "La Fecha es Incorrecta, Por favor Verifiquela", vbOKOnly +
            vbInformation, "Registro de Entradas"
            msd_fecha_sol.SetFocus
            msd_fecha_sol = limpia
        Else
            Cbo_Articulo.SetFocus
        End If
    End If
End Sub

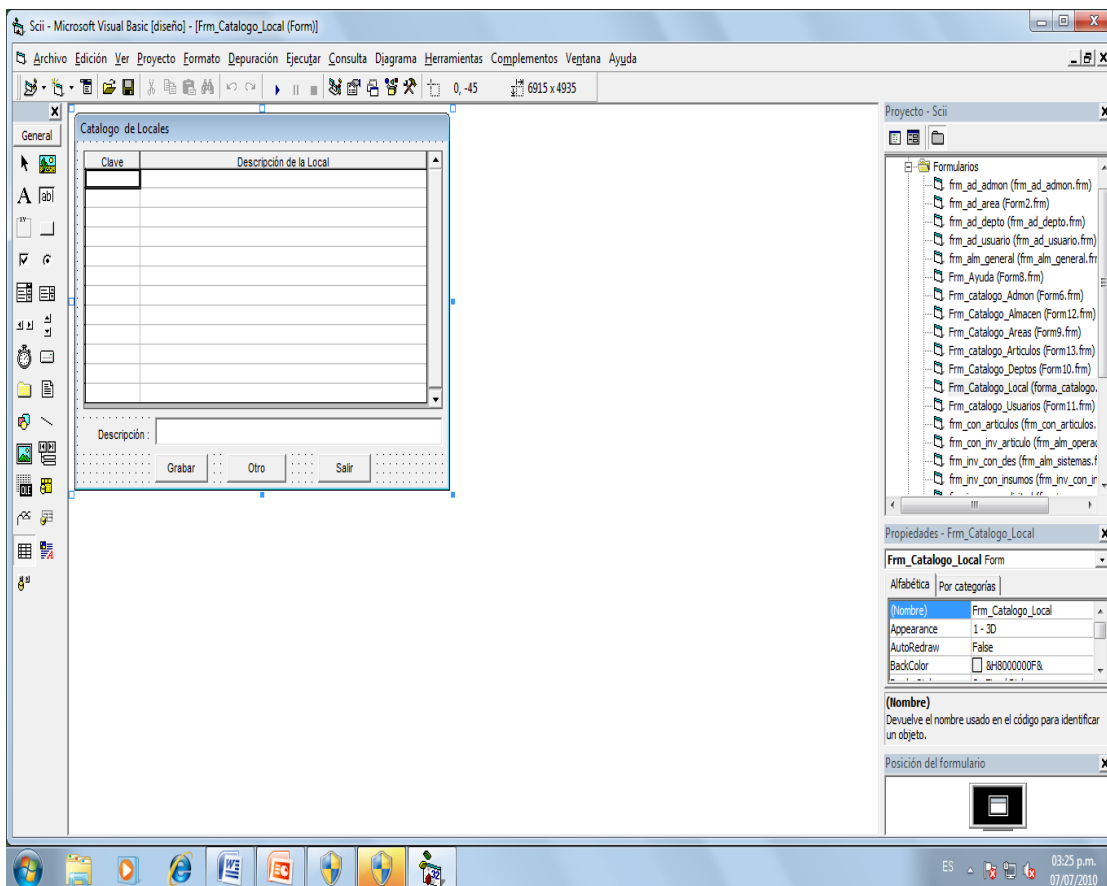
Private Sub txt_cantidad_recibida_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then cmd_aceptar.SetFocus
    If KeyAscii = 8 Then Exit Sub
    If Chr(KeyAscii) < "0" Or Chr(KeyAscii) > "9" Then
        KeyAscii = 0
    End If
End Sub

```

## FASES DE DESARROLLO CARACTERISTICAS DEL SISTEMA

- PANTALLAS DE CATALOGOS
- Y CODIGO DE APLICACION

Esta pantalla muestra las opciones de ingreso y de nuevas locales, asi como el codigo a utilizar:



```

Option Explicit
Private Sub cmdSalir_Click()
    Unload Me
End Sub
Private Sub llenar_catalogo()
    On Error GoTo Error_Llena_Catalogo
    sSQL = "SELECT * FROM cat_local ORDER BY c_cve_local"
    Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
    If Not recData.BOF And Not recData.EOF Then
        recData.MoveLast

        Listado.MaxRows = recData.RecordCount
        recData.MoveFirst
        iRen = 1
        Do While Not recData.EOF
Listado.Row = iRen
            Listado.Col = 1
            Listado.Text = recData("c_cve_local")
            Listado.Col = 2
            Listado.Text = recData("d_nom_Local")
recData.MoveNext
            iRen = iRen + 1
        Loop
    End If
    Screen.MousePointer = vbdefault
    On Error GoTo 0
Exit Sub
Error_Llena_Catalogo:
If Err.Number <> 0 Then
Call Mensajes_de_Error("Leyendo Locales")
End If
End Sub
Private Sub cmd_Grabar_Click()
    If txt_Descripcion <> "" Then
        On Error GoTo Error_Grabando
        sSQL = "SELECT * FROM cat_local WHERE c_cve_local IS NOT NULL"
        Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
        If Not recData.BOF And Not recData.EOF Then

```

```

recData.MoveLast
    txt_cve_Local = recData.RecordCount + 1
    txt_cve_Local.Refresh
Else
    txt_cve_Local = 1
    txt_cve_Local.Refresh
End If
wspEspacio.BeginTrans
sSQL = "INSERT INTO cat_local(c_cve_local,d_nom_local) VALUES(" &
txt_cve_Local & "," & txt_Descripcion & ")"
dbData.Execute sSQL, dbSQLPassThrough
sSQL = "INSERT INTO
cat_admon(c_cve_local,c_cve_admon,d_nom_admon) VALUES(" & txt_cve_Local
& ",1,'TITULAR ')"
dbData.Execute sSQL, dbSQLPassThrough
sSQL = "INSERT INTO
cat_area(c_cve_local,c_cve_admon,c_cve_area,d_nom_area) VALUES(" &
txt_cve_Local & ",1,1,'NULO'"
dbData.Execute sSQL, dbSQLPassThrough
sSQL = "INSERT INTO
cat_depto(c_cve_local,c_cve_admon,c_cve_area,c_cve_depto,d_nom_depto)
VALUES(" & txt_cve_Local & ",1,1,1,'NULO'"
dbData.Execute sSQL, dbSQLPassThrough
wspEspacio.CommitTrans

Call llena_catalogo
    cmd_grabar.Enabled = False
End If
    Screen.MousePointer = vbdefault
    On Error GoTo 0
    Exit Sub
Error_Grabando:
    If Err.Number <> 0 Then
Call Mensajes_de_Error("Grabando Locales")
End If
End Sub

Private Sub cmd_nuevalocal_Click()
    txt_cve_Local = Empty
    txt_Descripcion = Empty
    txt_Descripcion.SetFocus
    cmd_grabar.Enabled = True
End Sub

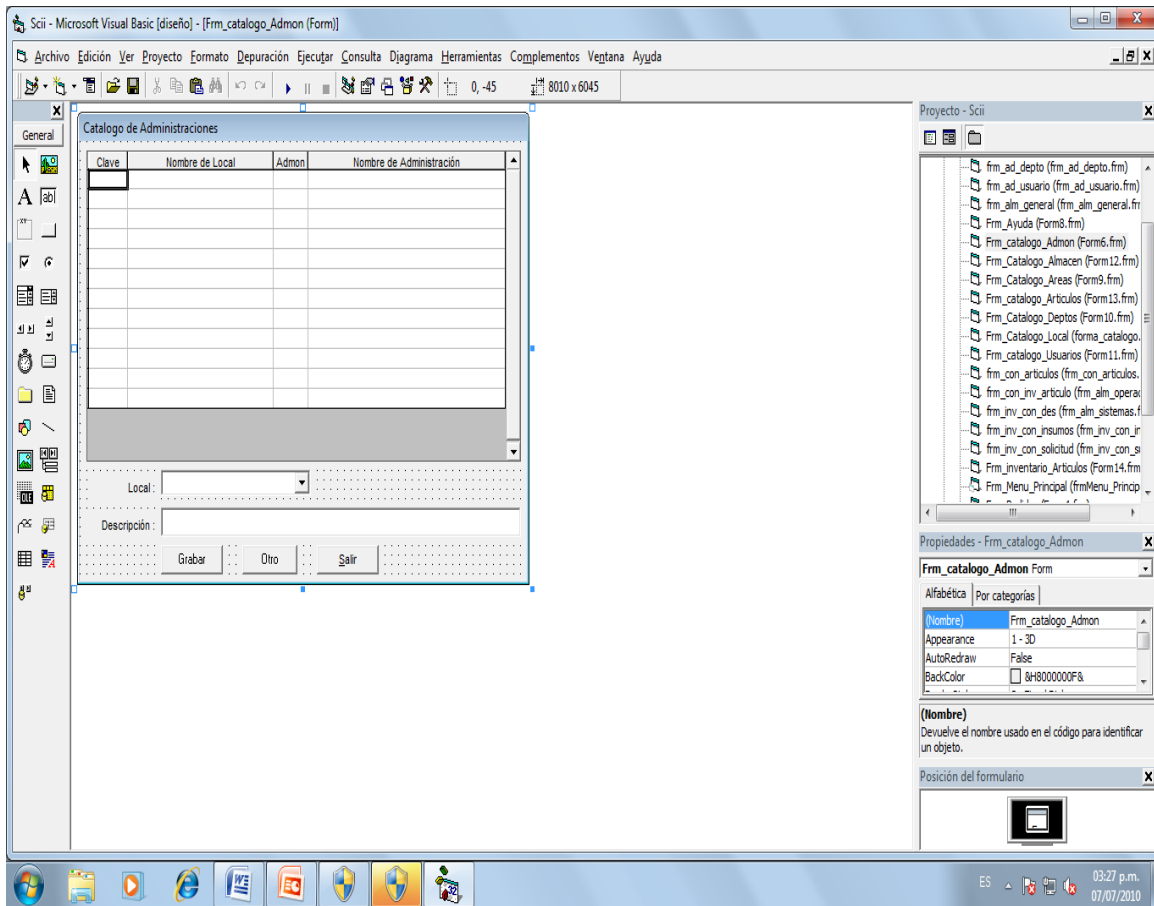
```

```
Private Sub cmd_salir_Click(Index As Integer)
    Unload Me
End Sub
```

```
Private Sub Form_Load()
    Screen.MousePointer = vbHourglass
    Call llenar_catalogo
    Screen.MousePointer = vbdefault
End Sub
```

```
Private Sub txt_Descripcion_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then cmd_grabar.SetFocus
    If KeyAscii = 8 Then Exit Sub
    Select Case Chr(KeyAscii)
        Case "{", "}", "[", "]", "|", "`", "¿", "?", "¡", "!", "=", "/", "\", ""
            KeyAscii = 0
        Case Else
            KeyAscii = Asc(UCase(Chr(KeyAscii)))
    End Select
End Sub
```

## Catalogo de administraciones:



Codigo:

```
Option Explicit
Private Sub cmdSalir_Click()
    Unload Me
End Sub
Private Sub llena_catalogo()
```

## On Error GoTo Error\_Llena\_Catalogo

```
' *****
' * Esta rutina busca las locales y sus administracion, para mostrarlas en el *
' * spread *
' *****

sSQL = "SELECT
cat_local.c_cve_local,cat_local.d_nom_local,cat_admon.c_cve_admon," & _
      "cat_admon.d_nom_admon FROM cat_local INNER JOIN cat_admon ON "
& _
      "cat_local.c_cve_local = cat_admon.c_cve_local WHERE
cat_admon.c_cve_admon<>999"
Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
If Not recData.EOF And Not recData.BOF Then
    recData.MoveLast
    Listado.MaxRows = recData.RecordCount

    recData.MoveFirst
    iRen = 1
    Do While Not recData.EOF
Listado.Row = iRen
        Listado.Col = 1
        Listado.Text = recData("c_cve_local")
        Listado.Col = 2
        Listado.Text = recData("d_nom_local")
        Listado.Col = 3
        Listado.Text = recData("c_cve_admon")
Listado.Col = 4
        Listado.Text = recData("d_nom_admon")
        recData.MoveNext
        iRen = iRen + 1
    Loop
End If
Screen.MousePointer = vbdefault
On Error GoTo 0
Exit Sub
Error_Llena_Catalogo:
    If Err.Number <> 0 Then
        Call Mensajes_de_Error("Leyendo Admonistraciones")
    End If
End Sub
```



```

Private Sub Cbo_locales_Click()
    If Cbo_locales <> "" Then
        txt_Descripcion.SetFocus
    End If
End Sub

Private Sub Cbo_locales_KeyPress(KeyAscii As Integer)
    KeyAscii = 0
    txt_Descripcion.SetFocus
End Sub

Private Sub cmd_grabar_Click()
    If txt_Descripcion <> "" And Cbo_locales <> "" Then
        On Error GoTo Error_Grabando
        sSQL = "SELECT * FROM cat_admon WHERE c_cve_local=" &
Val(Mid(Cbo_locales, 101))
        Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
        If Not recData.BOF And Not recData.EOF Then
            recData.MoveLast
            txt_cve_Admon = recData.RecordCount + 1
        End If
        wspEspacio.BeginTrans
        sSQL = "INSERT INTO
cat_admon(c_cve_local,c_cve_admon,d_nom_admon) VALUES(" & _
Val(Mid(Cbo_locales, 101)) & "," & txt_cve_Admon & "," & _
txt_Descripcion & ")"
        dbData.Execute sSQL, dbSQLPassThrough
        sSQL = "INSERT INTO
cat_area(c_cve_local,c_cve_admon,c_cve_area,d_nom_area) " & _
"VALUES(" & Val(Mid(Cbo_locales, 101)) & "," & txt_cve_Admon & _
",1,'TITULAR'"
        dbData.Execute sSQL, dbSQLPassThrough
        sSQL = "INSERT INTO
cat_depto(c_cve_local,c_cve_admon,c_cve_area,c_cve_depto,d_nom_depto) " &
_
        "VALUES(" & Val(Mid(Cbo_locales, 101)) & "," & txt_cve_Admon & _
        ",1,1,'NULO'"
        dbData.Execute sSQL, dbSQLPassThrough
        wspEspacio.CommitTrans
    Call llena_catalogo
        cmd_grabar.Enabled = False
    End If
    Screen.MousePointer = vbdefault

```

