

UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA EN COMPUTACIÓN



“DISEÑO DE UNA HERRAMIENTA CASE”

TRABAJO DE GRADUACIÓN
PARA OPTAR AL GRADO DE

INGENIERO EN CIENCIAS DE LA COMPUTACIÓN

PRESENTADO POR:

MARTÍNEZ ZELAYA, ANA LISSETTE
RAMÍREZ CHICAS, KARLA PATRICIA
VILLANUEVA URRUTIA, CATERINE BEATRIZ

CIUDADELA DON BOSCO

SEPTIEMBRE DE 2003

UNIVERSIDAD DON BOSCO

FACULTAD DE INGENIERÍA

AUTORIDADES

RECTOR:

ING. FEDERICO MIGUEL HUGUET RIVERA

SECRETARIO GENERAL:

HNO. MARIO OLMOS

DECANO DE LA FACULTAD DE INGENIERÍA:

ING. CARLOS BRAN

ASESOR DE TRABAJO DE GRADUACIÓN:

LIC. OSCAR ARNOLDO MÉNDEZ

JURADO EVALUADOR:

LIC. JORGE MAURICIO COTO

ING. RENÉ AMERICO HERNÁNDEZ MONTENEGRO

LIC. CARLOS ALBERTO MARTÍNEZ ZARAGOZA

UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA EN COMPUTACIÓN

JURADO EVALUADOR DEL TRABAJO DE GRADUACIÓN
“DISEÑO DE UNA HERRAMIENTA CASE”

Lic. Jorge Mauricio Coto

JURADO

Ing. René Américo Hernández

JURADO

Lic. Carlos Martínez Zaragoza

JURADO

Lic. Oscar Arnoldo Méndez

ASESOR

DEDICATORIA

- **A Dios Todopoderoso y la Santísima Virgen María,** por iluminarme en cada momento de mi vida y darme fortaleza para alcanzar esta meta.
- **A mis padres: Ana Elizabeth y José David,** por su amor, comprensión y sacrificio realizado durante el transcurso de la carrera y la realización del proyecto.
- **A mi Hermano: José David,** por su ayuda y apoyo durante el desarrollo de mi carrera.
- **A mis compañeras de Proyecto,** por su amistad, comprensión y esfuerzo en el desarrollo y culminación de esta meta.
- **A nuestro asesor: Lic. Oscar Méndez,** por guiarnos eficientemente para lograr nuestra meta.
- **A mi Familia y a todos mis Amigos,** que me animaron para seguir adelante, mis más sinceros agradecimientos.

Ana Lissette Martínez Zelaya

DEDICATORIA

A DIOS TODOPODEROSO y A LA VIRGEN MARIA:

Dándole gracias por haberme iluminado para alcanzar la meta propuesta.

A MIS PADRES

José Álvaro Ramírez Castro

Daysi Chicas Baires de Ramírez

Con profundo amor y gratitud por brindarme su apoyo en todo momento.

A MIS HERMANOS

Elba Ivett

Daysi Guadalupe

Jaime Enrique

Con mucho cariño, por su colaboración.

A MIS COMPAÑERAS

Caterine Beatriz Villanueva

Ana Lissette Martinez

Y DEMAS AMIGOS Y FAMILIA

Con respeto y agradecimiento por su decidida cooperación y estimulación en todo momento.

Karla Patricia Ramírez Chicas

DEDICATORIA

- **A mi DIOS Todopoderoso** por haberme iluminado y proporcionado la sabiduría necesaria para llevar a cabo mis objetivos y alcanzar esta nueva meta; porque sé que de no ser por su ayuda, ninguno de todos mis propósitos serían logrados y he estado siempre segura de que “todo lo puedo en Cristo que me fortalece”.
- **A mis queridos padres Mauricio y María Elena**, quienes se han encargado de formarme y educarme en el camino del bien, me ensañaron a ser responsable y a luchar por ser mejor cada día. Gracias por estar siempre a mi lado.
- **A mi querido esposo Nelson**, por su comprensión y apoyo. Gracias por confiar siempre en mi capacidad e impulsarme a seguir adelante.
- **A mis hermanos Mauricio y Wilfredo**, por su cariño y apoyo.
- **A mis compañeras de Tesis Lissette y Karla**, por todo el esfuerzo realizado para llevar a una excelente finalización el presente proyecto.
- **A nuestro asesor de Tesis Lic. Oscar Méndez**, por haber compartido sus valiosos conocimientos y experiencias profesionales para el desarrollo de este proyecto.
- **A Silvita**, por su cariño y colaboración en todas las etapas de mi carrera, porque en todas mis alegrías y tristezas ha estado a mi lado. Mil gracias Amiga.
- **A todos mis familiares y amigos**, por brindarme el apoyo necesario para seguir adelante a lo largo de mi carrera.

Caterine Beatriz Villanueva

INDICE

Introducción	1
Objetivos	2
Justificación	3
Alcances y Limitaciones	4
I. Generalidades	6
1.1. Definición del Tema	7
1.2. Marco Teórico	9
1.2.1 Marco Conceptual	9
II. Desarrollo de la Investigación	29
2.1 Metodología de la Investigación	30
2.1.1 Investigación Bibliográfica	30
2.1.2. Investigación de Campo	30
2.2 Técnicas de Investigación	30
2.2.1. Encuesta y Entrevista	30
2.2.2. Características y Definición de la Población	31
2.2.3. Cálculo de la Muestra	33
2.3 Resultados de la Investigación	35
2.3.1 Análisis de Resultados de Encuestas	35
2.3.2 Análisis de Resultados de Entrevistas	45
2.3.3 Análisis de la Situación Actual	49
2.3.4 Factibilidad de Implantación	49
2.3.4.1. Factibilidad Técnica	49
2.3.4.2. Factibilidad Económica	49
2.3.4.3. Factibilidad Operativa	50
III. Análisis y Diseño de la Herramienta CASE	51
3.1 Diagrama Entidad – Relación para el Diseño del Diccionario de Datos a ser utilizado por la Herramienta CASE: “Integral CASE”	52

3.2	Identificación de Procedimientos	65
3.2.1	Diseño de Salidas	65
3.2.2	Diseño de Entradas	65
3.2.3.	Diseño de Procesos	66
3.2.4.	Diagramas de Análisis	66
3.2.4.1	Diagrama de Jerarquía de Procesos para la Herramienta CASE “Integral CASE”	67
3.2.4.2	DFD de Procedimientos de la Herramienta CASE “Integral CASE”	69
IV.	Desarrollo del Sistema	73
4.1.	Diseño de Controles e Interfaces	74
4.1.1.	Creación de interfaz de usuario de la herramienta Integral CASE	74
4.1.1.1	Ventanas de Ingreso	74
4.1.1.2.	Opciones del Menú Principal	76
4.1.1.3.	Barra de Herramientas	81
4.1.1.4.	Árbol de Proyectos	81
4.1.1.5.	Barra de Controles	82
4.1.2.	Diseño y Creación de Clases Base	83
4.2.	Diseño de Algoritmos	86
4.2.1.	Diseño de Base de Datos para Diccionario de Datos	87
4.2.2.	Apertura de Proyectos	88
4.2.3.	Validación de Errores en el Diseño de Diagramas E-R.	89
4.2.4.	Generación automática de bases de datos.	90
4.2.5.	Conexión a Bases de Datos	92
4.2.6.	Diseño de pantallas de interfaz y mantenimiento	92
4.2.7.	Definición de Consultas	94
4.2.8.	Generación de Proyecto en Visual Basic	95

4.3. Opciones de documentación	100
V. Validación de Resultados	102
5.1. Procedimiento de Validación de Resultados	103
5.2. Análisis de resultados de encuestas	104
5.3. Resumen comparativo de resultados de validación	108
VI. Plan de implantación	109
6.1. Metodología de implantación	110
6.1.1. Plan de mercadeo	110
6.1.2. Plan de capacitación a usuarios	111
6.1.3. Plan de instalación	112
6.1.4. Plan de conversión	113
6.1.5. Revisión de resultados	113
6.2. Nuevo entorno de desarrollo de software	114
Conclusiones	116
Recomendaciones	118
Glosario Técnico	119
Bibliografía	126
ANEXOS	128
ANEXO A: Precios de Herramientas CASE	129
ANEXO B: Cálculo del tamaño de la muestra	130
ANEXO C: Formulario de Encuesta y Entrevista	134
ANEXO D: Lista de Empresas Encuestadas	140
ANEXO E: Manual de Usuario	141

INTRODUCCIÓN

En la actualidad existen muchas empresas de los diferentes sectores que poseen áreas de desarrollo de sistemas; dichos sistemas son elaborados ya sea para la venta a otras empresas o para la mecanización de diversas actividades internas. Es de vital importancia que los sistemas diseñados funcionen de la mejor manera, garantizando procesos eficientes y de calidad; para lograrlo, resulta conveniente apoyar la elaboración de éstos con herramientas que permitan incrementar la productividad y el control de calidad en el desarrollo.

El presente proyecto consiste en el diseño de una herramienta de Ingeniería de Software Asistida por Computadora (CASE), que automatice aspectos clave del diseño de sistemas en sus primeras etapas. El propósito primordial es proporcionar a los analistas y programadores de sistemas una alternativa de desarrollo, apoyando sus actividades con una herramienta accesible y completamente funcional de acuerdo a sus necesidades.

OBJETIVOS

OBJETIVO GENERAL

Diseñar una herramienta de Ingeniería de Software Asistida por Computadora (CASE), como apoyo a las fases de diseño y desarrollo del ciclo de vida de sistemas y a las metodologías de análisis estructurado.

OBJETIVOS ESPECÍFICOS

- Crear una herramienta que facilite al usuario la generación de diagramas Entidad-Relación (E-R) utilizando el modelo físico, con la capacidad de realizar verificaciones completas de errores en el diseño
- Proporcionar herramientas para la construcción de bases de datos en Microsoft Access, MySQL y Microsoft SQL Server; a partir de un diagrama E-R creado por el usuario.
- Incluir modelos de creación automática de interfaces para la fácil manipulación de la información y así lograr la creación de prototipos con rapidez y seguridad.
- Permitir la visualización de la información de la base de datos mediante la generación automática de reportes básicos.
- Mejorar la eficiencia y efectividad en el desarrollo de aplicaciones de bases de datos a través de la construcción de un adecuado diagrama E-R.

JUSTIFICACIÓN

En la actualidad se hace necesaria la utilización de herramientas que permitan incrementar la productividad y el control de calidad en los procesos de elaboración de sistemas; para ello existen herramientas en el mercado que contribuyen con la automatización del desarrollo, las cuales representan inversiones elevadas de adquisición para las empresas dedicadas al análisis y diseño de sistemas (**Ver Anexo A**).

De acuerdo a lo anterior y como apoyo a dicho sector, se realiza el diseño de una herramienta CASE que permita la creación de aplicaciones fundamentales para la gestión de información por medio de bases de datos relacionales, constituyendo el inicio de la construcción del resto de la aplicación.