```

On Error GoTo 0
    Exit Sub
Error_Grabando:
If Err.Number <> 0 Then
Call Mensajes_de_Error("Grabando admon")
End If
End Sub

Private Sub cmd_nuevaAdmon_Click()
    txt_cve_Admon = Empty
    txt_Descripcion = Empty
txt_Descripcion.SetFocus
    Cbo_locales = Empty
    cmd_grabar.Enabled = True
    Cbo_locales.SetFocus
End Sub

Private Sub cmd_salir_Click()
    Unload Me
End Sub

Private Sub Form_Load()
    Screen.MousePointer = vbHourglass
    Call llena_catalogo
! *****
! * Esta rutina busca las locales para mostrarlas en el combo Cbo_locales *
! *****

    sSQL = "SELECT * FROM cat_local ORDER BY c_cve_local"
    Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
    If Not recData.BOF And Not recData.EOF Then
        Do While Not recData.EOF

            Cbo_locales.AddItem recData("d_nom_local") & String(100 -
Len(recData("d_nom_local")), " ") & recData("c_cve_local")
            recData.MoveNext
        Loop
    End If
    Screen.MousePointer = vbdefault
End Sub

Private Sub txt_Descripcion_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then cmd_grabar.SetFocus
    If KeyAscii = 8 Then Exit Sub
    Select Case Chr(KeyAscii)

```

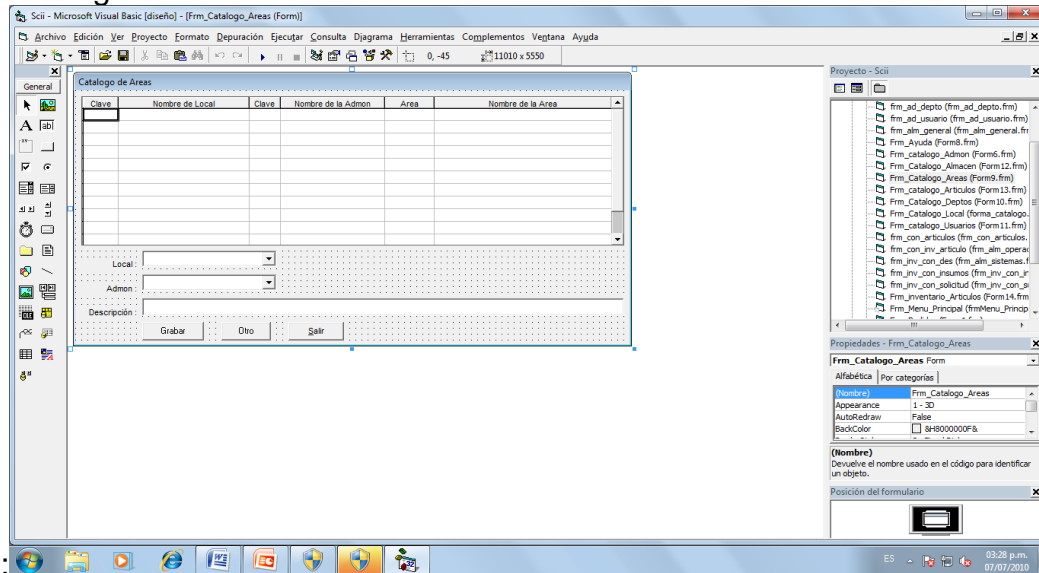
```

Case "{", "}", "[", "]", "|", "`", "¿", "?", "¡", "!", "=", "/", "\", ""
    KeyAscii = 0
Case Else
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Select
End Sub

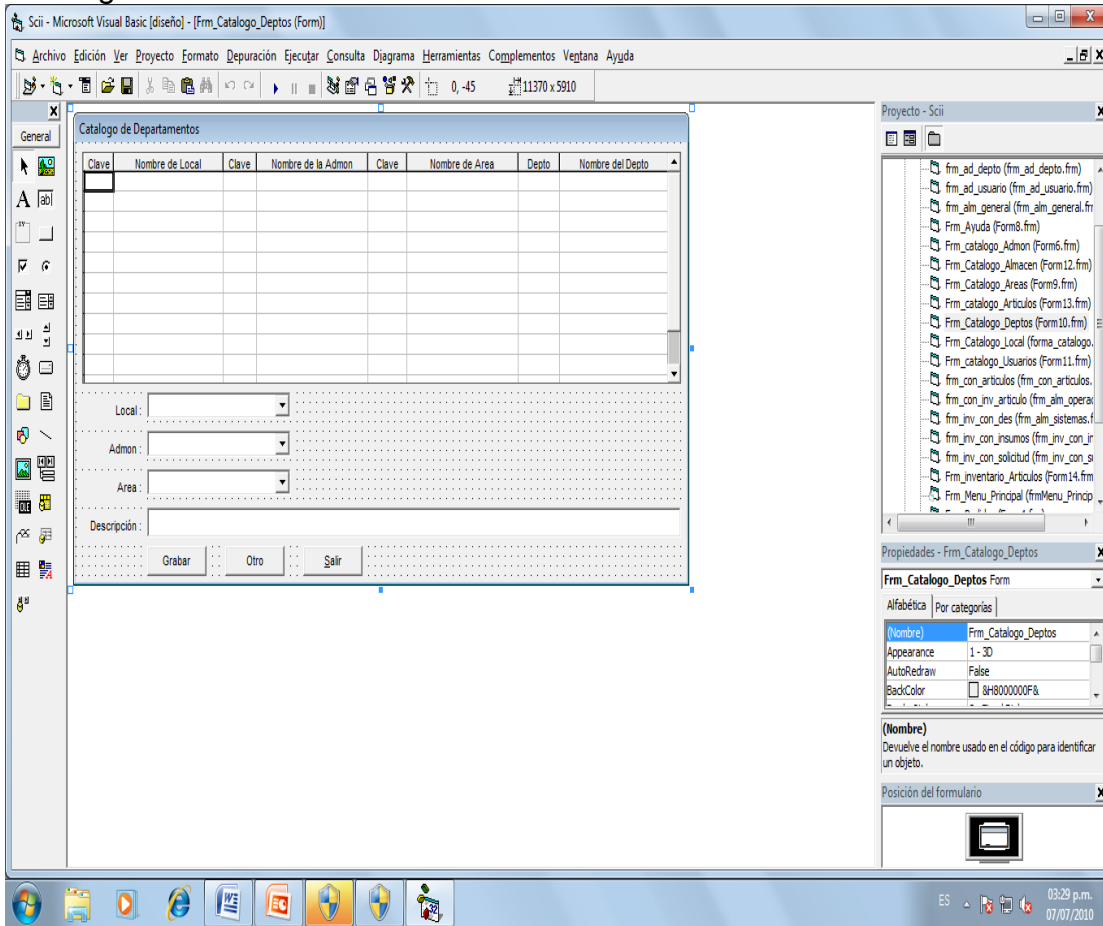
Private Sub cmd_salir_Click()
Unload Me
End Sub

```

## Catalogo de áreas



## Catalogo de áreas



### Option Explicit

```
Private Sub cmdSalir_Click()
```

```
    Unload Me
```

```
End Sub
```

```
Private Sub llena_catalogo()
```

```
    On Error GoTo Error_Llena_Catalogo
```

```
107
```

```
    ' * Esta rutina busca las locales, administracion y sus areas, *
```

```
    ' * para mostrarlas en el spread *
```

```
! *****
```

```
    sSQL = "SELECT
```

```
cat_local.c_cve_local,cat_local.d_nom_local,cat_admon.c_cve_admon," & _
```

```
    "cat_admon.d_nom_admon, cat_area.c_cve_area, cat_area.d_nom_area "
```

```
& _
```

```
    "FROM (cat_local INNER JOIN cat_admon ON cat_local.c_cve_local = " &
```

```
    "cat_admon.c_cve_local) INNER JOIN cat_area ON
```

```
(cat_admon.c_cve_admon = " & _
```

```

        "cat_area.c_cve_admon) AND (cat_local.c_cve_local =
cat_area.c_cve_local) " & _
        "WHERE c_cve_area<>999 " & _
        "ORDER BY cat_local.c_cve_local, cat_admon.c_cve_admon,
cat_area.c_cve_area"
    Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
    If Not recData.EOF And Not recData.EOF Then
        recData.MoveLast
        Listado.MaxRows = recData.RecordCount
        recData.MoveFirst
        iRen = 1
        Do While Not recData.EOF
Listado.Row = iRen
            Listado.Col = 1
            Listado.Text = recData("c_cve_local")
            Listado.Col = 2
            Listado.Text = recData("d_nom_local")
            Listado.Col = 3
            Listado.Text = recData("c_cve_admon")
            Listado.Col = 4
            Listado.Text = recData("d_nom_admon")
            Listado.Col = 5
Listado.Text = recData("c_cve_area")
            Listado.Col = 6
            Listado.Text = recData("d_nom_area")
            recData.MoveNext
            iRen = iRen + 1
        Loop
    End If
    Screen.MousePointer = vbdefault
On Error GoTo 0
    Exit Sub
Error_Llena_Catalogo:
If Err.Number <> 0 Then
    Call Mensajes_de_Error("Leyendo Areas")
    End If
End Sub

Private Sub Cbo_admon_Click()
    If Cbo_Admon <> "" Then
        txt_Descripcion.SetFocus
    End If
End Sub

Private Sub Cbo_Admon_KeyPress(KeyAscii As Integer)
KeyAscii = 0

```

```
txt_Descripcion.SetFocus
End Sub
```

```
Private Sub Cbo_locales_Click()
    If Cbo_locales <> "" Then
        sSQL = "SELECT * FROM cat_admon WHERE c_cve_local=" &
Val(Mid(Cbo_locales, 101))
        Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
        Cbo_Admon.Clear
        If Not recData.BOF And Not recData.EOF Then
            Do While Not recData.EOF
                Cbo_Admon.AddItem recData("d_nom_admon") & String(100 -
Len(recData("d_nom_admon")), " ") & recData("c_cve_admon")
                recData.MoveNext
            Loop
        End If

        Cbo_Admon.SetFocus
    End If
End Sub
```

```
Private Sub Cbo_locales_KeyPress(KeyAscii As Integer)
    KeyAscii = 0
    Cbo_Admon.SetFocus
End Sub
```

```
Private Sub cmd_Grabar_Click()
    If txt_Descripcion <> "" And Cbo_Admon <> "" And Cbo_locales <> "" Then
        On Error GoTo Error_Grabando
        sSQL = "SELECT * FROM cat_area WHERE c_cve_local=" &
Val(Mid(Cbo_locales, 101)) & _
        " AND c_cve_admon=" & Val(Mid(Cbo_Admon, 101))
        Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
        If Not recData.BOF And Not recData.EOF Then
            recData.MoveLast
            txt_cve_Area = recData.RecordCount + 1
        End If
        wspEspacio.BeginTrans
        sSQL = "INSERT INTO
cat_area(c_cve_local,c_cve_admon,c_cve_area,d_nom_area) VALUES(" & _
        Val(Mid(Cbo_locales, 101)) & "," & Val(Mid(Cbo_Admon, 101)) & "," &
txt_cve_Area & _
        "," & txt_Descripcion & ")"
        dbData.Execute sSQL, dbSQLPassThrough
```

```

        sSQL = "INSERT INTO
cat_depto(c_cve_local,c_cve_admon,c_cve_area,c_cve_depto," & _
        "d_nom_depto) VALUES(" & Val(Mid(Cbo_locales, 101)) & "," & _
        Val(Mid(Cbo_Admon, 101)) & "," & txt_cve_Area & ",1,'TITULAR')"
        dbData.Execute sSQL, dbSQLPassThrough
        wspEspacio.CommitTrans
    Call llena_catalogo
        cmd_grabar.Enabled = False
End If
    Screen.MousePointer = vbdefault
    On Error GoTo 0
    Exit Sub
Error_Grabando:
    If Err.Number <> 0 Then

Call Mensajes_de_Error("Grabando Area")
End If
End Sub

Private Sub cmd_nuevaArea_Click()
txt_Descripcion = Empty
    txt_Descripcion.SetFocus
Cbo_locales = Empty
    Cbo_Admon = Empty
    cmd_grabar.Enabled = True
    Cbo_locales.SetFocus
End Sub

Private Sub cmd_salir_Click()
    Unload Me
End Sub

```

```

Private Sub Form_Load()
    Screen.MousePointer = vbHourglass
    Call llena_catalogo
    ' *****
    ' * Esta rutina busca las locales para mostrarlas en el combo Cbo_locales *
    ' *****
    sSQL = "SELECT * FROM cat_local ORDER BY c_cve_local"
    Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
    If Not recData.EOF And Not recData.BOF Then
        Do While Not recData.EOF
            Cbo_locales.AddItem recData("d_nom_local") & String(100 -
Len(recData("d_nom_local")), " ") & recData("c_cve_local")
            recData.MoveNext
        Loop
    End If
    Screen.MousePointer = vbdefault
End Sub