ALCANCES Y LIMITACIONES

ALCANCES

- La aplicación será capaz de generar código en SQL para la creación de Bases de Datos y en Microsoft Visual Basic para el diseño automático de interfaces de usuario y conexiones ODBC.
- La herramienta CASE a diseñar será capaz de realizar una verificación de errores en el diseño de bases de datos, mediante la evaluación del diagrama E-R creado por el usuario
- Será posible el almacenamiento de diagramas E-R y la recuperación de los mismos para posteriores modificaciones
- La generación de bases de datos solamente será posible si el diagrama E-R creado por el usuario se encuentra libre de errores en el diseño.
- Proporcionar herramientas de diseño de sistemas que permitan definir el tipo de pantallas de mantenimiento (describiendo la posición de datos, mensajes y encabezados de dichas pantallas) que el usuario incluirá en el prototipo, así como también la creación rápida de menús.
- Proporcionar al desarrollador únicamente modelos de interfaz ya establecidos, con la finalidad de elegir entre éstos el tipo de diseño más conveniente a su aplicación.
- Las operaciones de mantenimiento se efectuarán respetando la integridad referencial en tablas relacionadas.

- La información será visualizada a partir de reportes de cada una de las tablas de la base de datos.

LIMITACIONES

- El sistema estará orientado para ser utilizado sólo por personas que posean al menos conocimientos básicos de análisis y diseño de sistemas.
- Para que el Sistema de Automatización de Software funcione en su totalidad, los usuarios deben poseer MySQL o Microsoft SQL Server; como gestor de bases de datos relacional (en caso de generar diagramas de estos tipos) y licencia de Microsoft Visual Basic como herramienta de desarrollo.
- Los formularios incluidos en el prototipo solamente permitirán operaciones básicas de mantenimiento a tablas.
- No se generarán reportes de información cruzada a partir de tablas relacionadas.
- La Herramienta CASE podrá ser utilizada solamente sobre plataforma Windows

CAPITULO I

GENERALIDADES

1.1. DEFINICIÓN DEL TEMA

“Diseño de una herramienta (CASE)”, denominada: **Integral CASE**

Integral CASE contiene plantillas para la creación de diagramas Entidad-Relación (E-R) utilizando el modelo físico. A partir del diagrama E-R creado por el usuario, la herramienta es capaz de generar automáticamente la base de datos correspondiente (realizando previamente una verificación de errores), proporcionando al usuario la opción de utilizar como gestor de bases de datos uno de los siguientes: Microsoft Access, MySQL y Microsoft SQL Server.

Una vez realizado el diseño de la base de datos el usuario puede crear un prototipo inicial de una aplicación en Microsoft Visual Basic, la que permitirá manipular la información de dicha base de datos; esto se logrará a través del diseño automático de la interfaz gráfica, utilizando herramientas de diseño de formularios; el código para conexión a la base de datos y código para los botones de mantenimiento a tablas (adición, modificación, consultas y eliminación de registros), permitiendo la visualización de los datos en reportes básicos.

El entorno de trabajo de Integral CASE posee una interfaz amigable al usuario, desarrollando todas las operaciones en idioma español.

La herramienta CASE hará uso de un **diccionario de datos (Ver Pág. 14)** para almacenar las características de las entidades, atributos, reglas de validación y relaciones entre entidades, que definen la estructura de la base de datos (Meta - Base de Datos), con el fin

de facilitar el control en la creación y edición del diagrama y de la base de datos generada. Además en dicho diccionario se almacena información sobre la distribución de objetos en las pantallas de mantenimiento a datos, lo que permitirá posteriores modificaciones en el diseño.

1.2. MARCO TEÓRICO

1.2.1. MARCO CONCEPTUAL

- **HERRAMIENTA**

Es cualquier dispositivo que, cuando se emplea en forma apropiada, mejora el desempeño de una tarea.

Importancia de las herramientas en el desarrollo de Sistemas

Las herramientas mejoran la forma en que ocurre el desarrollo y tienen influencia sobre la calidad del resultado final; extienden en tres formas la capacidad del analista de sistemas: proporcionan el potencial para mejorar la productividad del analista, facilitan el desarrollo de procesos más eficaces y mejoran la calidad del sistema. En otras palabras, tanto el proceso de desarrollo de sistemas como el producto que se obtiene con él (Fig. 1.2.1.1), pueden mejorarse con el uso de herramientas apropiadas.

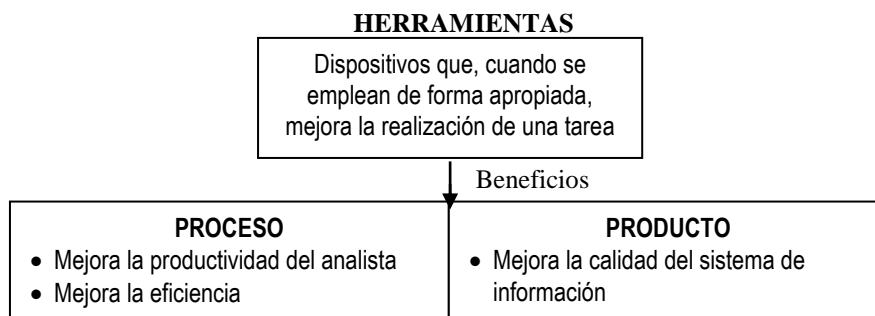


Figura 1.2.1.1
Beneficios obtenidos con el uso de herramientas en el desarrollo de sistemas de información

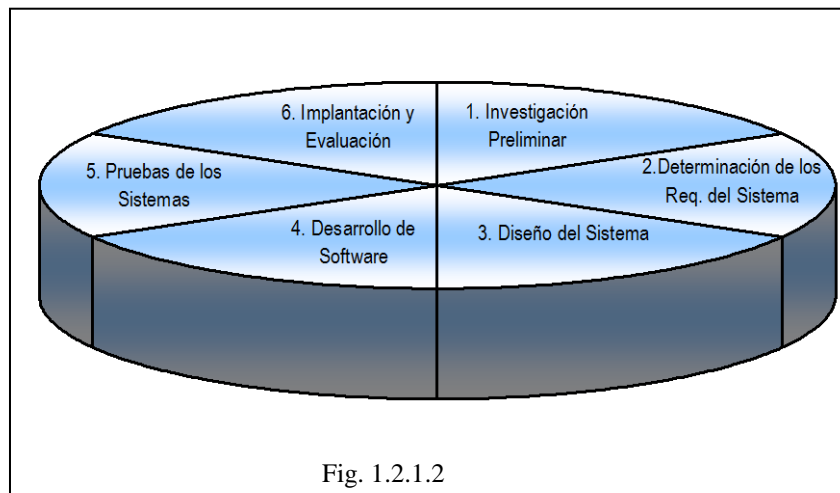
- **CICLO DE VIDA DE UN SISTEMA**

El desarrollo de sistemas, un proceso formado por las etapas de análisis y diseño, comienza cuando la administración o algunos miembros del personal encargado de desarrollar sistemas, detectan un sistema de la empresa que necesita mejoras.

El método del ciclo de vida para desarrollo de sistemas (Fig. 1.2.1.2) es el conjunto de actividades que los analistas, diseñadores y usuarios realizan para desarrollar e implantar un sistema de información.

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

a) Investigación preliminar, b) Determinación de los requerimientos del sistema, c) Diseño del sistema, d) Desarrollo de software, e) Prueba de los sistemas, f) Implantación y evaluación



Actividades del ciclo de vida clásico de desarrollo de sistemas

- **DIAGRAMA DE FLUJO DE DATOS**

Herramienta gráfica que se emplea para describir y analizar el movimiento de datos a través de un sistema, ya sea que éste fuera manual o automatizado incluyendo procesos, lugares para almacenar datos y retrasos en el sistema. Los diagramas de flujo de datos son la herramienta más importante y la base sobre la cual se desarrollan otros componentes. La transformación de datos de entrada en salida por medio de procesos puede describirse en forma lógica e independiente de los componentes físicos asociados con el sistema.

Notación:

Los diagramas lógicos de flujo de datos se pueden dibujar con sólo cuatro notaciones sencillas, es decir con símbolos especiales o iconos y anotaciones que los asocian con un sistema específico. El uso de íconos especiales para cada elemento depende de qué enfoque se utilice, el de **Yourdon** o el de **Gane y Sarson**.

- 1. Flujo de Datos.** Movimiento de datos en determinada dirección desde un origen hacia un destino en forma de documentos, cartas, llamadas telefónicas o virtualmente por cualquier otro medio. El flujo de datos es un “paquete” de datos.

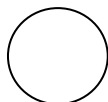


Yourdon

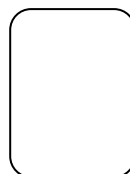


Gane y Sarson

- 2. Procesos.** Personas, procedimientos o dispositivos que utilizan o producen (transforman) datos. No se identifica el componente físico.



Yourdon



Gane y Sarson

- 3. Fuente o Destino de los Datos.** Fuentes o destinos externos de datos que pueden ser personas, programas, organizaciones u otras entidades que interactúan con el sistema pero se encuentran fuera de su frontera.

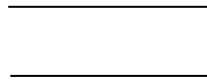


Yourdon

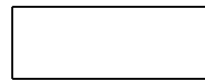


Gane y Sarson

4. Almacenamiento de datos. Es el lugar donde se guardan los datos o al que hacen referencia los procesos en el sistema. El almacenamiento de datos puede representar dispositivos tanto computarizados como no computarizados.



Yourdon



Gane y Sarson

- **DIAGRAMAS ENTIDAD – RELACIÓN**

El modelo de datos entidad-relación (**E-R**) se basa en una percepción de un mundo real que consiste en un conjunto de objetos básicos llamados entidades y relaciones entre estos. Se desarrolló para facilitar el diseño de bases de datos.

Tipos de Diagramas E-R

Existen dos tipos de representación de Diagramas E-R: El modelo físico de datos (PDM), que representa la estructura física de la base de datos, y el modelo conceptual de datos (CDM), que representa la estructura lógica.