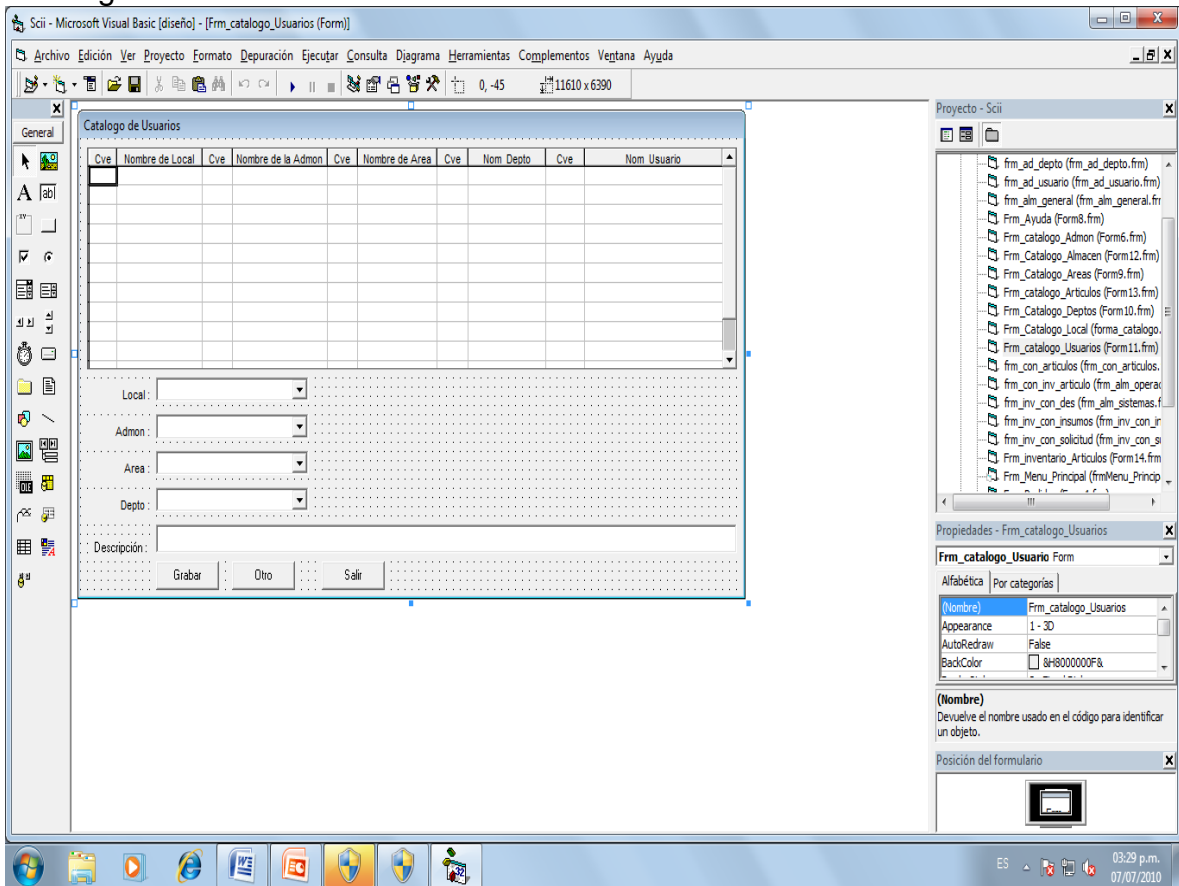
Private Sub txt_Descripcion_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then cmd_grabar.SetFocus
    If KeyAscii = 8 Then Exit Sub
    Select Case Chr(KeyAscii)

        Case "{", "}", "[", "]", "|", "\", ":", "?", ";", "!", "=", "/", "\, ""
            KeyAscii = 0
        Case Else
            KeyAscii = Asc(UCase(Chr(KeyAscii)))
    End Select
End Sub

```



## Catalogo de usuarios:



### Option Explicit

```
Private Sub Cbo_Admon_KeyPress(KeyAscii As Integer)
```

```
KeyAscii = 0
```

```
Cbo_Area.SetFocus
```

```
End Sub
```

```
Private Sub Cbo_Area_KeyPress(KeyAscii As Integer)
```

```
KeyAscii = 0
```

```
Cbo_Depto.SetFocus
```

```
End Sub
```

```
Private Sub Cbo_Depto_KeyPress(KeyAscii As Integer)
```

```
KeyAscii = 0
```

```
txt_descripcion.SetFocus
```

```
End Sub
```

```

Private Sub Cbo_locales_KeyPress(KeyAscii As Integer)
KeyAscii = 0
Cbo_Admon.SetFocus
End Sub

Private Sub cmd_nuevousuario_Click()
txt_descripcion = Empty
txt_descripcion.SetFocus
Cbo_locales = Empty
Cbo_Admon = Empty
Cbo_Area = Empty
Cbo_Depto = Empty

cmd_Grabar.Enabled = True
Cbo_locales.SetFocus
End Sub

Private Sub cmd_Salir_Click(Index As Integer)
Unload Me

End Sub

Private Sub cbo_area_Click()
If Cbo_locales <> "" And Cbo_Area <> "" Then
sSQL = "SELECT * FROM cat_depto WHERE c_cve_local=" &
Val(Mid(Cbo_locales, 101)) & _
" AND c_cve_admon=" & Val(Mid(Cbo_Admon, 101)) & " AND
c_cve_area=" & Val(Mid(Cbo_Area, 101))
Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
Cbo_Depto.Clear
If Not recData.BOF And Not recData.EOF Then
Do While Not recData.EOF
Cbo_Depto.AddItem recData("d_nom_depto") & String(100 -
Len(recData("d_nom_depto")), " ") & recData("c_cve_depto")
recData.MoveNext
Loop
End If
Cbo_Depto.SetFocus
End If
End Sub

Private Sub Cbo_depto_Click()
If Cbo_Depto <> "" Then
txt_descripcion.SetFocus
End If
End Sub

Private Sub cmd_Grabar_Click()

```

```

If Val(Mid(Cbo_locales, 101)) <= 0 Then MsgBox "Debe seleccionar una Local",
vbOKOnly + vbInformation, "Alta de Usuarios": Cbo_locales.SetFocus: Exit Sub
    If Val(Mid(Cbo_Admon, 101)) <= 0 Then MsgBox "Debe seleccionar una
Administración", vbOKOnly + vbInformation, "Alta de Usuarios":
Cbo_Admon.SetFocus: Exit Sub
        If Val(Mid(Cbo_Area, 101)) <= 0 Then MsgBox "Debe seleccionar el Area",
vbOKOnly + vbInformation, "Alta de Usuarios": Cbo_Area.SetFocus: Exit Sub
            If Val(Mid(Cbo_Depto, 101)) <= 0 Then MsgBox "Debe seleccionar un
Departamento", vbOKOnly + vbInformation, "Alta de Usuarios":
Cbo_Depto.SetFocus: Exit Sub
                If txt_descripcion = "" Then MsgBox "Debe indicar el Nombre del Usuario",
vbOKOnly + vbInformation, "Alta de Usuarios": txt_descripcion.SetFocus: Exit Sub
                    On Error GoTo Error_Grabando
                    sSQL = "SELECT * FROM cat_usuarios"
                    Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
                    If Not recData.BOF And Not recData.EOF Then
                        recData.MoveLast

                        txt_cve_Usuario = recData.RecordCount + 1
                    Else
                        txt_cve_Usuario = 1
                    End If
                    wspEspacio.BeginTrans
                    sSQL = "INSERT INTO
cat_usuarios(c_cve_local,c_cve_admon,c_cve_area,c_cve_depto,c_cve_usuario,d
_nom_usuario) VALUES(" & _
                        Val(Mid(Cbo_locales, 101)) & "," & Val(Mid(Cbo_Admon, 101)) & "," &
Val(Mid(Cbo_Area, 101)) & "," & Val(Mid(Cbo_Depto, 101)) & "," & txt_cve_Usuario
& _
                        "," & txt_descripcion & ")"
                    dbData.Execute sSQL, dbSQLPassThrough
                    wspEspacio.CommitTrans
                    Call llena_catalogo
                    cmd_Grabar.Enabled = False
                    Screen.MousePointer = vbdefault
                    On Error GoTo 0
                    Exit Sub
                Error_Grabando:
                    If Err.Number <> 0 Then
                        Call Mensajes_de_Error("Grabando usuarios")
                    End If
            End Sub

Private Sub cmd_nuevaArea_Click()
txt_descripcion = Empty
    txt_descripcion.SetFocus

```

```

Cbo_locales = Empty
  Cbo_Admon = Empty
  cmd_Grabar.Enabled = True
  Cbo_locales.SetFocus
End Sub

'Private Sub cmd_salir_Click()
'  Unload Me
'End Sub
Private Sub cmd_nuevodepto_Click()

txt_descripcion = Empty
  txt_descripcion.SetFocus
Cbo_locales = Empty
  Cbo_Admon = Empty
  Cbo_Area = Empty
  Cbo_Depto = Empty
  cmd_Grabar.Enabled = True
Cbo_locales.SetFocus
End Sub
Private Sub llena_catalogo()
  On Error GoTo Error_Llena_Catalogo
  ! *****
  ! * Esta rutina busca las locales, administracion y sus areas, *
  ! * para mostrarlas en el spread *
  ! *****
  sSQL = "SELECT cat_local.c_cve_local, cat_local.d_nom_local,
cat_admon.c_cve_admon, " & _
  "cat_admon.d_nom_admon, cat_area.c_cve_area, cat_area.d_nom_area,
cat_depto.c_cve_depto, " & _
"cat_depto.d_nom_depto, cat_usuarios.c_cve_usuario,
cat_usuarios.d_nom_usuario " & _
"FROM (((CAT_LOCAL INNER JOIN cat_admon on cat_local.c_cve_local =
cat_admon.c_cve_local) " & _
  "INNER JOIN cat_area ON (cat_area.c_cve_admon =
cat_admon.c_cve_admon) AND (cat_local.c_cve_local " & _
  "= cat_area.c_cve_local)) INNER JOIN cat_depto ON (cat_depto.c_cve_area =
cat_area.c_cve_area) AND " & _
  "(cat_depto.c_cve_admon = cat_area.c_cve_admon) AND
(cat_local.c_cve_local = cat_depto.c_cve_local)) " & _
  "INNER JOIN cat_usuarios ON (cat_usuarios.c_cve_depto =
cat_depto.c_cve_depto) AND (cat_usuarios.c_cve_area " & _
  "= cat_depto.c_cve_area) AND (cat_usuarios.c_cve_admon =
cat_depto.c_cve_admon) AND (cat_local.c_cve_local " & _
  "= cat_usuarios.c_cve_local); "
  Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)

```

```

If Not recData.EOF And Not recData.EOF Then
    recData.MoveLast
    Listado.MaxRows = recData.RecordCount
    recData.MoveFirst
    iRen = 1
    Do While Not recData.EOF
Listado.Row = iRen
        Listado.Col = 1
        Listado.Text = recData("c_cve_local")
        Listado.Col = 2
        Listado.Text = recData("d_nom_local")
        Listado.Col = 3
        Listado.Text = recData("c_cve_admon")
        Listado.Col = 4
        Listado.Text = recData("d_nom_admon")
        Listado.Col = 5
Listado.Text = recData("c_cve_area")
Listado.Col = 6
        Listado.Text = recData("d_nom_area")
        Listado.Col = 7
        Listado.Text = recData("c_cve_depto")
        Listado.Col = 8
        Listado.Text = recData("d_nom_depto")
        Listado.Col = 9
        Listado.Text = recData("c_cve_usuario")
        Listado.Col = 10
        Listado.Text = recData("d_nom_usuario")
recData.MoveNext
        iRen = iRen + 1
    Loop
End If
Screen.MousePointer = vbdefault
On Error GoTo 0
Exit Sub
Error_Llena_Catalogo:
If Err.Number <> 0 Then
Call Mensajes_de_Error("Leyendo usuarios")
End If
End Sub

Private Sub Cbo_admon_Click()
    If Cbo_locales <> "" And Cbo_Admon <> "" Then
        sSQL = "SELECT * FROM cat_area WHERE c_cve_local=" &
Val(Mid(Cbo_locales, 101)) & _
            " AND c_cve_admon=" & Val(Mid(Cbo_Admon, 101))
        Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
    End If
End Sub

```

```

Cbo_Area.Clear
If Not recData.BOF And Not recData.EOF Then
    Do While Not recData.EOF
        Cbo_Area.AddItem recData("d_nom_area") & String(100 -
Len(recData("d_nom_area")), " ") & recData("c_cve_area")
        recData.MoveNext
    Loop
End If
Cbo_Area.SetFocus
End If
End Sub

```

```

Private Sub Cbo_locales_Click()
    If Cbo_locales <> "" Then
        sSQL = "SELECT * FROM cat_admon WHERE c_cve_local=" &
Val(Mid(Cbo_locales, 101))
        Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
        Cbo_Admon.Clear
        If Not recData.BOF And Not recData.EOF Then
            Do While Not recData.EOF
                Cbo_Admon.AddItem recData("d_nom_admon") & String(100 -
Len(recData("d_nom_admon")), " ") & recData("c_cve_admon")
                recData.MoveNext
            Loop
        End If
        Cbo_Admon.SetFocus
    End If
End Sub

```

```

Private Sub Form_Load()
    Screen.MousePointer = vbHourglass
    Call llenar_catalogo
    ! *****
    ! * Esta rutina busca las locales para mostrarlas en el combo Cbo_locales *
    ! *****
    sSQL = "SELECT * FROM cat_local ORDER BY c_cve_local"
    Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
    If Not recData.BOF And Not recData.EOF Then
        Do While Not recData.EOF
            Cbo_locales.AddItem recData("d_nom_local") & String(100 -
Len(recData("d_nom_local")), " ") & recData("c_cve_local")
            recData.MoveNext
        Loop
    End If
    Screen.MousePointer = vbdefault

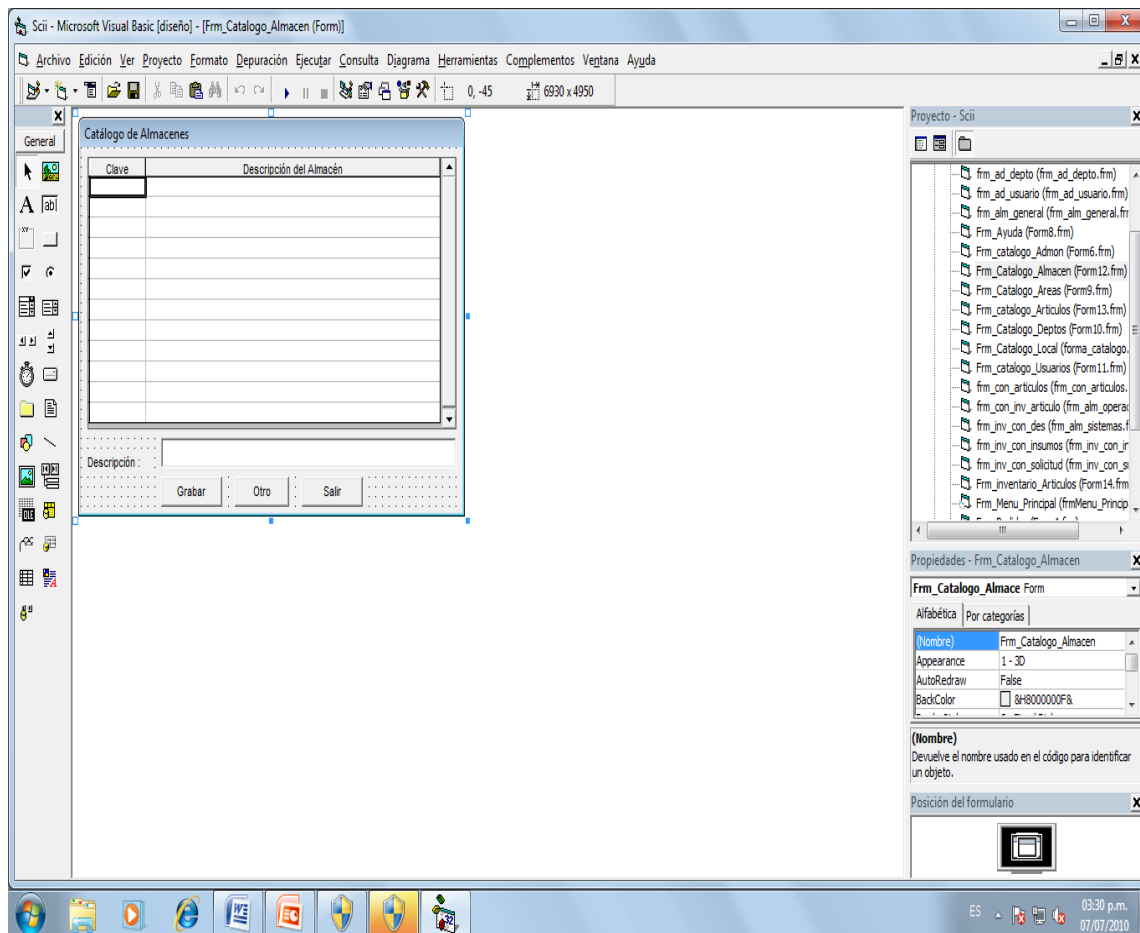
```

End Sub

```
Private Sub txt_Descripcion_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then cmd_Grabar.SetFocus
    If KeyAscii = 8 Then Exit Sub
    Select Case Chr(KeyAscii)
        Case "{", "}", "[", "]", "|", "`", "¿", "?", "¡", "!", "=", "/", "\", ""
            KeyAscii = 0
        Case Else
            KeyAscii = Asc(UCase(Chr(KeyAscii)))
    End Select
End Sub
```

End Sub

Catalogo de almacenes:



Option Explicit