A continuación se muestra una tabla comparativa entre PDM y CDM:

PDM : Physical Data Model	CDM : Conceptual Data Model
<p>Definición:</p> <p>El PDM considera los detalles de aplicación real física. Toma en cuenta en qué sistema se creará la base de datos. El usuario puede modificar el PDM desarrollado.</p>	<p>Definición:</p> <p>Cuando diseña una BDD el proceso de diseño normalmente comienza en el nivel conceptual.</p> <p>En el nivel conceptual, no es necesario considerar los detalles de la implementación física actual.</p>

	<p>Un CDM representa la estructura global lógica de una base de datos, que es independiente de cualquier software o la estructura de base de datos almacenados. Un modelo conceptual a menudo contiene datos de los objetos, pero no siempre implementados en la base de datos física. Lo anterior proporciona una representación formal de los datos necesarios para utilizarlos en una empresa o una actividad de negocios.</p>
<p>Roles:</p> <ul style="list-style-type: none"> - Representa la organización física de datos en un formato gráfico - Genera creación de bases de datos y modificación de scripts. - Define integridad referencial activa, triggers y constraints. - Ingeniería inversa de bases de datos existentes. 	<p>Roles:</p> <ul style="list-style-type: none"> - Representa la organización de los datos en un formato gráfico - Verifica la validez del diseño de los datos - Genera un Modelo de Datos físico (PDM), que especifica la aplicación física de la base de datos
<p>Objetos:</p> <p><i>Tabla:</i> Colección de filas (archivos) y columnas (campos).</p> <p><i>Columna:</i> Estructura de datos que contiene un artículo de los datos individual dentro de una fila (registro), modelo equivalente de un campo de la bases de datos.</p> <p><i>Llave:</i> Columna o columnas cuyos valores únicos identifican cada fila en una tabla. Una llave se puede designar como una llave primaria o alternativa.</p> <p><i>Llave primaria:</i> Columna o columnas cuyos valores únicos identifican cada fila en una tabla, y se designa como identificador primario de cada fila en una tabla.</p> <p><i>Llave Alternativa:</i> Columna o columnas</p>	<p>Objetos:</p> <p><i>Dominio:</i> Fija valores para los cuales los registros de datos son válidos.</p> <p><i>Registro de datos:</i> Pieza elemental de información</p> <p><i>Entidad:</i> Persona, lugar, cosa, o concepto que tiene características de interés a la empresa y sobre el cual puede guardar información.</p> <p><i>Atributo de la entidad:</i> Pieza elemental de información conectada a una entidad</p> <p><i>Enlace inherente:</i> Relación especial que define una entidad con un caso especial de una asociación entre entidades</p>

cuyos valores únicos identifican cada fila en una tabla, y que no es una llave primaria.

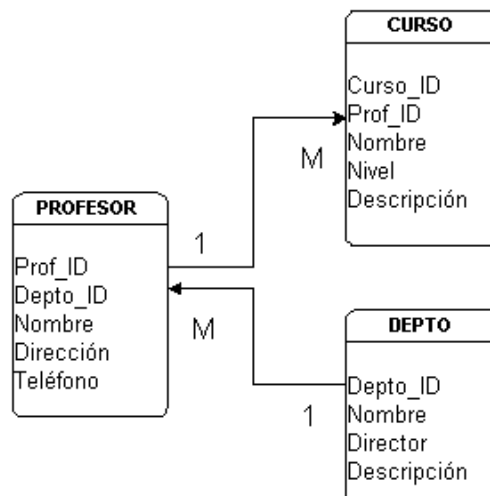
Llave Foránea: Columna o columnas cuyos valores dependen y emigran de una llave primaria o alterna en otra tabla.

Índice: Estructura de datos asociados con una o más columnas en una tabla, en el que el valor de la columna es ordenado de tal manera que permita un acceso rápido a los datos.

Relaciones: Enlace entre la llave primaria o una alterna en una tabla padre, y una llave foránea de una tabla hija. Dependiendo de las propiedades seleccionadas, una referencia también puede enlazar columnas que son independientes de columnas de llaves primarias o alternas.

Una base de datos que se ajusta a un diagrama E-R puede representarse por medio de una colección de tablas. Para cada conjunto de entidades y para cada conjunto de relaciones en la base de datos, existe una tabla única a la que se le asigna el nombre del conjunto correspondiente. Cada tabla tiene un número de columnas, que tienen nombres únicos.

Ejemplo:



- **DICCIONARIO DE DATOS**

El diccionario de datos contiene las características de las entidades y atributos, que definen la estructura de la Base de Datos (Meta - Base de Datos). El objetivo del diccionario de datos es:

- Facilitar el control de cada una de las entidades y atributos que forman parte de la estructura de Base de Datos del Sistema.
- Controlar dinámicamente la estructura de la interfaz al usuario, para las diferentes pantallas del sistema.

Ventajas de utilizar un Diccionario de Datos

El uso de un diccionario de datos brinda las siguientes ventajas:

- La posibilidad de generar automáticamente todas las tablas definidas para el sistema.
- El permitir que el sistema reconfigure dinámicamente la interfaz al usuario, con respecto a cómo y cuales son los atributos que se deben presentar en las diferentes pantallas.

- **HERRAMIENTA CASE**

Las siglas **CASE** significan: Ingeniería de Software Asistida por Computadora (Computer Aided Software Engineering)

- **COMPONENTES DE UNA HERRAMIENTA CASE**

En general, las herramientas tipo CASE incluyen los siguientes cinco componentes:

- a) **Herramientas para diagramación:** dan soporte al análisis y documentación de los requerimientos de una aplicación. Por lo general, incluyen facilidades para producir diagramas de flujo de datos, las cuales son esenciales para brindar apoyo a la metodología de análisis estructurado.

En este componente puede definirse que la herramienta CASE a desarrollar, implicará la creación del Diagrama Entidad- Relación utilizando el modelo físico de datos (PDM) ya que éste especifica la aplicación física que permite la construcción de la base de datos; con este modelo a partir de la representación gráfica se consideran los detalles de una aplicación física real.

- b) **Depósito centralizado de información:** la captura, análisis, procesamiento y distribución de todos los sistemas de información es asistida por un *depósito de información* centralizado o diccionario de datos, el cual contiene detalles sobre los componentes del sistema, tales como datos, flujo de datos y procesos; asimismo, también incluye información que describe el volumen y frecuencia de cada una de las actividades.

- c) **Generador de interfaces:** las interfaces con el sistema son los medios que permiten a los usuarios interactuar con una aplicación, ya sea para dar entrada a información y datos o para recibir información. Los *generadores de interfaces*

ofrecen la capacidad para preparar imitaciones y prototipos para las interfaces con los usuarios. Por lo general, soportan la rápida creación de menús de demostración para el sistema, de pantallas de presentación y del formato de los informes.

- d) Generadores de código:** los generadores de código automatizan la preparación de software. Éstos incorporan métodos que permiten convertir las especificaciones del sistema en código ejecutable.

Los generadores de código producen un porcentaje del código fuente de la aplicación. El resto debe ser escrito por los programadores. La codificación manual, que es el nombre que recibe este proceso, sigue siendo necesaria.

- e) Herramientas de administración:** los sistemas CASE también ayudan a los gerentes de proyecto a mantener la efectividad y eficiencia de todo el proceso de desarrollo de una aplicación. Este componente de CASE ayuda a los gerentes de desarrollo a calendarizar las actividades de análisis y diseño así como la asignación de recursos a las diferentes actividades del proyecto.

- **TIPOS DE CASE**

No existe una única clasificación de herramientas CASE y, en ocasiones, es difícil incluirlas en una clase determinada. Podrían clasificarse atendiendo a:

- Las plataformas que soportan.

- Las fases del ciclo de vida del desarrollo de sistemas que cubren.
- La arquitectura de las aplicaciones que producen.
- Su funcionalidad.

Las herramientas CASE, en función de las fases del ciclo de vida abarcadas, se pueden agrupar de la forma siguiente:

a) Herramientas integradas, I-CASE (Integrated CASE, CASE integrado):

abarcan todas las fases del ciclo de vida del desarrollo de sistemas. Son llamadas también CASE workbench.

b) Herramienta(s) que comprende(n) alguna(s) fase(s) del ciclo de vida de desarrollo de software:

- **Herramientas de alto nivel, U-CASE** (Upper CASE - CASE superior) o front-end, orientadas a la automatización de las primeras actividades del proceso de desarrollo. Proporcionan soporte para el desarrollo de modelos gráficos de sistemas y procesos.
- **Herramientas de bajo nivel, L-CASE** (Lower CASE - CASE inferior) o back-end, dirigidas a las últimas fases del desarrollo: construcción e implantación.

- **Juegos de herramientas o toolkits**, son el tipo más simple de herramientas CASE. Automatizan una fase dentro del ciclo de vida. Dentro de este grupo se encuentran las herramientas de reingeniería, orientadas a la fase de mantenimiento.

La siguiente gráfica muestra varios tipos de herramientas, tanto de tipo front-end (Alto Nivel) como de tipo back-end (Bajo Nivel):

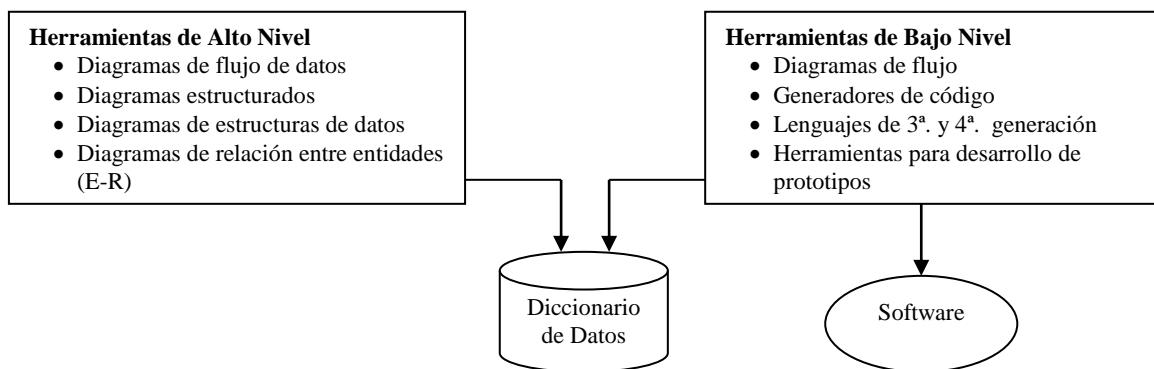


Figura 1.2.1.3
Ejemplos de Tipos de Herramientas

Otra posible clasificación, utilizando la funcionalidad como criterio principal, es la siguiente:

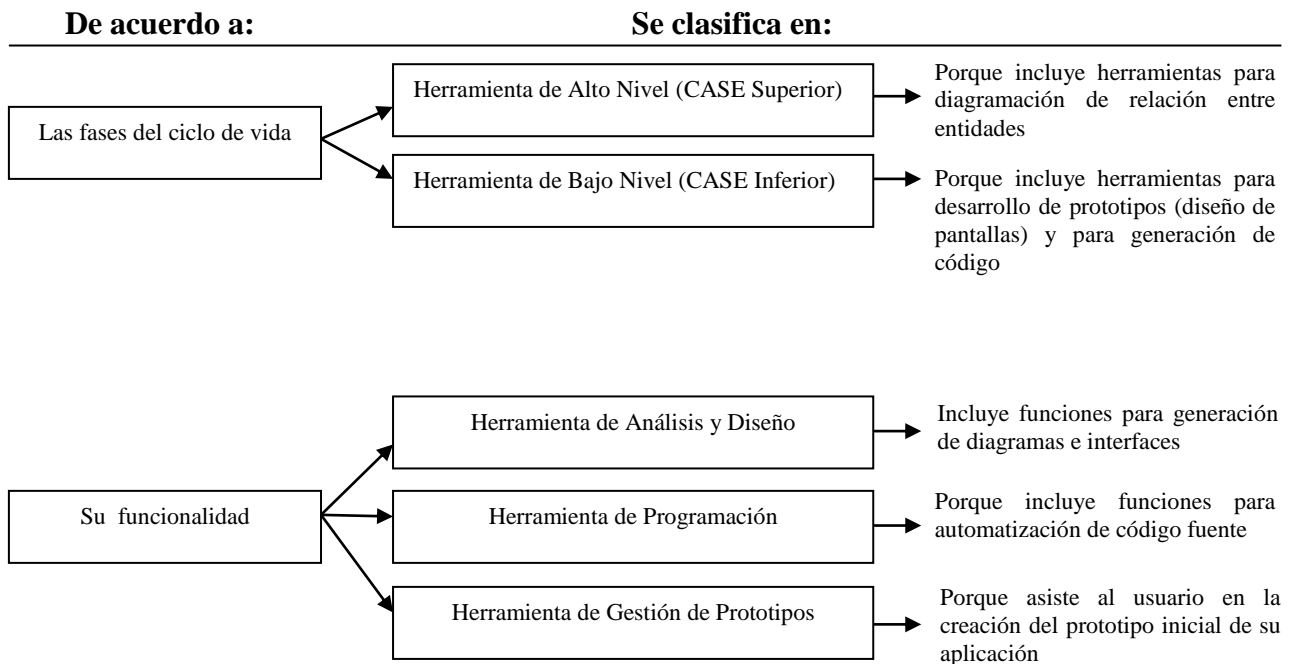
- Herramientas de planificación de sistemas de gestión:** Sirven para modelizar los requisitos de información estratégica de una organización. Su objetivo principal es ayudar a comprender mejor cómo se mueve la información entre las distintas unidades organizativas. Proporcionan una ayuda importante cuando se diseñan nuevas estrategias para los sistemas de información y cuando los métodos y sistemas actuales no satisfacen las necesidades de la organización.

- b) Herramientas de análisis y diseño:** Permiten al desarrollador crear un modelo del sistema que se va a construir y también la evaluación de la validez y consistencia de este modelo. Proporcionan un grado de confianza en la representación del análisis y ayudan a eliminar errores con anticipación. Se tienen:
- Herramientas de análisis y diseño (Modelamiento), Herramientas de creación de prototipos y de simulación, Herramientas para el diseño y desarrollo de interfaces, Máquinas de análisis y diseño (Modelamiento).
- c) Herramientas de programación:** Se engloban aquí los compiladores, los editores y los depuradores de los lenguajes de programación convencionales. Ejemplos de estas herramientas son: Herramientas de codificación convencionales, Herramientas de codificación de cuarta generación, Herramientas de programación orientadas a los objetos.
- d) Herramientas de integración y prueba:** Sirven de ayuda a la adquisición, medición, simulación y prueba de los equipos lógicos desarrollados. Entre las más utilizadas están: herramientas de análisis estático, herramientas de codificación de cuarta generación, herramientas de programación orientadas a los objetos.
- e) Herramientas de gestión de prototipos:** Los prototipos son utilizados ampliamente en el desarrollo de aplicaciones, para la evaluación de especificaciones de un sistema de información, o para un mejor entendimiento de cómo los requisitos de un sistema de información se ajustan a los objetivos perseguidos.

f) **Herramientas de mantenimiento:** La categoría de herramientas de mantenimiento se puede subdividir en: herramientas de ingeniería inversa, herramientas de reestructuración y análisis de código, herramientas de reingeniería, herramientas de gestión de proyectos.

- **CLASIFICACIÓN DE LA HERRAMIENTA CASE DISEÑADA: Integral CASE**

De acuerdo a los tipos de herramientas mostrados en el apartado anterior, la herramienta CASE **Integral CASE** se clasifica de la siguiente manera:



- **BENEFICIOS Y DEBILIDADES DE UNA HERRAMIENTA CASE**

BENEFICIOS

Entre los beneficios ofrecidos por la tecnología CASE se encuentran los siguientes:

- Facilidad para llevar a cabo la tarea de revisión de especificaciones del sistema así como de representaciones gráficas.
- Facilidad para desarrollar prototipos de sistemas por medio de la capacidad para cambiar especificaciones y, por otro lado, para determinar el efecto que sobre el desempeño del sistema tendrían otras alternativas.
- Generación de código
- Soporte para mantenimiento como resultado de haber guardado las especificaciones del sistema en un depósito central de información.
- Aumentar las posibilidades de satisfacer los requerimientos del usuario.

DEBILIDADES

Las herramientas CASE tienen puntos débiles significativos, que van desde la confiabilidad en los métodos estructurados hasta su alcance limitado, los cuales amenazan con minar los beneficios potenciales descritos con anterioridad.

- **Confiabilidad en los métodos estructurados:** Muchas herramientas CASE están construidas teniendo como base las metodologías del análisis estructurado y del ciclo de vida de desarrollo de sistemas. Por si sola, esta característica puede convertirse en la principal limitante ya que no todas las organizaciones emplean métodos de análisis estructurado.
- **Conflictos en el uso de los diagramas:** Las herramientas difieren en el uso que hacen de los diagramas. Algunas son herramientas exclusivamente para gráficas, que se abocan al dibujo de diagramas para el análisis de entrada y salida de datos. Este tipo de herramienta pueden restringir ya sea el proceso de desarrollo normal seguido por una organización o el estilo particular de trabajo de los analistas.
- **Diagramas no utilizados:** Las herramientas con representaciones gráficas y documentación mejoran la productividad de desarrollo de software. Sin embargo, en algunos casos, las herramientas gráficas, automatizadas o manuales, no se emplean del todo o tal vez no se utilicen en la forma en que deberían emplearse.
- **Función limitada:** Aunque una herramienta puede apoyar varias fases del ciclo de vida de desarrollo de sistemas o adaptarse a diferentes metodologías de desarrollo, por lo general su enfoque primario esta dirigido hacia una fase o método específico.
- **Alcance limitado:** Aunque muchas herramientas basadas en computadora incluyen la capacidad de verificar las especificaciones para determinar su consistencia, virtualmente no llevan a cabo ningún análisis de los requerimientos de la aplicación.

Por tanto, el alcance de las actividades de desarrollo asociado con las herramientas existentes es bastante limitado.

- **Las tareas humanas siguen siendo críticas:** La tecnología CASE ofrece herramientas que soportan las funciones de modelado, verificación, manejo de datos y de utilería que son necesarias para mejorar la productividad del desarrollo. Sin embargo, las herramientas deben estar en manos de personas con experiencia y deben adaptarse a la arquitectura de la información así como a las metodologías de desarrollo utilizadas por la organización.

- **HERRAMIENTAS DE DESARROLLO Y GESTORES DE BASE DE DATOS**

Lenguaje de Programación: Visual Basic

¿Qué es Visual Basic? La palabra "Visual" hace referencia al método que se utiliza para crear la interfaz gráfica de usuario (GUI). En lugar de escribir numerosas líneas de código para describir la apariencia y la ubicación de los elementos de la interfaz, simplemente puede agregar objetos prefabricados en su lugar dentro de la pantalla.

La palabra "Basic" hace referencia al lenguaje BASIC (Beginners All-Purpose Symbolic Instruction Code).

Visual Basic ha evolucionado a partir del lenguaje BASIC original y ahora contiene centenares de instrucciones, funciones y palabras clave, muchas de las cuales están directamente relacionadas con la interfaz gráfica de Windows.

Características Fundamentales de Visual Basic:

- Posibilidades de acceso a datos para creación de bases de datos, aplicaciones cliente y componentes de servidor.
- Tecnología ActiveX.
- Programación en Internet, proporciona técnicas de programación necesarias para trabajar con servidores Web, protocolos, y aplicaciones dynamic HTML.
- Creación de archivos .EXE
- Programación orientada a objetos.

Gestor de Base De Datos: Microsoft Access

Por medio de Microsoft Access, se puede administrar toda la información desde un único *archivo de base de datos*.

Dentro del *archivo*, se puede dividir los datos en contenedores de almacenamiento independientes denominados *tablas*; además se puede ver, agregar y actualizar datos de la tabla por medio de *formularios en pantalla*; se puede asimismo buscar y recuperar sólo los datos que desee por medio de *consultas*; y es posible analizar o imprimir datos con un diseño específico por medio de *informes*.

También permite a otros usuarios ver, actualizar o analizar datos de la base de datos desde Internet o desde una intranet por medio de páginas de acceso a datos.

Utiliza el motor de base de datos Microsoft Jet, empleando los objetos de acceso a datos de Microsoft (DAO, Data Access Objects) para tener acceso y manipular datos.

Gestor de Base De Datos: Microsoft SQL Server

SQL Server contiene muchas innovaciones en su interfaz y facilidad de uso, y al mismo tiempo incorpora herramientas sofisticadas para usuarios de nivel avanzado. Microsoft SQL Server proporciona soporte para un conjunto de características que aportan las siguientes ventajas:

- Facilidad de instalación, distribución y utilización: incluye un conjunto de herramientas administrativas y de desarrollo que mejoran la capacidad para instalar, distribuir, administrar y utilizar SQL Server entre varios sitios.
- Escalabilidad: puede utilizarse el mismo motor de base de datos a través de plataformas que van desde equipos portátiles que ejecutan Microsoft Windows 95 ó 98 hasta grandes servidores con varios procesadores que ejecutan Microsoft Windows NT.
- Almacenamiento de datos: incluye herramientas para extraer y analizar datos resumidos para el proceso analítico en línea (OLAP, *Online Analytical Processing*). SQL Server incluye también herramientas para diseñar gráficamente las bases de datos y analizar los datos mediante preguntas en lenguaje normal.
- Integración del sistema con otro software de servidor
- SQL Server se integra con el correo electrónico, Internet y Windows.

Gestor de Base De Datos: MySQL

MySQL es un sistema de administración de bases de datos muy potente. La principal virtud es que es totalmente gratuito y se distribuye bajo licencia GNU (General Public License), por lo que es una fuerte alternativa ante sistemas como SQL u Oracle. Muchas empresas han decidido comenzar a utilizar MySQL como gestor de bases de datos para sus proyectos, y se han dado cuenta de que es realmente muy poderoso, y no tiene nada que envidiarle a sus competidores.

Las principales ventajas que posee MySQL son:

- Trabaja en múltiples plataformas

- Existen versiones para Windows y Linux
- Si no se tiene el dinero para comprar un SQL Server para utilizar con las aplicaciones ASP (Active Server Pages), MySQL es una buena opción, ya que es un producto gratuito y de excelente calidad.
- También se puede utilizar en Linux con aplicaciones en PHP.
- MySQL es rápido, confiable y si se utilizan herramientas de administración visuales, es muy fácil de usar.

CAPITULO II

DESARROLLO DE LA INVESTIGACIÓN

2.1 METODOLOGÍA DE LA INVESTIGACIÓN

2.1.1 INVESTIGACIÓN BIBLIOGRÁFICA

Para conocer el impacto, beneficios y debilidades que se obtienen con la utilización de herramientas CASE tanto a nivel nacional como internacional, se han consultado diversas fuentes de información tales como: trabajos de investigación realizados con anterioridad, libros de análisis y diseño de sistemas y artículos publicados en Internet.

2.1.2. INVESTIGACIÓN DE CAMPO

Se ha realizado una investigación utilizando técnicas como entrevistas y encuestas, dirigidas a una muestra de población definida, a partir de las empresas de San Salvador que poseen áreas de desarrollo de software.

2.2 TÉCNICAS DE INVESTIGACIÓN

2.2.1. ENCUESTA Y ENTREVISTA

La encuesta: Se ha aplicado como instrumento para recopilar información, con el objetivo de conocer la utilización de diversas metodologías de análisis y diseño de sistemas. Se creó una encuesta (**Ver Anexo C**) dirigida a personas dedicadas al análisis y diseño, la cual contiene preguntas que reflejan los siguientes datos:

1. Métodos que utilizan los analistas y programadores para realizar sus tareas
2. Cantidad de analistas y programadores que utilizan herramientas CASE para el análisis y diseño.

3. Componentes que constituyen las herramientas CASE utilizadas por analistas y programadores

Los resultados anteriores permiten determinar la viabilidad de implantación de la herramienta CASE en diseño, así como la aceptación que ésta tendrá en nuestro medio.

La entrevista: con la entrevista diseñada (**Ver Anexo C**) se determinó los beneficios que obtienen las empresas en el área de desarrollo de sistemas, al emplear herramientas de automatización de procesos. A través de éste instrumento, la personas entrevistadas definieron sus conocimientos sobre herramientas CASE, los beneficios que se obtienen con su implantación, las debilidades observadas, las deficiencias de las metodologías tradicionales, y los costos que conlleva su implantación.

2.2.2. CARACTERÍSTICAS Y DEFINICIÓN DE LA POBLACIÓN

Se definieron las siguientes características para los elementos que constituyen la población:

- a) Empresas pertenecientes a los sectores: **Industria, Comercio, Servicios y Construcción.**
- b) Pertenecientes al sector público o privado
- c) Que posean un área destinada al desarrollo de sistemas informáticos
- d) Ubicadas en el Área Metropolitana de San Salvador; delimitando así el área geográfica de aplicación de las encuestas y entrevistas, tomando en cuenta que es en ésta área donde se concentra el mayor número de empresas del país que cumplen las características antes mencionadas

Para definir la población se tomó como base un estudio realizado en la tesis “Definición del Rol Profesional del Gestor de Información en El Salvador”, en la cual se distribuyeron encuestas a una muestra definida de empresas en los diferentes rubros (Industria, Servicios, Construcción y Comercio).

Total de empresas por rubros y empresas encuestadas

Tesis “Definición del Rol Profesional del Gestor de Información en El Salvador”

Sector	Total existentes	Porcentaje	Empresas Encuestadas
Industria	852	43.16%	31
Servicios	376	19.05%	14
Construcción	262	13.27%	10
Comercio	484	24.52%	18
TOTALES	1974	100%	73

En dichas encuestas se ve reflejada la cantidad de instituciones que desarrollan sistemas informáticos, a través de la pregunta:

¿Existe en su empresa alguna área destinada al diseño de software?

A la cual el **46.58%** respondió que sí

Dicho porcentaje se obtiene de las siguientes cifras:

Empresas que poseen área de desarrollo de software

Sector	Total	Porcentaje del total encuestadas
Industria	12	38%
Servicios	12	85%
Construcción	3	33%
Comercio	7	39%
TOTAL	34	

2.2.3. CÁLCULO DE LA MUESTRA

- Variables utilizadas para definición de la muestra

Las encuestas tienen como objetivo conocer las técnicas y herramientas que utilizan los analistas y programadores para el desarrollo de sistemas, así como el nivel de utilización de herramientas CASE en nuestro medio. Para la distribución de dichas encuestas, se definió la muestra tomando como Universo a la cantidad de empresas que poseen área de desarrollo de software: **34 empresas**

Cálculo del tamaño de la muestra (Ver Anexo B)

Fórmula:

$$n = \frac{NZ^2P(1-P)}{(N-1)e^2 + Z^2P(1-P)}$$

Donde:

N = tamaño de la población

P = proporción

Z = nivel de confianza a elegir

e = error máximo

n = tamaño de la muestra

- **Calculando muestra por sectores:**

a) **Industria**

N=12

P=0.38

Z=1.96 con 95% de confianza

e= 0.12

$$n = \frac{12 * 1.96^2 * 0.38 * (1 - 0.38)}{(12 - 1) * 0.12^2 + 1.96^2 * 0.38 * (1 - 0.38)}$$

n = 6.65

n ≈ 7.0

b) **Servicios**

N=12

P=0.85

Z=1.96 con 95% de confianza

e= 0.12

n = 9.06

n ≈ 9.0

c) **Construcción**

N=3

P=0.33

Z=1.96 con 95% de confianza

e= 0.12

n = 2.9

n ≈ 3.0

d) **Comercio**

N=7

P=0.39

Z=1.96 con 95% de confianza

e= 0.12

n = 6.3

n ≈ 6.0

- **Empresas a encuestar por sectores:**

Sector	Empresas que desarrollan (%)	Empresas a Encuestar
Industria	38	7
Servicios	85	9
Construcción	33	3
Comercio	39	6
TOTAL		25

2.3 RESULTADOS DE LA INVESTIGACIÓN

2.3.1 ANALISIS DE RESULTADOS DE ENCUESTAS

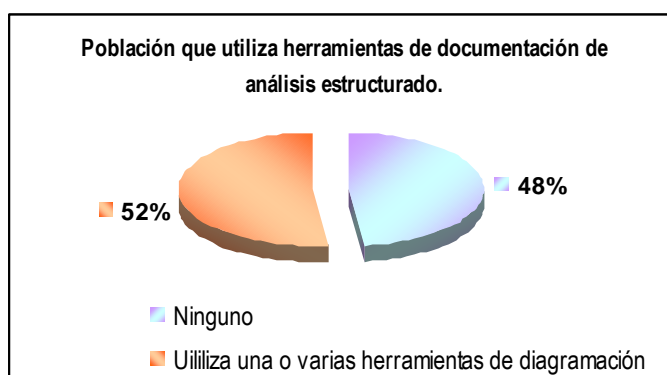
De las encuestas distribuidas a la muestra poblacional (**Ver Anexo D**), se han obtenido los siguientes resultados por cada una de las interrogantes planteadas.

Pregunta No 1

¿Cuáles herramientas de documentación de análisis estructurado emplea para el desarrollo de sus aplicaciones?

Respuesta:

ALTERNATIVAS	CANTIDAD
Diagrama de Flujo de Datos (DFD)	6
Flujogramas	2
Diagramas de Transición de Estados (DTE)	1
Diagrama Entidad – Relación	7
Ninguno	12
Otros	1



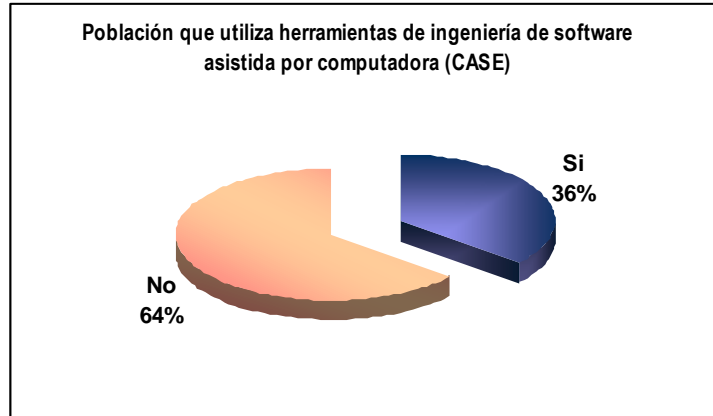
El 48% de los entrevistados no utilizan ninguna metodología de análisis estructurado, mientras que el 52% utilizan por lo menos una herramienta de documentación.

Pregunta No 2

¿Utiliza alguna herramienta de ingeniería de software asistida por computadora (CASE)?

Respuesta:

RESPUESTA	EMPRESAS	%
Si	9	36%
No	16	64%



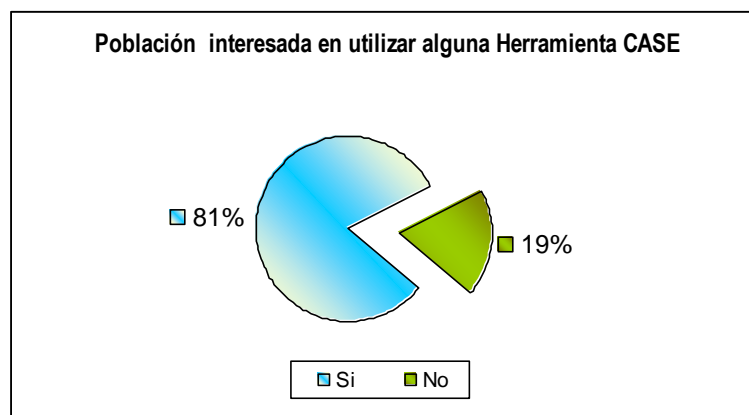
Del total de la población encuestada; el 36% utiliza alguna herramienta de Ingeniería de software asistida por computadora (CASE), mientras que el 64% no utilizan ninguna.

Pregunta No 3

Si actualmente NO utiliza una herramienta CASE para el desarrollo de software ¿Estaría interesado en utilizar una? ¿Porque?

Respuesta:

RESPUESTA	EMPRESAS
Si	13
No	3



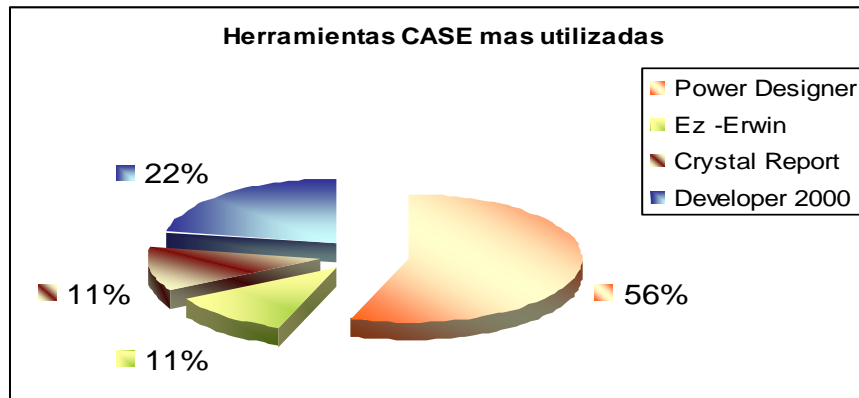
De la muestra de la población que **NO utiliza** una herramienta CASE el 81.25% estaría interesado en utilizar alguna, ya que le facilitaría el trabajo de sus aplicaciones, mientras el 18.75% no están interesados.