```
Private Sub llena_catalogo()
```

```
Screen.MousePointer = vbHourglass
```

```
    sSQL = "SELECT * FROM cat_almacen ORDER BY c_cve_almacen"
```

```
    Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
```

```
dbSQLPassThrough)
```

```

If Not recData.BOF And Not recData.EOF Then
    recData.MoveLast
    Listado.MaxRows = recData.RecordCount
    recData.MoveFirst
    iRen = 1
    Do While Not recData.EOF
Listado.Row = iRen
        Listado.Col = 1
        Listado.Text = recData("c_cve_almacen")
        Listado.Col = 2
        Listado.Text = recData("d_des_almacen")
recData.MoveNext
        iRen = iRen + 1
    Loop
End If
Screen.MousePointer = vbdefault
End Sub

Private Sub cmd_Grabar_Click()
    If txt_Descripcion <> "" Then
        sSQL = "SELECT * FROM cat_almacen WHERE c_cve_almacen IS NOT
NULL"
        Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
        If Not recData.BOF And Not recData.EOF Then
            recData.MoveLast
            txt_cve_Almacen = recData.RecordCount
            txt_cve_Almacen.Refresh
        Else
            txt_cve_Almacen = 1
            txt_cve_Almacen.Refresh
        End If
        wspEspacio.BeginTrans

        sSQL = "INSERT INTO cat_almacen(c_cve_almacen,d_des_almacen)
VALUES(" & txt_cve_Almacen & "," & txt_Descripcion & ")"
        dbData.Execute sSQL, dbSQLPassThrough
        wspEspacio.CommitTrans
    Call llena_catalogo
        cmd_grabar.Enabled = False
    End If
End Sub

Private Sub cmd_nuevoalmacen_Click()
txt_cve_Almacen = Empty
    txt_Descripcion = Empty
    txt_Descripcion.SetFocus

```



```
cmd_grabar.Enabled = True
End Sub
```

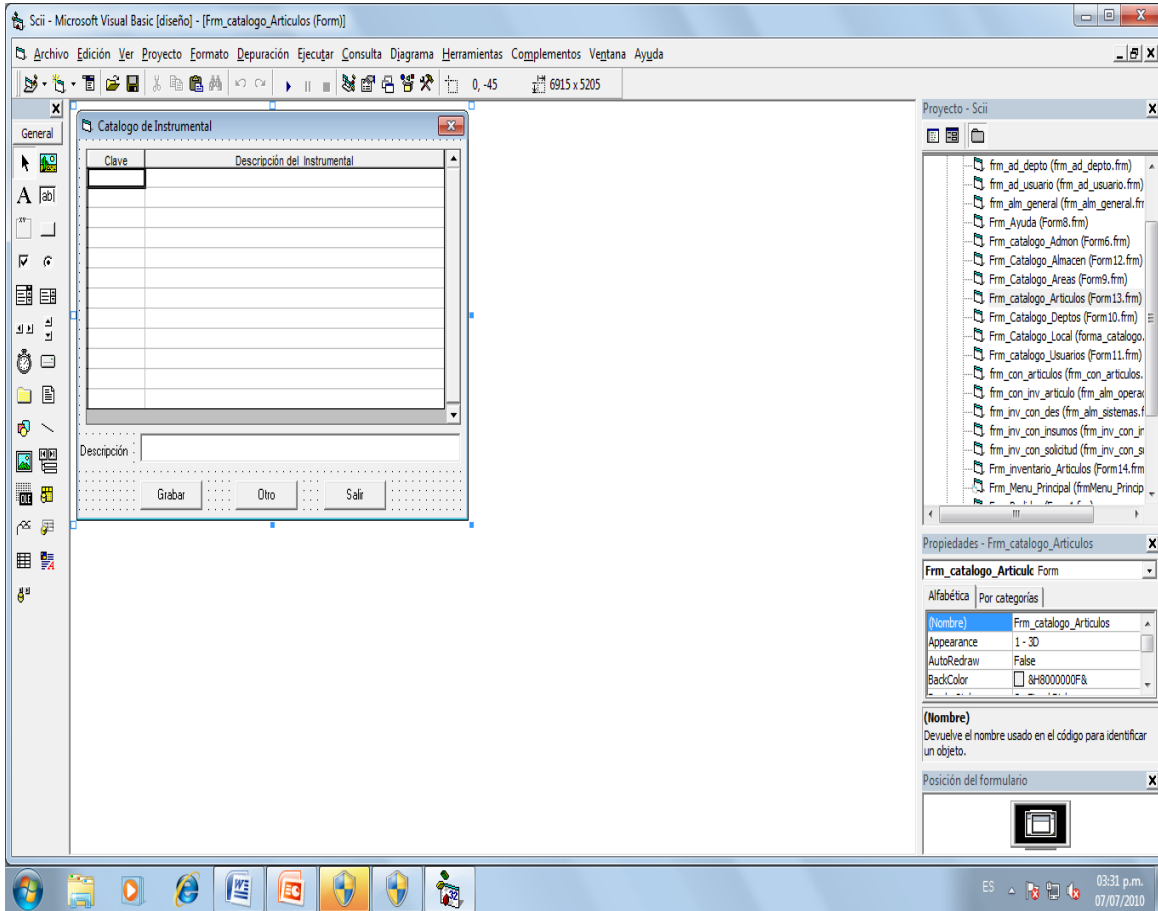
```
Private Sub cmd_Salir_Click()
Unload Me
End Sub
```

```
Private Sub Form_Load()
Call llena_catalogo
End Sub
```

```
Private Sub txt_Descripcion_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then cmd_grabar.SetFocus
If KeyAscii = 8 Then Exit Sub
Select Case Chr(KeyAscii)
Case "{", "}", "[", "]", "|", "\", "¿", "?", "¡", "!", "=", "/", "\", ""
KeyAscii = 0
Case Else
KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Select
```

```
End Sub
Private Sub cmd_Salir_Click()
Unload Me
End Sub
```

## Catalogo de Instrumental:



### Option Explicit

```
Private Sub cmd_nuevoarticulo_Click()  
txt_cve_instrumental = Empty  
txt_Descripcion = Empty  
txt_Descripcion.SetFocus  
cmd_grabar.Enabled = True  
End Sub
```

```
Private Sub cmd_salir_Click()  
Unload Me  
End Sub
```

```
Private Sub llena_catalogo()  
Screen.MousePointer = vbHourglass  
sSQL = "SELECT * FROM cat_instrumental ORDER BY c_cve_articulo"  
Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,  
dbSQLPassThrough)  
If Not recData.EOF And Not recData.BOF Then
```

```

    recData.MoveLast
    Listado.MaxRows = recData.RecordCount
    recData.MoveFirst
    iRen = 1
    Do While Not recData.EOF
Listado.Row = iRen
        Listado.Col = 1
        Listado.Text = recData("c_cve_articulo")
        Listado.Col = 2
        Listado.Text = recData("d_des_articulo")
    recData.MoveNext
        iRen = iRen + 1
    Loop
End If
    Screen.MousePointer = vbdefault
End Sub

Private Sub cmd_Grabar_Click()
If txt_Descripcion <> "" Then
    On Error GoTo Error_Grabando
    sSQL = "SELECT * FROM cat_instrumental WHERE c_cve_articulo IS NOT
NULL"

        Set recData = dbData.OpenRecordset(sSQL, dbOpenSnapshot,
dbSQLPassThrough)
        If Not recData.BOF And Not recData.EOF Then
            recData.MoveLast
            txt_cve_instrumental = recData.RecordCount + 1
            txt_cve_instrumental.Refresh
        Else
            txt_cve_instrumental = 1
            txt_cve_instrumental.Refresh
        End If
        wspEspacio.BeginTrans
        sSQL = "INSERT INTO cat_instrumental(c_cve_articulo,d_des_articulo)
VALUES(" & txt_cve_instrumental & "," & txt_Descripcion & ")"
        dbData.Execute sSQL, dbSQLPassThrough
        wspEspacio.CommitTrans
    Call llena_catalogo
        cmd_grabar.Enabled = False
End If
    Screen.MousePointer = vbdefault
    On Error GoTo 0
    Exit Sub
Error_Grabando:
    If Err.Number <> 0 Then
        Call Mensajes_de_Error("Grabando instrumental")
    End If
End Sub

```

End If

End Sub

Private Sub Form\_Load()

    Call llena\_catalogo

End Sub

Private Sub txt\_Descripcion\_KeyPress(KeyAscii As Integer)

    If KeyAscii = 13 Then cmd\_grabar.SetFocus

    If KeyAscii = 8 Then Exit Sub

    Select Case Chr(KeyAscii)

        Case "{", "}", "[", "]", "|", "\", "¿", "?", "¡", "!", "=", "/", "\\", ""

            KeyAscii = 0

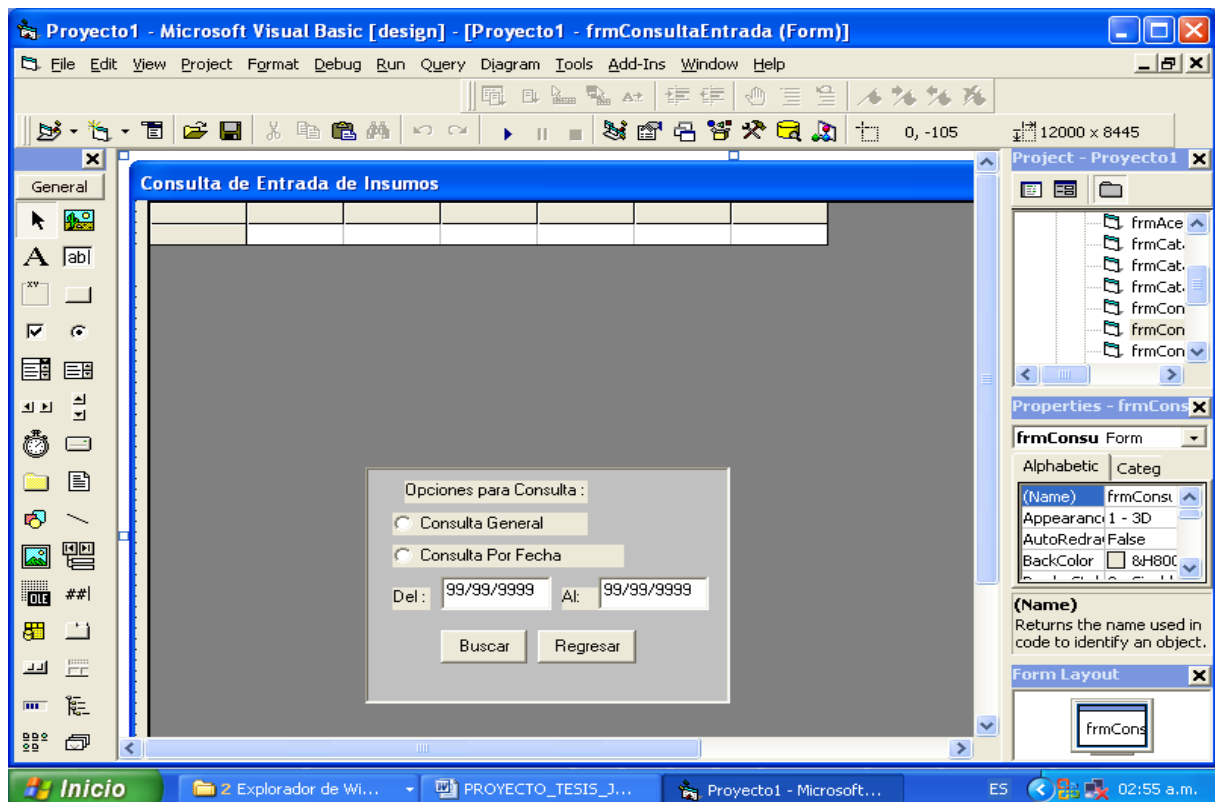
        Case Else

            KeyAscii = Asc(UCase(Chr(KeyAscii)))

    End Select

End Sub

Esta pantalla muestra las opciones de consulta de Solicitudes:



Option Explicit

Private Sub cmd\_buscar\_Click()

On Error GoTo Error\_Grabando

iRen = 1

If SSOption1 Then