Pregunta No 4

¿Cuál es el nombre de la herramienta CASE que utiliza?

Respuesta:

RESPUESTA	Población
Power Designer	5
Ez -Erwin	1
Crystal Report	1
Developer 2000	2



Entre las herramientas CASE más utilizadas se encuentran PowerDesigner en mayor proporción, seguido por Developer 2000 , Ez-Erwin y Crystal Report.

Pregunta No 5

Indique si la herramienta CASE que utiliza posee los siguientes componentes:

Respuesta:

RESPUESTA	PowerDesigner	Ez-Erwin	Crystal Report	Developer 2000
Herramientas para Diagramación	x	x	x	X
Diccionario de Datos (Metadatos)	x	x		
Generador de Base de Datos	x	x		
Generador de Interfaces				X
Generador de código	x	x		X
Herramientas de Administración	x			X

Según los resultados se observa que PowerDesigner y Ez-Erwin son herramientas de Alto Nivel (Upper- CASE) orientadas a la automatización de las primeras actividades del proceso de desarrollo. Proporcionando soporte para el desarrollo de modelos gráficos de sistemas y procesos, abarcando gran parte de los componentes de una herramienta CASE.

Developer 2000 es una herramienta de Bajo Nivel (Low - CASE) dirigida a las últimas fases del desarrollo como son construcción e implantación. Este utiliza la mayoría de componentes, excepto generadores de bases de datos y metadatos.

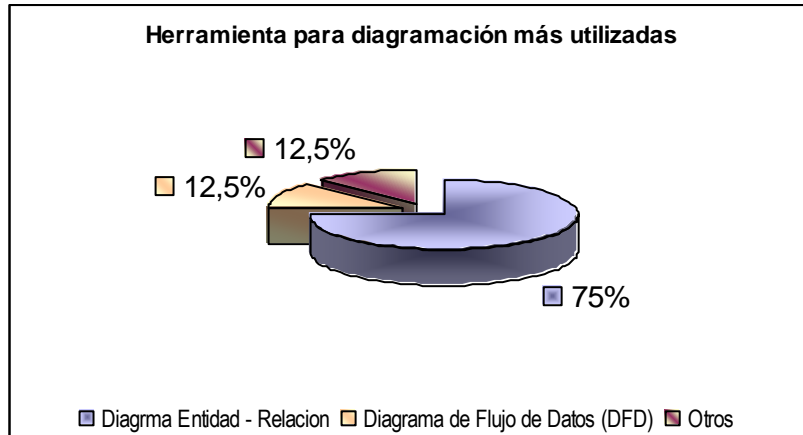
Mientras que Crystal Report es una herramienta de tipo toolkits dirigida a automatizar una fase dentro del ciclo de vida, proporcionando componentes de diseño.

Pregunta No 6

Si la herramienta CASE que utiliza posee herramientas para diagramación, indique cuáles de las siguientes:

Respuesta:

RESPUESTA	PowerDesigner	Ez-Erwin	Crystal Report	Developer 2000
Diagrama de Flujo de Datos (DFD)		X		
Flujogramas				
Diagramas de Transición de Estados (DTE)				
Diagrama Entidad – Relación	X	X		X
Otros			X	



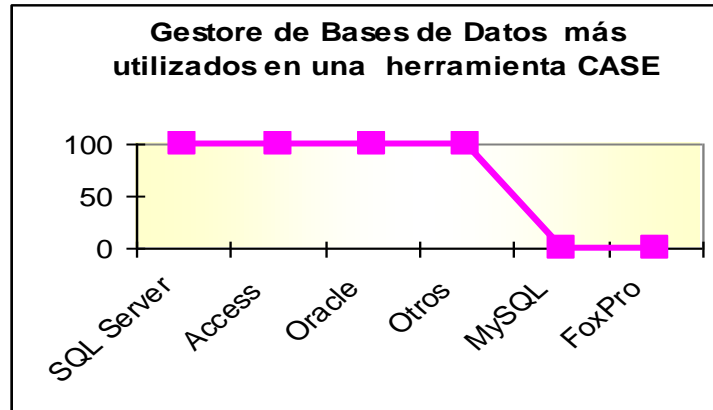
Se demuestra que la herramienta para diagramación más utilizada es el Diagrama Entidad - Relación ya que de las cuatro herramientas CASE que utiliza la población encuestada, el 75% incluyen herramientas para este tipo de diagramación.

Pregunta No 7

Si la herramienta CASE que utiliza incluye generadores de bases de datos, seleccione el gestor(es) con los cuales interactúa la herramienta:

Respuesta:

RESPUESTA	PowerDesigner	Ez-Erwin
Microsoft SQL Server	X	X
Microsoft Access	X	X
MySQL		
Oracle	X	X
Otros	X	X
FoxPro		



De las herramientas CASE que incluyen generadores de bases de Datos se encuentra PowerDesigner y Ez-Erwin que incluyen generadores de bases de Datos como: SQL Server, MS Access, Oracle, Sybase, Informix, Interbase y otros

Pregunta No 8

Si la herramienta CASE que utiliza posee generadores de código, indique el lenguaje de programación en que se realiza la codificación:

Respuesta:

RESPUESTA	DESARROLLADORES
C/C++	0
Visual Basic	0
Visual FoxPro	0
Delphi	0
Java	0
Otros:	9

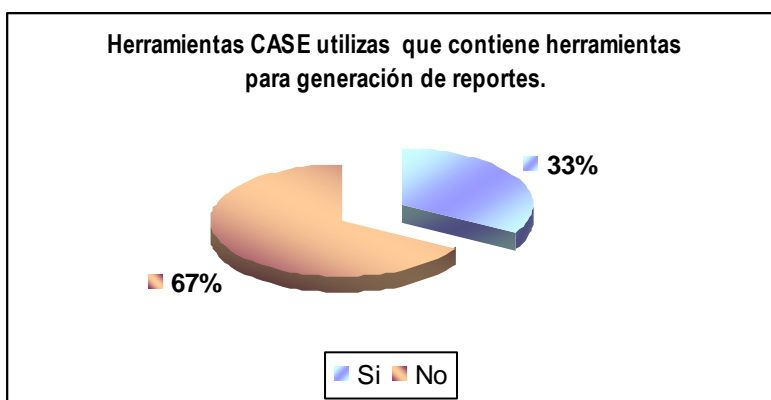
Se puede observar que las herramientas CASE poseen generadores de código para la generación de script en SQL como es el caso de Power Designer y Ez -Erwin, mientras que en Developer 2000 genera código PL/SQL, además de generación de código para crear sus propias interfaces para manipulación de datos.

Pregunta No9

¿La herramienta CASE que utiliza contiene herramientas para generación de reportes?

Respuesta:

RESPUESTA	DESARROLLADORES	%
Si	3	33.3%
No	6	66.67 %



De la población encuestada, el 33.33% de las herramientas CASE utilizadas contienen herramientas para generación de de reportes.

Pregunta No10

¿Los reportes generados por la herramienta CASE muestran información cruzada a partir de tablas relacionadas?

Respuesta:

RESPUESTA	PORCENTAJE
Si	100%
No	0



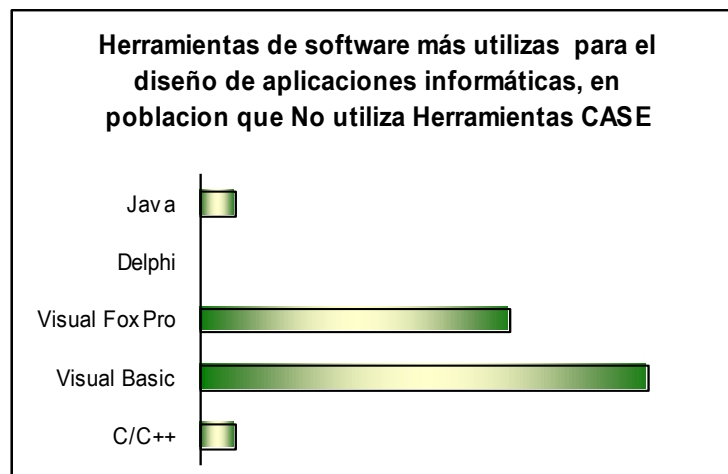
Se demuestra que de las herramientas CASE utilizadas generadores de reportes, muestran información cruzada a partir de tablas relacionadas.

Pregunta No11

¿Qué herramientas de software utiliza para el diseño de sus aplicaciones informáticas?

Respuesta:

RESPUESTA	DESARROLLADORES
C/C++	1
Visual Basic	12
Visual FoxPro	9
Delphi	0
Java	1



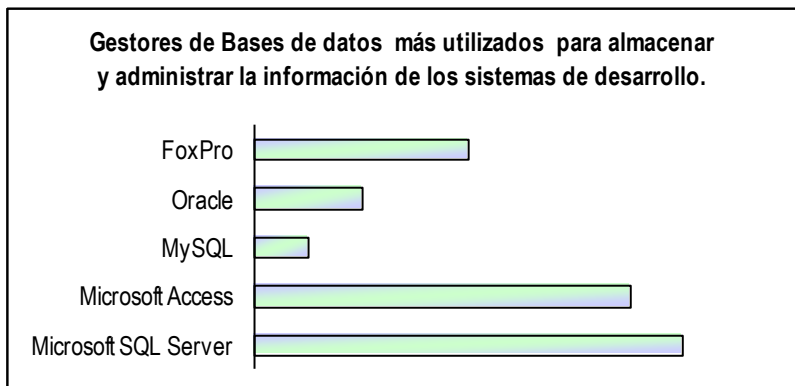
De la muestra poblacional que **NO utiliza** herramientas CASE, la herramienta de software más utilizada para el diseño de las aplicaciones informáticas es Visual Basic seguida por Visual FoxPro , y en muchas de las empresas se puede observar que utilizan más de una herramienta para el desarrollo de sus aplicaciones.

Pregunta No12

¿Qué gestor de base de datos utiliza para almacenar y administrar la información de los sistemas que desarrolla?