```
        dbsql = "SELECT CAT_INV_ENTRADAS.CVE_PEDIDO," & _  
" CAT_INV_ENTRADAS.CVE_INSUMO, CAT_INV_ENTRADAS.NOM_INSUMO,"  
& _  
        " CAT_INV_ENTRADAS.CAN_ENTRA,  
CAT_INV_ENTRADAS.FECHA_ENTRA," & _  
        " CAT_INV_ENTRADAS.CVE_SURTE,  
CAT_INV_ENTRADAS.NOM_SURTE," & _  
        " CAT_INV_ENTRADAS.CVE_RECIBE,  
CAT_INV_ENTRADAS.NOM_RECIBE " & _  
" FROM CAT_INV_ENTRADAS"
```

Set rsAux = dbData.OpenRecordset(dbsql)

If Not rsAux.BOF And Not rsAux.EOF Then

rsAux.MoveLast

msfg\_alm\_operacion.Rows = rsAux.RecordCount + 1

rsAux.MoveFirst

Do While Not rsAux.EOF

msfg\_alm\_operacion.Row = iRen

msfg\_alm\_operacion.Col = 1

msfg\_alm\_operacion.Text = rsAux("cve\_pedido")

msfg\_alm\_operacion.Col = 2

msfg\_alm\_operacion.Text = rsAux("nom\_insumo")

msfg\_alm\_operacion.Col = 3

msfg\_alm\_operacion.Text = rsAux("fecha\_entra")

msfg\_alm\_operacion.Col = 4

msfg\_alm\_operacion.Text = rsAux("can\_entra")

msfg\_alm\_operacion.Col = 5

msfg\_alm\_operacion.Text = rsAux("nom\_surte")

msfg\_alm\_operacion.Col = 6

msfg\_alm\_operacion.Text = rsAux("nom\_recibe")

msfg\_alm\_operacion.ColAlignment(6) = 6

rsAux.MoveNext

iRen = iRen + 1

Loop

End If

Else

If SSOption2 Then

```

dbsql = "SELECT CAT_INV_ENTRADAS.CVE_PEDIDO,
CAT_INV_ENTRADAS.CVE_INSUMO," & _
" CAT_INV_ENTRADAS.NOM_INSUMO, CAT_INV_ENTRADAS.CAN_ENTRA," &
_
" CAT_INV_ENTRADAS.FECHA_ENTRA,
CAT_INV_ENTRADAS.CVE_SURTE," & _
" CAT_INV_ENTRADAS.NOM_SURTE,
CAT_INV_ENTRADAS.CVE_RECIBE," & _
" CAT_INV_ENTRADAS.NOM_RECIBE" & _
" FROM CAT_INV_ENTRADAS WHERE
(((CAT_INV_ENTRADAS.FECHA_ENTRA) Between #" & Inicio & " # And # " &
Final & " #))"
Set rsAux = dbData.OpenRecordset(dbsql)
If Not rsAux.BOF And Not rsAux.EOF Then
    rsAux.MoveLast
    msfg_alm_operacion.Rows = rsAux.RecordCount + 1
    rsAux.MoveFirst
    Do While Not rsAux.EOF
        msfg_alm_operacion.Row = iRen
        msfg_alm_operacion.Col = 1
        msfg_alm_operacion.Text = rsAux("cve_pedido")
        msfg_alm_operacion.Col = 2
        msfg_alm_operacion.Text = rsAux("nom_insumo")
        msfg_alm_operacion.Col = 3
msfg_alm_operacion.Text = rsAux("fecha_entra")
        msfg_alm_operacion.Col = 4
        msfg_alm_operacion.Text = rsAux("can_entra")
        msfg_alm_operacion.Col = 5
msfg_alm_operacion.Text = rsAux("nom_surte")
        msfg_alm_operacion.Col = 6
        msfg_alm_operacion.Text = rsAux("nom_recibe")
        msfg_alm_operacion.ColAlignment(6) = 6
        rsAux.MoveNext
        iRen = iRen + 1
    Loop
End If
End If
End If

msfg_alm_operacion.MergeCol(1) = True
msfg_alm_operacion.MergeCol(2) = True
msfg_alm_operacion.MergeCol(3) = True
msfg_alm_operacion.MergeCol(4) = True
msfg_alm_operacion.MergeCol(5) = True
msfg_alm_operacion.MergeCol(6) = True
msfg_alm_operacion.Col = 1
msfg_alm_operacion.ColSel = msfg_alm_operacion.Cols - 1

```

```

    msfg_alm_operacion.Sort = flexSortGenericAscending
cmd_otra.SetFocus
    ssp_Consulta.Visible = False
    On Error GoTo 0
    Exit Sub
Error_Grabando:
    If Err.Number <> 0 Then
        Call Mensajes_de_Error("NO EXITE LA FECHA SOLICITADA")
    End If

End Sub

Private Sub cmd_otra_Click()
    msfg_alm_operacion.Rows = 1
End Sub

Private Sub cmd_Regresar_Click()
    cmd_otra.SetFocus
    ssp_Consulta.Visible = False
End Sub

Private Sub cmd_salir_Click()
Unload Me
End Sub

Private Sub Command1_Click()
ssp_Consulta.Visible = True
    Inicio = limpia
    Final = limpia
    cmd_buscar.SetFocus
End Sub

```

```

Private Sub Form_Load()
iRen = 1
| *****
| * Encabezados en las columnas fijas y anchura de las columnas *
| *****

msfg_alm_operacion.Rows = iRen + 1
msfg_alm_operacion.ColWidth(0) = 0
msfg_alm_operacion.FixedAlignment(1) = flexAlignCenterCenter:
msfg_alm_operacion.TextArray(1) = "Pedido"
msfg_alm_operacion.ColWidth(1) = 800
msfg_alm_operacion.FixedAlignment(2) = flexAlignCenterCenter:
msfg_alm_operacion.TextArray(2) = "Insumo"
msfg_alm_operacion.ColWidth(2) = 3500
msfg_alm_operacion.FixedAlignment(3) = flexAlignCenterCenter:
msfg_alm_operacion.TextArray(3) = "Fecha Entrada"
msfg_alm_operacion.ColWidth(3) = 1500
msfg_alm_operacion.FixedAlignment(4) = flexAlignCenterCenter:
msfg_alm_operacion.TextArray(4) = "Cantidad"
msfg_alm_operacion.ColWidth(4) = 800
msfg_alm_operacion.FixedAlignment(5) = flexAlignCenterCenter:
msfg_alm_operacion.TextArray(5) = "Surte"
msfg_alm_operacion.ColWidth(5) = 3000
msfg_alm_operacion.FixedAlignment(6) = flexAlignCenterCenter:
msfg_alm_operacion.TextArray(6) = "Recibe"
msfg_alm_operacion.ColWidth(6) = 4000
End Sub

'Private Sub SSOption1_Click(Value As Integer)
' cmd_buscar.SetFocus
'End Sub

'Private Sub SSOption2_Click(Value As Integer)
' Inicio.SetFocus
'End Sub

Private Sub inicio_KeyPress(KeyAscii As Integer)
If KeyAscii = 8 Then Exit Sub
If KeyAscii = 13 Then
inicio_LostFocus
End If
End Sub

```



```

Private Sub inicio_LostFocus()
    If Inicio <> limpia And InStr(1, Inicio, " ") = 0 Then
        If InStr(1, Inicio, " ") > 0 Or Not IsDate(Inicio) Or Mid(Inicio, 4, 2) > 12 Then
            MsgBox "Fecha Incorrecta", 48, At
            Inicio = limpia
        Inicio.SetFocus
    Else
        Final.SetFocus
    End If
    Else
        Final.SetFocus
    End If
End Sub

```

```

Private Sub final_KeyPress(KeyAscii As Integer)
    If KeyAscii = 8 Then Exit Sub
    If KeyAscii = 13 Then
        final_LostFocus
    End If
End Sub

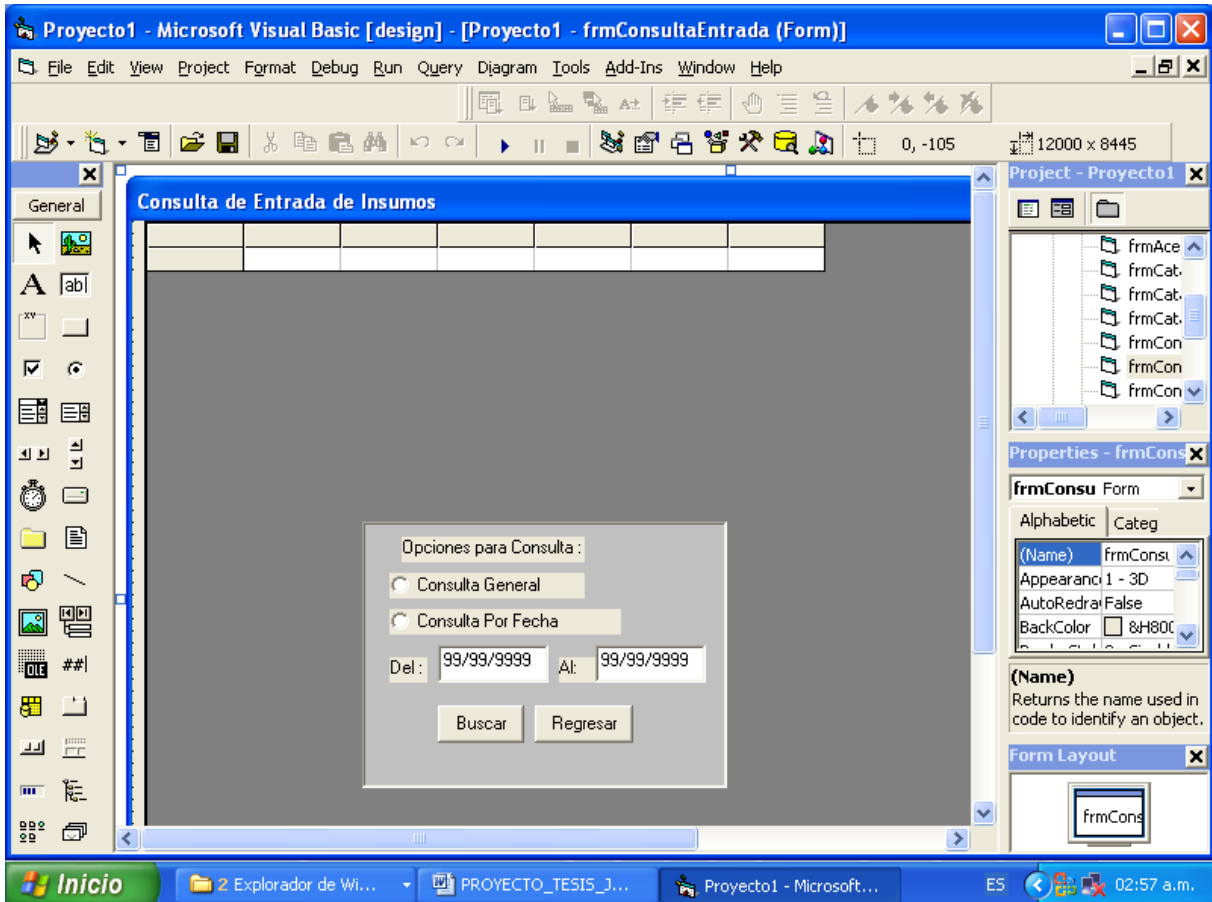
```

```

Private Sub final_LostFocus()
    If Final <> limpia And InStr(1, Final, " ") = 0 Then
        If InStr(1, Final, " ") > 0 Or Not IsDate(Final) Or Mid(Final, 4, 2) > 12 Then
            MsgBox "Fecha Incorrecta ", 48, At
            Final = limpia
            Final = limpia
            Final.SetFocus
        Else
            cmd_buscar.SetFocus
    End If
    Else
        cmd_buscar.SetFocus
    End If
End Sub

```

Pantalla de consulta de devoluciones:



Option Explicit

Private Sub cmd\_buscar\_Click()

On Error GoTo Error\_Grabando

iRen = 1

    If SSOption1 Then

```

        dbsql = "SELECT CAT_INV_ENTRADAS.CVE_PEDIDO," & _
" CAT_INV_ENTRADAS.CVE_INSUMO, CAT_INV_ENTRADAS.NOM_INSUMO," & _
" CAT_INV_ENTRADAS.CAN_ENTRA, CAT_INV_ENTRADAS.FECHA_ENTRA," & _
" CAT_INV_ENTRADAS.CVE_SURTE, CAT_INV_ENTRADAS.NOM_SURTE," & _
" CAT_INV_ENTRADAS.CVE_RECIBE, CAT_INV_ENTRADAS.NOM_RECIBE " & _
" FROM CAT_INV_ENTRADAS"
    
```

```

Set rsAux = dbData.OpenRecordset(dbsql)
  If Not rsAux.BOF And Not rsAux.EOF Then
    rsAux.MoveLast
    msfg_alm_operacion.Rows = rsAux.RecordCount + 1
    rsAux.MoveFirst
    Do While Not rsAux.EOF
      msfg_alm_operacion.Row = iRen
      msfg_alm_operacion.Col = 1
      msfg_alm_operacion.Text = rsAux("cve_pedido")
      msfg_alm_operacion.Col = 2
      msfg_alm_operacion.Text = rsAux("nom_insumo")
      msfg_alm_operacion.Col = 3
msfg_alm_operacion.Text = rsAux("fecha_entra")
      msfg_alm_operacion.Col = 4
      msfg_alm_operacion.Text = rsAux("can_entra")
      msfg_alm_operacion.Col = 5
msfg_alm_operacion.Text = rsAux("nom_surte")
      msfg_alm_operacion.Col = 6
      msfg_alm_operacion.Text = rsAux("nom_recibe")
      msfg_alm_operacion.ColAlignment(6) = 6
      rsAux.MoveNext
      iRen = iRen + 1
    Loop

```

```

        End If
    Else

        If SSOption2 Then
            dbsql = "SELECT CAT_INV_ENTRADAS.CVE_PEDIDO,
            CAT_INV_ENTRADAS.CVE_INSUMO," & _
            " CAT_INV_ENTRADAS.NOM_INSUMO, CAT_INV_ENTRADAS.CAN_ENTRA," &
            _
            " CAT_INV_ENTRADAS.FECHA_ENTRA,
            CAT_INV_ENTRADAS.CVE_SURTE," & _
            " CAT_INV_ENTRADAS.NOM_SURTE,
            CAT_INV_ENTRADAS.CVE_RECIBE," & _
            " CAT_INV_ENTRADAS.NOM_RECIBE" & _
            " FROM CAT_INV_ENTRADAS WHERE
            (((CAT_INV_ENTRADAS.FECHA_ENTRA) Between #" & Inicio & " # And # " &
            Final & " #))"
            Set rsAux = dbData.OpenRecordset(dbsql)
            If Not rsAux.BOF And Not rsAux.EOF Then
                rsAux.MoveLast
                msfg_alm_operacion.Rows = rsAux.RecordCount + 1
                rsAux.MoveFirst
                Do While Not rsAux.EOF
                    msfg_alm_operacion.Row = iRen
                    msfg_alm_operacion.Col = 1
                    msfg_alm_operacion.Text = rsAux("cve_pedido")
                    msfg_alm_operacion.Col = 2
                    msfg_alm_operacion.Text = rsAux("nom_insumo")
                    msfg_alm_operacion.Col = 3
                    msfg_alm_operacion.Text = rsAux("fecha_entra")
                    msfg_alm_operacion.Col = 4
                    msfg_alm_operacion.Text = rsAux("can_entra")
                    msfg_alm_operacion.Col = 5
                    msfg_alm_operacion.Text = rsAux("nom_surte")
                    msfg_alm_operacion.Col = 6
                    msfg_alm_operacion.Text = rsAux("nom_recibe")
                    msfg_alm_operacion.ColAlignment(6) = 6
                    rsAux.MoveNext
                    iRen = iRen + 1
                Loop
            End If
        End If
    End If
End If

```

```

msfg_alm_operacion.MergeCol(1) = True
msfg_alm_operacion.MergeCol(2) = True
msfg_alm_operacion.MergeCol(3) = True
msfg_alm_operacion.MergeCol(4) = True
msfg_alm_operacion.MergeCol(5) = True
msfg_alm_operacion.MergeCol(6) = True
msfg_alm_operacion.Col = 1
msfg_alm_operacion.ColSel = msfg_alm_operacion.Cols - 1
msfg_alm_operacion.Sort = flexSortGenericAscending
cmd_otra.SetFocus
ssp_consulta.Visible = False
On Error GoTo 0
Exit Sub
Error_Grabando:
If Err.Number <> 0 Then
    Call Mensajes_de_Error("NO EXITE LA FECHA SOLICITADA")
End If

End Sub

Private Sub cmd_otra_Click()
    msfg_alm_operacion.Rows = 1
End Sub

Private Sub cmd_Regresar_Click()
    cmd_otra.SetFocus
    ssp_consulta.Visible = False
End Sub

Private Sub cmd_salir_Click()
Unload Me
End Sub

Private Sub Command1_Click()
ssp_consulta.Visible = True
Inicio = limpia

```

```

Final = limpia
    cmd_buscar.SetFocus
End Sub

```

```

Private Sub Form_Load()

```

```

    iRen = 1

```

```

    ! *****

```

```

    ! * Encabezados en las columnas fijas y anchura de las columnas *

```

```

    ! *****

```

```

        msfg_alm_operacion.Rows = iRen + 1
        msfg_alm_operacion.ColWidth(0) = 0
        msfg_alm_operacion.FixedAlignment(1) = flexAlignCenterCenter:
msfg_alm_operacion.TextArray(1) = "Pedido"
        msfg_alm_operacion.ColWidth(1) = 800
        msfg_alm_operacion.FixedAlignment(2) = flexAlignCenterCenter:
msfg_alm_operacion.TextArray(2) = "Insumo"
        msfg_alm_operacion.ColWidth(2) = 3500
        msfg_alm_operacion.FixedAlignment(3) = flexAlignCenterCenter:
msfg_alm_operacion.TextArray(3) = "Fecha Entrada"
        msfg_alm_operacion.ColWidth(3) = 1500
        msfg_alm_operacion.FixedAlignment(4) = flexAlignCenterCenter:
msfg_alm_operacion.TextArray(4) = "Cantidad"
        msfg_alm_operacion.ColWidth(4) = 800
        msfg_alm_operacion.FixedAlignment(5) = flexAlignCenterCenter:
msfg_alm_operacion.TextArray(5) = "Surte"
        msfg_alm_operacion.ColWidth(5) = 3000
        msfg_alm_operacion.FixedAlignment(6) = flexAlignCenterCenter:
msfg_alm_operacion.TextArray(6) = "Recibe"
        msfg_alm_operacion.ColWidth(6) = 4000
    End Sub

```

```

'Private Sub SSOption1_Click(Value As Integer)

```

```

    ' cmd_buscar.SetFocus

```

```

'End Sub

```

```

'Private Sub SSOption2_Click(Value As Integer)

```

```

    ' Inicio.SetFocus

```

```

'End Sub

```

```

Private Sub inicio_KeyPress(KeyAscii As Integer)
    If KeyAscii = 8 Then Exit Sub
    If KeyAscii = 13 Then
        inicio_LostFocus
    End If
End Sub

```

```

Private Sub inicio_LostFocus()
    If Inicio <> limpia And InStr(1, Inicio, " ") = 0 Then
        If InStr(1, Inicio, " ") > 0 Or Not IsDate(Inicio) Or Mid(Inicio, 4, 2) > 12 Then
            MsgBox "Fecha Incorrecta", 48, At
            Inicio = limpia
            Inicio.SetFocus
        Else
            Final.SetFocus
        End If
    Else
        Final.SetFocus
    End If
End Sub

```

```

Private Sub final_KeyPress(KeyAscii As Integer)
    If KeyAscii = 8 Then Exit Sub
    If KeyAscii = 13 Then
        final_LostFocus
    End If
End Sub

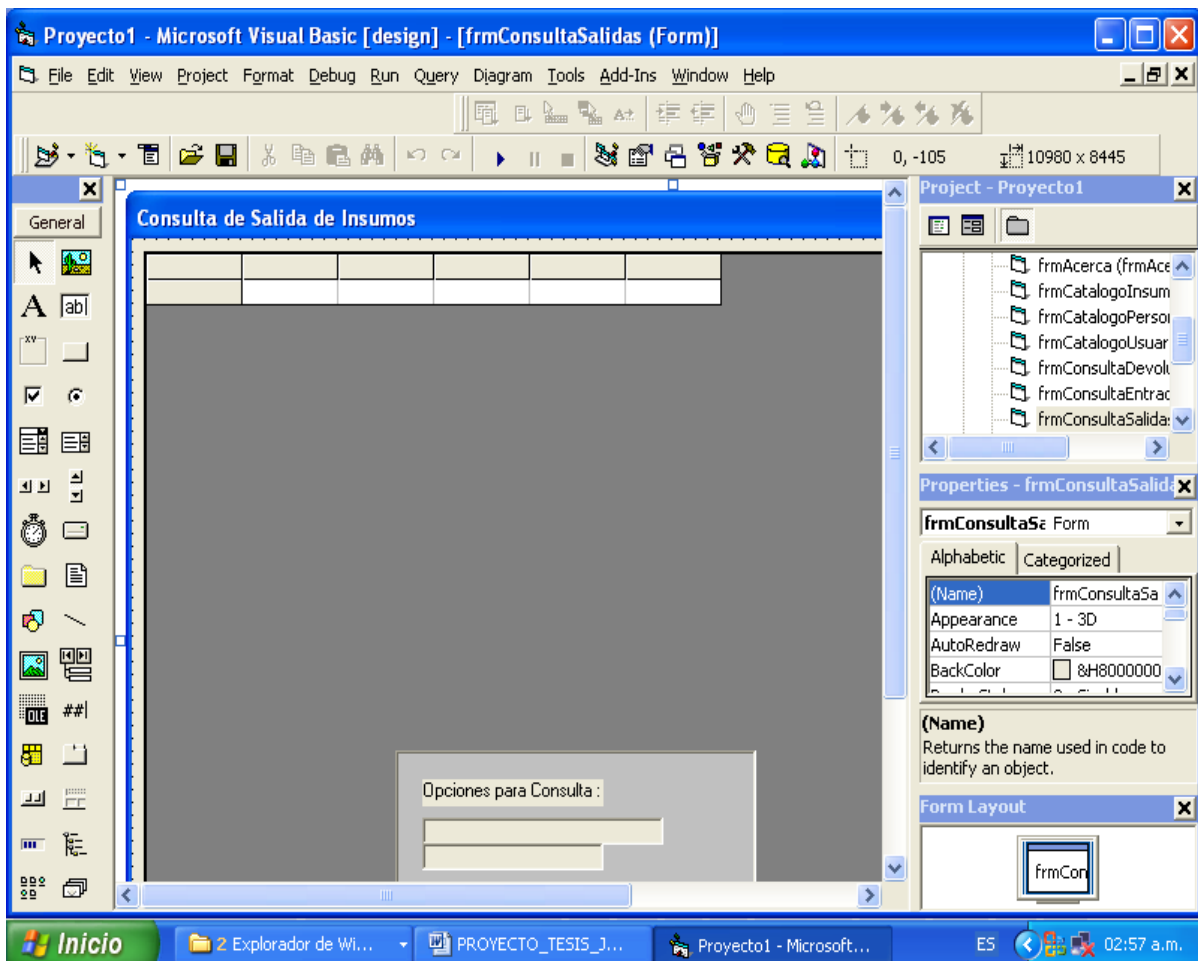
```

```

Private Sub final_LostFocus()
    If Final <> limpia And InStr(1, Final, " ") = 0 Then
        If InStr(1, Final, " ") > 0 Or Not IsDate(Final) Or Mid(Final, 4, 2) > 12 Then
            MsgBox "Fecha Incorrecta ", 48, At
            Final = limpia
            Final = limpia
            Final.SetFocus
        Else
            cmd_buscar.SetFocus
        End If
    Else
        cmd_buscar.SetFocus
    End If
End Sub

```

## Pantalla de consulta de salidas de instrumental



Option Explicit

Private Sub cmd\_buscar\_Click()

On Error GoTo Error\_Grabando

iRen = 1

    If SSOption1 Then

```
        dbsql = "SELECT CAT_INV_SALIDAS.CVE_PEDIDO," & _  
" CAT_INV_SALIDAS.CVE_INSUMO, CAT_INV_SALIDAS.NOM_INSUMO," & _  
                " CAT_INV_SALIDAS.CAN_SALE,  
CAT_INV_SALIDAS.FECHA_SALE," & _  
                " CAT_INV_SALIDAS.CVE_ENTREGA,  
CAT_INV_SALIDAS.NOM_ENTREGA," & _  
                " CAT_INV_SALIDAS.OBSERVACIONES" & _
```



```
" FROM CAT_INV_SALIDAS"
```

```
'      dbsql = "SELECT CAT_INV_DEVOLUCIONES.CVE_PEDIDO," & _  
'          " CAT_INV_DEVOLUCIONES.CVE_INSUMO," & _  
'          " CAT_INV_DEVOLUCIONES.NOM_INSUMO," & _  
'          " CAT_INV_DEVOLUCIONES.CAN_ENTRA," & _  
'          " CAT_INV_DEVOLUCIONES.FECHA_ENTRA," & _  
'          " CAT_INV_DEVOLUCIONES.NOM_RECIBE," & _  
'          " CAT_INV_DEVOLUCIONES.NOM_DEVUELVE," & _  
'          " CAT_INV_DEVOLUCIONES.NOM_AREA " & _  
'          " FROM CAT_INV_DEVOLUCIONES"
```

```
Set rsAux = dbData.OpenRecordset(dbsql)  
If Not rsAux.BOF And Not rsAux.EOF Then  
    rsAux.MoveLast  
    msfg_alm_operacion.Rows = rsAux.RecordCount + 1  
    rsAux.MoveFirst  
    Do While Not rsAux.EOF  
        msfg_alm_operacion.Row = iRen  
        msfg_alm_operacion.Col = 1  
            msfg_alm_operacion.Text = rsAux("cve_pedido")  
            msfg_alm_operacion.Col = 2  
            msfg_alm_operacion.Text = rsAux("nom_insumo")  
            msfg_alm_operacion.Col = 3  
msfg_alm_operacion.Text = rsAux("fecha_sale")  
msfg_alm_operacion.Col = 4  
            msfg_alm_operacion.Text = rsAux("can_sale")  
            msfg_alm_operacion.Col = 5  
            msfg_alm_operacion.Text = rsAux("nom_entrega")  
            msfg_alm_operacion.ColAlignment(5) = 5  
            rsAux.MoveNext  
            iRen = iRen + 1  
        Loop  
    End If  
Else  
  
If SSOption2 Then
```

```
    dbsql = "SELECT CAT_INV_DEVOLUCIONES.CVE_PEDIDO," & _  
" CAT_INV_DEVOLUCIONES.CVE_INSUMO," & _  
        " CAT_INV_DEVOLUCIONES.NOM_INSUMO," & _  
        " CAT_INV_DEVOLUCIONES.CAN_ENTRA," & _  
        " CAT_INV_DEVOLUCIONES.FECHA_ENTRA," & _  
        " CAT_INV_DEVOLUCIONES.NOM_RECIBE," & _  
        " CAT_INV_DEVOLUCIONES.NOM_DEVUELVE," & _
```

```

" CAT_INV_DEVOLUCIONES.NOM_AREA " & _
      " FROM CAT_INV_SALIDAS WHERE
(((CAT_INV_SALIDAS.FECHA_SALE) Between #" & Inicio & " # And # " & Final &
" #))"
    Set rsAux = dbData.OpenRecordset(dbsql)
    If Not rsAux.BOF And Not rsAux.EOF Then
        rsAux.MoveLast
        msfg_alm_operacion.Rows = rsAux.RecordCount + 1
        rsAux.MoveFirst
        Do While Not rsAux.EOF
            msfg_alm_operacion.Row = iRen
            msfg_alm_operacion.Col = 1
            msfg_alm_operacion.Text = rsAux("cve_pedido")
            msfg_alm_operacion.Col = 2
            msfg_alm_operacion.Text = rsAux("nom_insumo")
            msfg_alm_operacion.Col = 3
msfg_alm_operacion.Text = rsAux("fecha_sale")
msfg_alm_operacion.Col = 4
            msfg_alm_operacion.Text = rsAux("can_sale")
            msfg_alm_operacion.Col = 5
            msfg_alm_operacion.Text = rsAux("nom_entrega")
            msfg_alm_operacion.ColAlignment(5) = 5
            rsAux.MoveNext
            iRen = iRen + 1
        Loop
    End If
End If
msfg_alm_operacion.MergeCol(1) = True
msfg_alm_operacion.MergeCol(2) = True
msfg_alm_operacion.MergeCol(3) = True
msfg_alm_operacion.MergeCol(4) = True
msfg_alm_operacion.MergeCol(5) = True
msfg_alm_operacion.Col = 1
msfg_alm_operacion.ColSel = msfg_alm_operacion.Cols - 1
msfg_alm_operacion.Sort = flexSortGenericAscending
cmd_otra.SetFocus
ssp_consulta.Visible = False
On Error GoTo 0
Exit Sub
Error_Grabando:
If Err.Number <> 0 Then
    Call Mensajes_de_Error("NO EXITE LA FECHA SOLICITADA")
End If

```