Respuesta:

RESPUESTA	DESARROLLADORES
Microsoft SQL Server	8
Microsoft Access	7
MySQL	1
Oracle	2
FoxPro	4



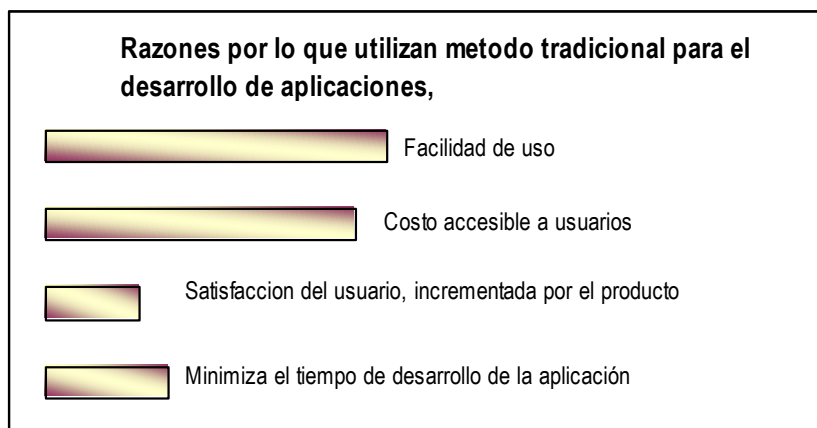
De la población que **NO utiliza** herramientas CASE, los gestores de bases de datos que más se utilizan para almacenar y administrar la información de los sistemas que desarrollan son SQL Server, MS Access y MS FoxPro. También hay que hacer notar que muchas empresas utilizan más de un gestor de bases de datos.

Pregunta No13

¿Por qué razón(es) utiliza ésta(s) herramienta(s)?

Respuesta:

RESPUESTA	DESARROLLADORES
Minimiza el tiempo de desarrollo de la aplicación	4
Satisfacción del usuario, incrementada por el producto	3
Costo accesible a usuarios	10
Facilidad de uso	11



Las principales razones por las que se prefieren los lenguajes de programación antes mencionados son: el costo, por ser accesibles a los usuarios y la facilidad de uso para programadores.

2.3.2 ANALISIS DE RESULTADOS DE ENTREVISTAS

A continuación se presentan una síntesis de los resultados de las entrevistas realizadas:

Pregunta No 1

¿Qué concepto tiene de una herramienta CASE?

Respuesta:

El personal entrevistado concibe que es una herramienta para automatización de procesos para el Diseño de sistemas.

Pregunta No 2

¿Cuáles son los beneficios que se obtienen al utilizar una herramienta CASE?

Respuesta:

- × Generación de código
- × Documentación de procesos
- × Accesibilidad de información
- × Optimización de tiempo empleado en desarrollo y mantenimiento de información.

Pregunta No 3

¿Qué debilidades observa en las herramientas CASE?

Respuesta:

- × Poco uso de herramientas CASE y de metodologías de análisis estructurado en El Salvador
- × Usuarios lo estiman complejo el desarrollar con herramientas CASE.
- × Permiten solamente especificación de datos generales.

Pregunta No 4

De las herramientas CASE que usted conoce, ¿cuál es la que ofrece mayores beneficios en el mercado actual? ¿Por qué?

Respuesta:

Power Designer, Erwin, ya que son las más conocidas y utilizadas.

Pregunta No 5

¿Considera que es más eficiente y efectivo utilizar una herramientas CASE en el desarrollo de sistemas, que realizar los procesos con el método tradicional? ¿Por qué?

Respuesta:

Si, por la facilidad de actualización de información y generación de reportes. Haciéndose notar que es más efectivo y eficiente utilizar una herramienta CASE cuando se inicia el desarrollo de un sistema.

Pregunta No 6

¿A cuánto asciende (en promedio) el costo de las herramientas CASE existentes en el mercado actual?

Respuesta:

Oscilan entre \$2,000 a \$20,000

Pregunta No 7

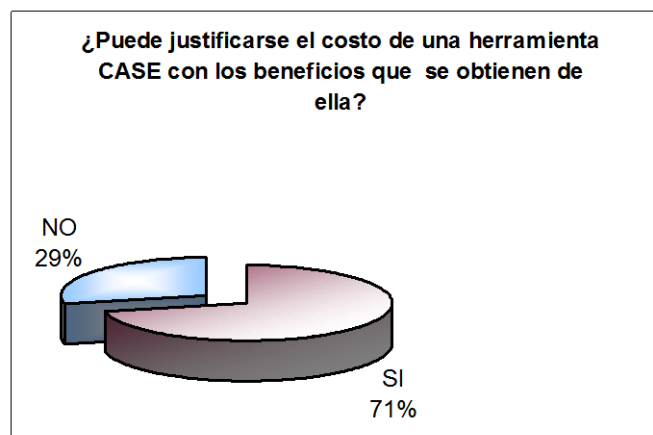
¿Puede justificarse el costo de una herramienta CASE con los beneficios que se obtienen de ella? ¿Por qué?

Respuesta:

Si, por los resultados que se observan, ya que se busca un sistema automatizado eficiente que minimice tiempo en el desarrollo.

Sin embargo, algunas empresas no consideran necesaria la inversión por no incurrir en costos muy elevados. De los profesionales entrevistados, algunos pertenecen a empresas medianas o grandes en las cuales el giro del negocio está orientado a la obtención de beneficios económicos con el desarrollo de sistemas y no sólo a la creación de nuevas aplicaciones para uso interno, por lo que pueden recuperar posteriormente el costo que implica la obtención de una herramienta CASE.

Los resultados son mostrados en la siguiente gráfica:



Pregunta No 8

¿De qué manera podría garantizarse el éxito de implantación de una herramienta CASE?

Respuesta:

- × Capacitación de personal que la utilizará
- × Fomentando el uso de la herramienta CASE
- × Documentando las aplicaciones

2.3.3 ANÁLISIS DE LA SITUACIÓN ACTUAL

Con el estudio realizado, al efectuar la tabulación de las encuestas y entrevistas se observa que las herramientas CASE no son muy populares en nuestro medio, sobre todo por los elevados costos que su implantación implica. Sin embargo, muchas empresas están interesadas en utilizarlas por la agilización de tareas que proporcionan.

En la medida que existan herramientas que permitan agilizar estas tareas, será posible desarrollar sistemas de mayor calidad y en menor tiempo.

2.3.4 FACTIBILIDAD DE IMPLANTACIÓN

El resultado de la investigación descrita anteriormente conlleva a la determinación de que la implantación de la herramienta CASE **Integral CASE** es factible. En esta parte se describen los tres criterios de factibilidad tomados en cuenta:

2.3.4.1. Factibilidad Técnica

Se ha determinado, de acuerdo a la investigación, que la herramienta puede ser utilizada en un **75%** en las instituciones que realizan tareas de desarrollo de sistemas, por la compatibilidad que ofrece con tres de los gestores de bases de datos más utilizados: MS Access, MS SQL Server, MySQL y con un lenguaje de desarrollo que está siendo cada vez más popular en nuestro medio: MS Visual Basic

2.3.4.2. Factibilidad Económica

Las encuestas y entrevistas realizadas, reflejan que los beneficios obtenidos al implantar una herramienta CASE hacen que los costos de inversión sean aceptables,

ya que el 71% de las empresas entrevistadas consideran que el costo de una herramienta CASE se justifica con los beneficios que se obtienen de ella.

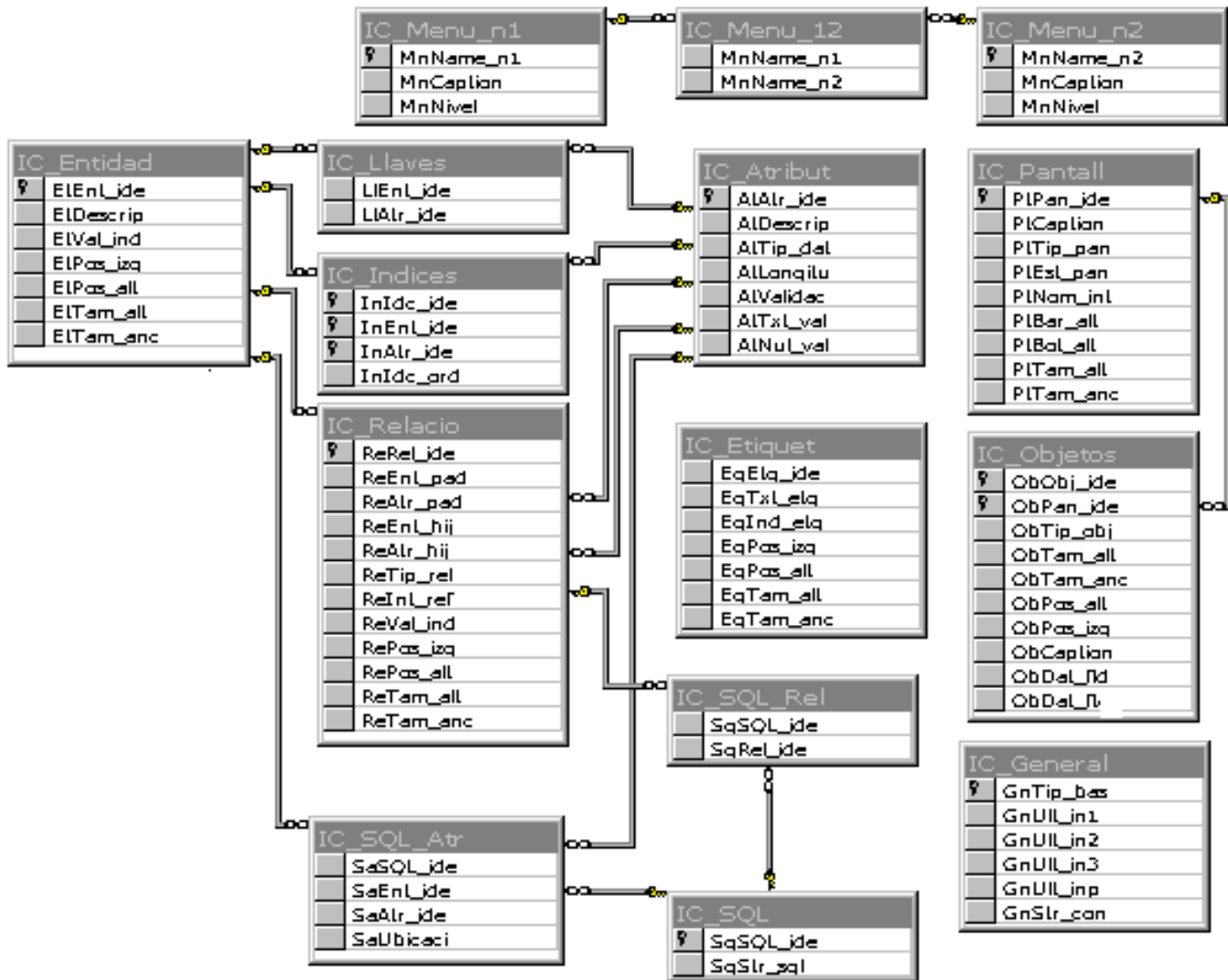
2.3.4.3. Factibilidad Operativa

El **81.25%** de los analistas y programadores coincide en que la implantación de una herramienta CASE facilitaría en gran medida el desarrollo de sus aplicaciones, debido a que estas herramientas están diseñadas para aumentar la eficiencia en el diseño. Por lo que se considera que al ser implantada, existirá solamente un **18.75%** de resistencia al cambio.