```

End Sub
Private Sub cmd_otra_Click()
    msfg_alm_operacion.Rows = 1
End Sub

Private Sub cmd_Regresar_Click()
    cmd_otra.SetFocus
    ssp_Consulta.Visible = False
End Sub

Private Sub cmd_salir_Click()
Unload Me
End Sub

Private Sub Command1_Click()
ssp_Consulta.Visible = True
    Inicio = limpia
Final = limpia
    cmd_buscar.SetFocus
End Sub

Private Sub Form_Load()
iRen = 1
    ! *****
    ! * Encabezados en las columnas fijas y anchura de las columnas *
    ! *****

    msfg_alm_operacion.Rows = iRen + 1
    msfg_alm_operacion.ColWidth(0) = 0
    msfg_alm_operacion.FixedAlignment(1) = flexAlignCenterCenter:
msfg_alm_operacion.TextArray(1) = "Pedido"
    msfg_alm_operacion.ColWidth(1) = 800
    msfg_alm_operacion.FixedAlignment(2) = flexAlignCenterCenter:
msfg_alm_operacion.TextArray(2) = "Insumo"
    msfg_alm_operacion.ColWidth(2) = 3500
    msfg_alm_operacion.FixedAlignment(3) = flexAlignCenterCenter:
msfg_alm_operacion.TextArray(3) = "Fecha de Salida"
    msfg_alm_operacion.ColWidth(3) = 1500
    msfg_alm_operacion.FixedAlignment(4) = flexAlignCenterCenter:
msfg_alm_operacion.TextArray(4) = "Cantidad"
    msfg_alm_operacion.ColWidth(4) = 800

```

```
msfg_alm_operacion.FixedAlignment(5) = flexAlignCenterCenter:
msfg_alm_operacion.TextArray(5) = "Persona que Entrega"
msfg_alm_operacion.ColWidth(5) = 4500
End Sub
```

```
'Private Sub SSOption1_Click(Value As Integer)
' cmd_buscar.SetFocus
'End Sub
```

```
'Private Sub SSOption2_Click(Value As Integer)
' Inicio.SetFocus
'End Sub
```

```
Private Sub inicio_KeyPress(KeyAscii As Integer)
    If KeyAscii = 8 Then Exit Sub
    If KeyAscii = 13 Then
        inicio_LostFocus
    End If
End Sub
```

```
Private Sub inicio_LostFocus()
    If Inicio <> limpia And InStr(1, Inicio, " ") = 0 Then
        If InStr(1, Inicio, " ") > 0 Or Not IsDate(Inicio) Or Mid(Inicio, 4, 2) > 12 Then
            MsgBox "Fecha Incorrecta", 48, At
            Inicio = limpia
        Inicio.SetFocus
    Else
        Final.SetFocus
    End If
Else
    Final.SetFocus
End If
End Sub
```

```
Private Sub final_KeyPress(KeyAscii As Integer)
    If KeyAscii = 8 Then Exit Sub
    If KeyAscii = 13 Then
        final_LostFocus

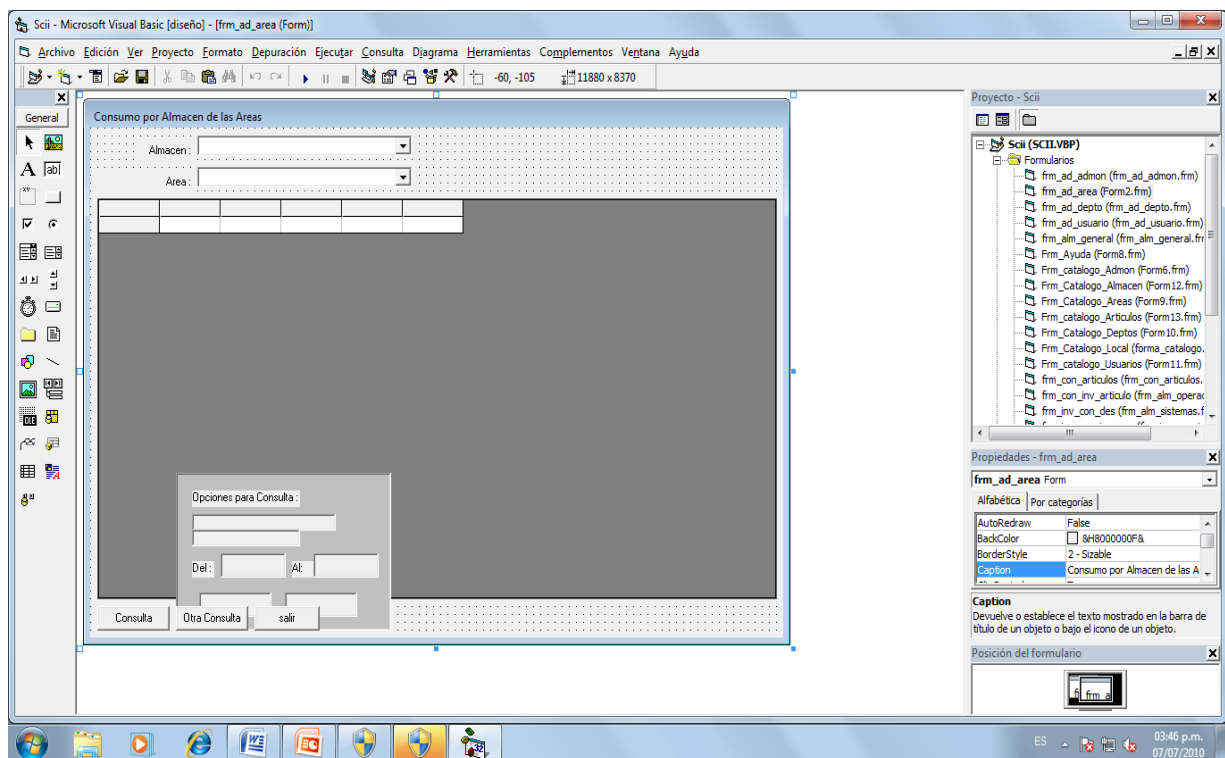
        End If
    End Sub
```

```

Private Sub final_LostFocus()
    If Final <> limpia And InStr(1, Final, " ") = 0 Then
        If InStr(1, Final, " ") > 0 Or Not IsDate(Final) Or Mid(Final, 4, 2) > 12 Then
            MsgBox "Fecha Incorrecta ", 48, At
            Final = limpia
            Final = limpia
            Final.SetFocus
        Else
            cmd_buscar.SetFocus
        End If
    Else
        cmd_buscar.SetFocus
    End If
End Sub

```

Pantalla de consulta de consumo por almacen:



Option Explicit

```
Private Sub cbo_almacen_Click()  
    If cbo_almacen <> "" Then  
        cmd_consulta.SetFocus  
    End If  
End Sub
```

```
Private Sub cmd_buscar_Click()  
    iRen = 1  
    If SSOption1 Then  
        If cbo_almacen <> "" Then
```

```
sSQL = "SELECT cat_articulos.d_des_articulo, mm_inv_articulos.d_nom_articulo,"  
& _  
        "mm_inv_gral.n_cantidad " & _  
"FROM mm_inv_articulos INNER JOIN (mm_inv_gral INNER JOIN cat_articulos "  
& _  
        "ON mm_inv_gral.c_cve_articulo = cat_articulos.c_cve_articulo) ON "  
& _  
        "(mm_inv_articulos.c_num_articulo = mm_inv_gral.c_num_articulo)  
AND " & _  
"(mm_inv_articulos.c_cve_articulo = cat_articulos.c_cve_articulo)"
```

```
    Set rsData = dbData.OpenRecordset(sSQL)  
    If Not rsData.BOF And Not rsData.EOF Then  
        rsData.MoveLast  
        msfg_alm_sistemas.Rows = rsData.RecordCount + 1  
        rsData.MoveFirst  
        Do While Not rsData.EOF  
            msfg_alm_sistemas.Row = iRen  
            msfg_alm_sistemas.Col = 1  
            msfg_alm_sistemas.Text = rsData("d_des_articulo")  
            msfg_alm_sistemas.Col = 2  
            msfg_alm_sistemas.Text = rsData("d_nom_articulo")  
            msfg_alm_sistemas.Col = 3  
            msfg_alm_sistemas.ColAlignment(3) = 3  
            msfg_alm_sistemas.Text = rsData("n_cantidad")  
            rsData.MoveNext  
            iRen = iRen + 1  
        Loop  
    End If
```

```

    End If
    Elseif SSOption2 Then
        If cbo_almacen <> "" Then
sSQL = "SELECT cat_articulos.d_des_articulo, mm_inv_articulos.d_nom_articulo,"
& _
            "mm_inv_gral.n_cantidad " & _
"FROM mm_inv_articulos INNER JOIN (mm_inv_gral INNER JOIN cat_articulos "
& _
            "ON mm_inv_gral.c_cve_articulo = cat_articulos.c_cve_articulo) ON "
& _
            "(mm_inv_articulos.c_num_articulo = mm_inv_gral.c_num_articulo)
AND " & _
"(mm_inv_articulos.c_cve_articulo = cat_articulos.c_cve_articulo)"

```

```

Set rsData = dbData.OpenRecordset(sSQL)
    If Not rsData.BOF And Not rsData.EOF Then
        rsData.MoveLast
        msfg_alm_sistemas.Rows = rsData.RecordCount + 1
        rsData.MoveFirst
        Do While Not rsData.EOF
            msfg_alm_sistemas.Row = iRen
            msfg_alm_sistemas.Col = 1
            msfg_alm_sistemas.Text = rsData("d_des_articulo")
            msfg_alm_sistemas.Col = 2
            msfg_alm_sistemas.Text = rsData("d_nom_articulo")
            msfg_alm_sistemas.Col = 3
            msfg_alm_sistemas.ColAlignment(3) = 3
msfg_alm_sistemas.Text = rsData("n_cantidad")
rsData.MoveNext
            iRen = iRen + 1
        Loop
    End If
End If
msfg_alm_sistemas.MergeCol(1) = True
msfg_alm_sistemas.MergeCol(2) = True
msfg_alm_sistemas.MergeCol(3) = True
msfg_alm_sistemas.Col = 1
msfg_alm_sistemas.ColSel = msfg_alm_sistemas.Cols - 1
msfg_alm_sistemas.Sort = flexSortGenericAscending
cmd_otra.SetFocus
ssp_Consulta.Visible = False

```

End Sub

```
Private Sub cmd_consulta_Click()  
    ssp_Consulta.Visible = True  
    Inicio = limpia  
    Final = limpia  
    cmd_buscar.SetFocus  
End Sub
```

```
Private Sub cmd_otra_Click()  
    msfg_alm_sistemas.Rows = 1  
End Sub
```

```
Private Sub cmd_Regresar_Click()  
    cmd_otra.SetFocus  
    ssp_Consulta.Visible = False  
End Sub
```

```
Private Sub cmd_salir_Click()  
Unload Me  
End Sub
```

```
Private Sub Form_Load()  
    ' ' rutina de error  
    sSQL = "SELECT * FROM cat_almacen ORDER BY c_cve_almacen"  
    Set rsData = dbData.OpenRecordset(sSQL)  
    If rsData.RecordCount > 0 Then  
        Do While Not rsData.EOF  
cbo_almacen.AddItem rsData("d_des_almacen") & String(30 -  
Len(rsData("d_des_almacen")), ".") & rsData("c_cve_almacen")  
rsData.MoveNext  
        Loop  
    End If  
    sSQL = "SELECT * FROM cat_almacen ORDER BY c_cve_almacen"  
    Set rsData = dbData.OpenRecordset(sSQL)  
    If rsData.RecordCount > 0 Then  
        Do While Not rsData.EOF  
  
cbo_almacen.AddItem rsData("d_des_almacen") & String(100 -  
Len(rsData("d_des_almacen")), ".") & rsData("c_cve_almacen")  
rsData.MoveNext  
        Loop  
    End If  
    If iAlmacen > 1 Then
```



```

cbo_almacen.Enabled = False
    cbo_almacen = Busca_Almacen(iAlmacen)
End If
iRen = 1

! *****
! * Encabezados en las columnas fijas y anchura de las columnas *
! *****

    msfg_alm_sistemas.Rows = iRen + 1
    msfg_alm_sistemas.ColWidth(0) = 0
    msfg_alm_sistemas.FixedAlignment(1) = flexAlignCenterCenter:
msfg_alm_sistemas.TextArray(1) = "Artículo"
    msfg_alm_sistemas.ColWidth(1) = 2800
    msfg_alm_sistemas.FixedAlignment(2) = flexAlignCenterCenter:
msfg_alm_sistemas.TextArray(2) = "Descripción"
    msfg_alm_sistemas.ColWidth(2) = 6000
    msfg_alm_sistemas.FixedAlignment(3) = flexAlignCenterCenter:
msfg_alm_sistemas.TextArray(3) = "Cantidad"
    msfg_alm_sistemas.ColWidth(3) = 1000
End Sub

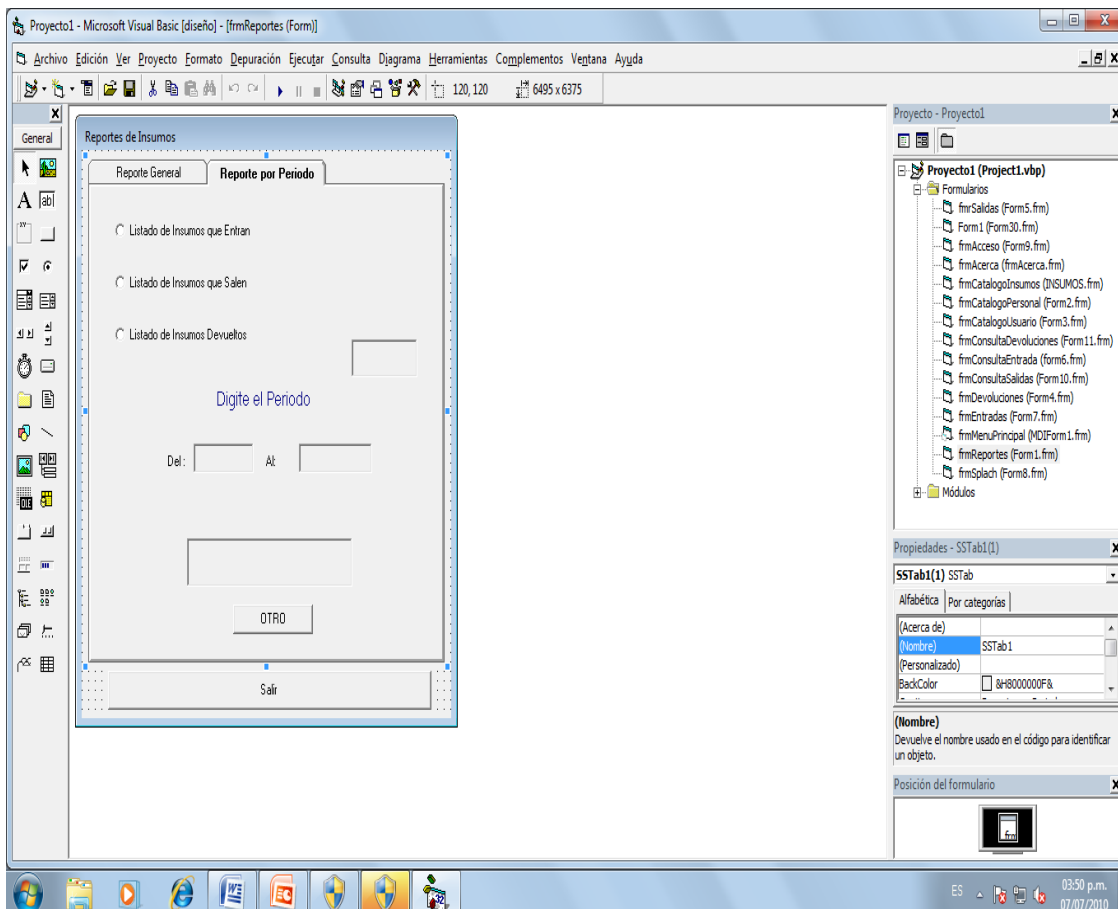
'Private Sub SSOption1_Click(Value As Integer)
'    cmd_buscar.SetFocus
'End Sub
'
'Private Sub SSOption2_Click(Value As Integer)
'    Inicio.SetFocus
'End Sub

```

## FASES DE DESARROLLO CARACTERISTICAS DEL SISTEMA

- **PANTALLAS PRINCIPALES DE  
REPORTES Y CODIGO**

Esta pantalla muestra los reportes en forma general y por periodo de los registros capturados:



Codigo a utilizar en esta pantalla:

```
Private Sub Command1_Click()
Unload Me
End Sub
Private Sub Command2_Click()
dbsql = " select * From cat_inv_devoluciones  "
  Set rsAux = dbData.OpenRecordset(dbsql)
  rsAux.MoveLast
  If rsAux.RecordCount = 0 Then

INICIO = 1
  INICIO.Refresh
  End If
  INICIO = "_/_/_"
  FINAL = "_/_/_"

  INICIO.SetFocus
End Sub

Private Sub final_KeyPress(KeyAscii As Integer)
If KeyAscii = 8 Then Exit Sub
  If KeyAscii = 13 Then
    final_LostFocus
  End If
End Sub

Private Sub final_LostFocus()
If FINAL <> limpia And InStr(1, FINAL, " ") = 0 Then
  If InStr(1, FINAL, " ") > 0 Or Not IsDate(FINAL) Or Mid(FINAL, 4, 2) > 12 Then
MsgBox "Fecha Incorrecta ", 48, At
  FINAL = limpia
FINAL = limpia
  FINAL.SetFocus
  Else
    imprime1.SetFocus
  End If
Else
  imprime1.SetFocus
  End If
End Sub

Private Sub IMPRIME_Click()
If Option1(0).Value = True Then
REP_ENTRADA.ReportFileName = "C:\AALMACEN\REPORTES\ENTRADA1.rpt"
REP_ENTRADA.SelectionFormula = ""
```

```

    REP_ENTRADA.Action = 1
End If

IMPRIME.SetFocus

If Option2(1).Value = True Then
REP_ENTRADA.ReportFileName = "C:\AALMACEN\REPORTES\SALidas.rpt"
REP_ENTRADA.SelectionFormula = ""
    REP_ENTRADA.Action = 1
End If
    IMPRIME.SetFocus
    If Option3(2).Value = True Then
REP_ENTRADA.ReportFileName =
"C:\AALMACEN\REPORTES\DEVOLUCIONES.rpt"
REP_ENTRADA.SelectionFormula = ""
    REP_ENTRADA.Action = 1
End If
    IMPRIME.SetFocus
End Sub
Private Sub imprime1_Click()
    Dim entrada As String
    Dim salidas
    If Option4.Value = True Then
        If INICIO <> " / / " Then
            If IsDate(INICIO) And IsDate(FINAL) Then
dbsql = "SELECT CAT_INV_ENTRADAS.CVE_PEDIDO,
CAT_INV_ENTRADAS.CVE_INSUMO," & _
        " CAT_INV_ENTRADAS.NOM_INSUMO,
CAT_INV_ENTRADAS.CAN_ENTRA," & _
        " CAT_INV_ENTRADAS.FECHA_ENTRA,
CAT_INV_ENTRADAS.CVE_SURTE," & _
        " CAT_INV_ENTRADAS.NOM_SURTE,
CAT_INV_ENTRADAS.CVE_RECIBE," & _
        " CAT_INV_ENTRADAS.NOM_RECIBE" & _
        " FROM CAT_INV_ENTRADAS WHERE
(((CAT_INV_ENTRADAS.FECHA_ENTRA) Between " & INICIO & " And " &
FINAL & " ))"
                ENTRADAN.ReportFileName =
"C:\AALMACEN\REPORTES\entrada1.rpt"
                ENTRADAN.SelectionFormula =
"{CAT_INV_ENTRADAS.FECHA_ENTRA}>=Date(" & _
                Mid(INICIO, 7, 4) & "," & Mid(INICIO, 4, 2) & "," & _
                Mid(INICIO, 1, 2) & ") AND
{CAT_INV_ENTRADAS.FECHA_ENTRA}<=Date(" & _
                Mid(FINAL, 7, 4) & "," & Mid(FINAL, 4, 2) & "," & _
                Mid(FINAL, 1, 2) & ")"
                ENTRADAN.Action = 1
            End If
        End If
    End If
End Sub

```

```

End If
  End If
End If

If Option5.Value = True Then
  If INICIO <> " / / " Then
    If IsDate(INICIO) And IsDate(FINAL) Then
      ENTRADAN.ReportFileName =
"C:\AALMACEN\REPORTES\salidas.rpt"
      ENTRADAN.SelectionFormula =
"{CAT_INV_SALIDAS.FECHA_SALE}>=Date(" & _
Mid(INICIO, 7, 4) & "," & Mid(INICIO, 4, 2) & "," & _
Mid(INICIO, 1, 2) & ") AND
{CAT_INV_SALIDAS.FECHA_SALE}<=Date(" & _
Mid(FINAL, 7, 4) & "," & Mid(FINAL, 4, 2) & "," & _
Mid(FINAL, 1, 2) & ")"
      ENTRADAN.Action = 1

    End If
  End If
End If

If Option6.Value = True Then
  If INICIO <> " / / " Then
    If IsDate(INICIO) And IsDate(FINAL) Then
      ENTRADAN.ReportFileName =
"C:\AALMACEN\REPORTES\DEVOLUCIONES.rpt"
      ENTRADAN.SelectionFormula =
"{CAT_INV_DEVOLUCIONES.FECHA_ENTRA}>=Date(" & _
Mid(INICIO, 7, 4) & "," & Mid(INICIO, 4, 2) & "," & _
Mid(INICIO, 1, 2) & ") AND
{CAT_INV_DEVOLUCIONES.FECHA_ENTRA}<=Date(" & _
Mid(FINAL, 7, 4) & "," & Mid(FINAL, 4, 2) & "," & _
Mid(FINAL, 1, 2) & ")"
      ENTRADAN.Action = 1

    End If
  End If
End If
Option4.SetFocus
End Sub
Private Sub inicio_KeyPress(KeyAscii As Integer)
If KeyAscii = 8 Then Exit Sub

```

```

        If KeyAscii = 13 Then
            inicio_LostFocus
        End If
    End Sub
Private Sub inicio_LostFocus()

    On Error GoTo Error_Grabando
    If INICIO <> limpia And InStr(1, INICIO, " ") = 0 Then
    If InStr(1, INICIO, " ") > 0 Or Not IsDate(INICIO) Or Mid(INICIO, 4, 2) > 12 Then
    MsgBox "Fecha Incorrecta", 48, At
        INICIO = limpia
    INICIO.SetFocus
    Else
        FINAL.SetFocus
    End If
    Else
        FINAL.SetFocus
    End If

    On Error GoTo 0
    Exit Sub
Error_Grabando:
    If Err.Number <> 0 Then
    Call Mensajes_de_Error("NO EXITE LA FECHA SOLICITADA")
    End If

End Sub

Private Sub Option4_Click()
    INICIO.SetFocus
End Sub

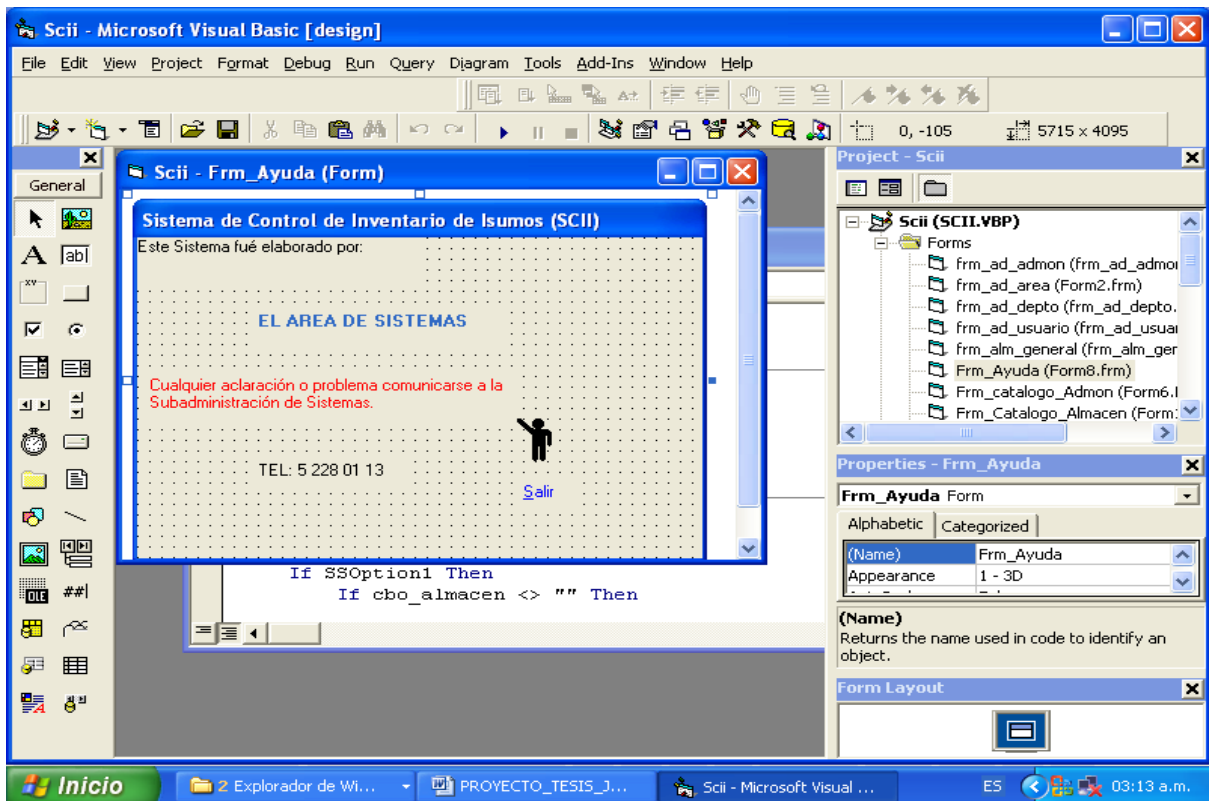
Private Sub Option5_Click()
    INICIO.SetFocus
End Sub

Private Sub Option6_Click()
    INICIO.SetFocus
End Sub

```

## FASES DE DESARROLLO CARACTERISTICAS DEL SISTEMA

- PANTALLA DE AYUDA EN SOPORTE
- TECNICO.



### **2.3.6.- PRUEBAS DEL SISTEMA OBJETIVOS Y PUNTOS A CONSIDERAR.**

El presente sistema tiene la finalidad de funcionar a corto plazo, motivo por el cual se hace presente llevar a acabo las pruebas necesarias para su funcionamiento y estas para dicho proyecto, se especificaron en “pruebas de corrida” y “en partalelo”, es decir con una función operatorio actual y el nuevo proyecto.

### **2.3.7.- INSTALACIÓN:**

El nuevo proyecto se puede instalar en cuanto sea necesario, siempre y cuando le de un visto bueno el responsable de la organización, para ponerlo en operación.

### **2.3.8.- MANTENIMIENTO:**

Preventivo.

El mantenimiento será destinado a prevenir cualquier eventualidad que pueda aparecer tanto en hardware como en software, que refiera el Sistema.





**Conclusiones:**

**Referente al desarrollo del presente sistema, se determina que cualquier cambio en el ámbito sistemático de un proceso es relevante, ya que se obtendrán beneficios que permitirán tener un mejor funcionamiento de las actividades correspondientes a la labor en Protección Civil, manifestando un saludable clima de actividades laborales.**

## **Bibliografía.**

### **Metodología de la programación**

Algoritmos, diagramas de flujo y programas  
Osvaldo Cairo  
Editorial Alfaomega.

### **Oracle 9i Programación pl/sql**

#### **Desarrolle potentes aplicaciones pl/sql**

Scout Urman  
Autor de títulos Oracle 8  
Programación pl/sql y oracle 8  
Mc Graw Hill

### **Iniciación al método del camino crítico**

Agustín Montaña  
Trillas.

### **Diseño de la base de datos relacionales**

Adoración de Miguel  
Mario Piattini  
Esperanza Marcos  
Alfaomega

### **Java 2**

Curso de programación  
Francisco Javier Ceballos  
Alfa omega

### **Vba con office 2000**

Peter monadjemi  
Alfaomega

### **Programación en algoritmos, estructuras de datos y objetos**

ANSI/iso c++  
C frente a c ++  
Luis Joyanes Aguilar  
Mc graw hill

### **Circuitos lógicos**

Teoría y técnica  
A. Paz Huguet  
Edit. Cedel

### **Fundamentos de Informática**

Luís a. Ureña  
Antonio M. Sánchez Maria t. Martín José M. Mantas