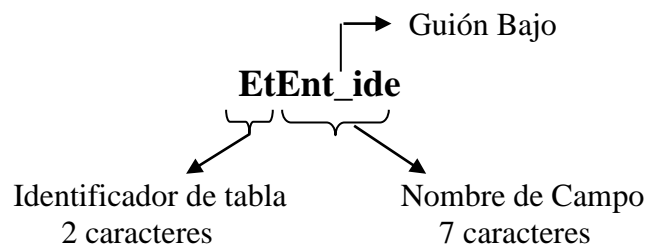
CAPITULO III

ANÁLISIS Y DISEÑO DE LA HERRAMIENTA CASE

3.1 DIAGRAMA ENTIDAD - RELACION PARA EL DISEÑO DEL DICCIONARIO DE DATOS A SER UTILIZADO POR LA HERRAMIENTA CASE: "INTEGRAL CASE"



Nombres de Campos:



TABLAS DEL DICCIONARIO DE DATOS

Tabla: IC_Entidad

Tabla que almacena la información de cada una de las entidades que forman parte del Diagrama E-R. Además, se utiliza para comprobar que el usuario no introduzca nombres iguales a diferentes entidades en un mismo diagrama.

Detalle de Campos

Campo	Tipo	Long.	Descripción
EtEnt_ide	Text	64	Campo que almacena el nombre que el usuario otorga a la Entidad. Si no se define un nombre específico, en este campo se almacena el texto Tabla seguido de un valor auto numérico.
EtDescrip	Text	255	Este es un campo opcional, el usuario puede introducir en él una breve descripción de las características de la entidad que está creando.
EtVal_ind	Integer		Almacena el número que identifica a una Entidad de un grupo de Entidades.
EtPos_izq	Single		Almacena la distancia entre el borde izquierdo interno de la Entidad y el borde izquierdo del área de diseño.
EtPos_alt	Single		Almacena la distancia entre el borde superior interno

			de la Entidad y el borde superior del área de diseño.
EtTam_alt	Single		Campo que almacena el alto del objeto Entidad.
EtTam_anc	Single		Campo que almacena el ancho del objeto Entidad.

Tabla: IC_Atribut

Almacena los atributos que pertenecen a una determinada entidad del Diagrama E-R. Las modificaciones realizadas a éstos son automáticamente reflejadas en la Tabla Atributo, también se utiliza para realizar consultas y verificar que no se repitan nombres de atributos para una misma entidad.

Detalle de Campos

Campos	Tipo	Long.	Descripción
AtAtr_ide	Text	64	Campo que almacena el nombre que el usuario proporciona a los atributos que forman parte de la Entidad.
AtDescrip	Text	255	Almacena una descripción de las características del atributo.
AtTip_dat	Text	15	Almacena la clase de valores que el usuario puede guardar en el Atributo, es decir se almacena el tipo de dato.
AtLongitu	Integer		Máximo número de caracteres que se puede

			introducir en el Atributo.
AtValidac	Text	255	Expresión que limita los valores que el usuario puede introducir en el Atributo.
AtTxt_val	Text	255	El usuario escribe un Mensaje de Error que aparece cuando introduce un valor prohibido por el Campo Validación.
AtNul_val	Integer		Determina si el atributo permite valores null

Tabla: IC_EntAtri

Almacena la información de la relación existente entre las tablas Entidad y Atributo, es decir que define los atributos que pertenecen a una determinada entidad.

Detalle de Campos:

Campos	Tipo	Longitud
EaEnt_ide	Text	64
EaAtr_ide	Text	64

Tabla: IC_Llaves

La información de todas las llaves creadas es almacenada en esta tabla. Se definen las llaves primarias de cada entidad del Diagrama.

Detalle de Campos:

Campos	Tipo	Longitud
LIEnt_ide	Text	64
LIAttr_ide	Text	64

Tabla: IC_Indices

Guarda datos sobre índices de búsquedas y ordenamientos definidos en las Entidades.

Solamente será utilizado cuando el gestor seleccionado sea Microsoft Access.

Detalle de Campos:

Campos	Tipo	Long.	Descripción
InIdc_ide	Text	64	Almacena el nombre que hace referencia al índice creado.
InEnt_ide	Text	64	Entidad a la que pertenece el índice
InAttr_ide	Text	64	Atributos involucrados en el índice
InIdc_ord	Text	25	Almacena el orden del Índice creado (ascendente o descendente)

Tabla: IC_Relacio

Almacena las relaciones definidas entre Entidades.

Detalle de Campos:

Campos	Tipo	Long.	Descripción
ReRel_ide	Text	64	Nombre que identifica a la relación creada.
ReEnt_pad	Text	64	Identifica la entidad principal de una relación.
ReAtr_pad	Text	64	Identifica el atributo principal de una relación
ReEnt_hij	Text	64	Identifica la entidad externa de una relación
ReAtr_hij	Text	64	Identifica el atributo foráneo de una relación
ReTip_rel	Integer		Especifica el tipo de relación a crear(uno a uno, uno a muchos ó muchos a muchos)
ReInt_ref	Integer		Especifica la integridad referencial aplicada a la relación.
ReVal_ind	Integer		Almacena el número que identifica a una Relación.
RePos_izq	Single		Almacena la distancia entre el borde izquierdo interno de la relación y el borde izquierdo del área de diseño.
RePos_alt	Single		Almacena la distancia entre el borde superior interno de la relación y el borde superior del área de diseño.
ReTam_alt	Single		Almacena el alto del objeto relación.
ReTam_anc	Single		Almacena el ancho del objeto relacionado.

Tabla: IC_Pantall

Almacena los datos de pantallas diseñadas por el usuario.

Detalle de Campos:

Campos	Tipo	Long.	Descripción
PtPant_ide	Text	255	Nombre que el usuario otorga para identificar a la pantalla que se está diseñando.
PtCaption	Text	55	Almacena el texto mostrado en la barra de título del objeto Pantalla.
PtTip_pan	Integer	1	Almacena un indicador del tipo de pantalla
PtEst_pan	Integer	1	Almacena el estilo para pantallas de interfaz
PtNom_int	Text	55	Ruta de acceso al archivo de fondo de interfaz
PtBar_alt	Single		Posición de la barra de botones de mantenimiento
PtBot_alt	Single		Posición de la barra de botones Guardar y Cancelar
PtPos_alt	Single		Campo que almacena el alto del objeto Pantalla.
PtPos_anc	Single		Campo que almacena el ancho del objeto Pantalla.

Tabla: IC_Objetos

Almacena los objetos contenidos en cada una de las pantallas diseñadas.

Detalle de Campos:

Campos	Tipo	Long.	Descripción
ObObj_ide	Text	64	Guarda el nombre del objeto a crear sobre la

			pantalla de diseño.
ObPant_ide	Text	64	Relacionada con el campo Pant_id de la Tabla Pantallas, permite determinar o relacionar objetos a una determinada pantalla.
ObTip_obj	Text	15	Almacena el tipo de Objeto (texto, label, Listbox, etc.).
ObTam_alt	Single		Almacena el alto del objeto a crear en la pantalla.
ObTam_anc	Single		Almacena el ancho del objeto en la pantalla.
ObPos_izq	Single		Almacena la distancia entre el borde izquierdo interno del objeto y el borde izquierdo de la Pantalla.
ObPos_alt	Single		Almacena la distancia entre el borde superior interno del objeto y el borde superior de la Pantalla.
ObCaption	Text	55	Título de la Pantalla.
ObDat_fld	Text	64	Indica el atributo con el cual está relacionado el objeto.
ObDat_fte	Text	64	Indica la entidad de la cual se obtendrán los datos.

Tabla: IC_Etiquet

Almacena las etiquetas contenidas en el área de diseño.

Detalle de Campos:

Campos	Tipo	Long.	Descripción
EqEtq_ide	Text	64	Almacena el nombre correspondiente a la etiqueta creada sobre el área de diseño.
EqTxt_etq	Text	255	Guarda el texto editado por el usuario.
EqInd_etq	Integer		Almacena el número que identifica a una etiqueta.
EqPos_izq	Single		Almacena la distancia entre el borde izquierdo interno de la etiqueta y el borde izquierdo del área de diseño.
EqPos_alt	Single		Almacena la distancia entre el borde superior interno de la etiqueta y el borde superior del área de diseño.
EqTam_alt	Single		Campo que almacena el alto del objeto etiqueta.
EqTam_anc	Single		Almacena el ancho del objeto etiqueta.

Tabla: IC_General

Almacena información de variables de entorno utilizadas en el proyecto.

Detalle de Campos:

Campos	Tipo	Long.	Descripción
GnTip_bas	Integer		Almacena el tipo de gestor de base de datos seleccionado por el usuario
GnUlt_in1	Integer		Almacena el último valor de índice que hace referencia a la última entidad creada en el Diagrama E-R.
GnUlt_in2	Integer		Almacena el último valor de índice que hace referencia a la última relación creada en el Diagrama E-R.
GnUlt_in3	Integer		Almacena el último valor de índice que hace referencia a la última etiqueta de texto creada en el Diagrama E-R.
GnStr_con	Texto		Contiene las sentencias para la conexión a la base de datos, definidas en la conexión ODBC

Tabla: IC_Menu_n1

Almacena información de menús de nivel 1

Detalle de Campos:

Campos	Tipo	Long.	Descripción
MnName_n1	Text	64	Almacena el nombre usado en el código para identificar al objeto en el nivel 1.
MnCaption	Text	64	Devuelve o establece el texto mostrado en primer plano en el menú.
MnNivel	Integer		Guarda el nivel del menú.

Tabla: IC_Menu_n2

Almacena información de menús de nivel 2

Detalle de Campos:

Campos	Tipo	Long.	Descripción
MnName_n2	Text	64	Almacena el nombre usado en el código para identificar al objeto en el nivel 2.
MnCaption	Text	64	Devuelve o establece el texto mostrado en primer plano en el menú.
MnNivel	Integer		Guarda el nivel del menú.

Tabla: IC_Menu_12

Almacena datos de unión de menús de nivel 1 y 2.

Detalle de Campos:

Campos	Tipo	Long.	Descripción
MnName_n1	Text	64	Almacena el nombre usado en el código para identificar al objeto en el nivel 1.
MnName_n2	Text	64	Almacena el nombre usado en el código para identificar al objeto en el nivel 2.

Tabla: IC_SQL

Tabla que almacena las consultas creadas por el usuario.

Detalle de Campos:

Campos	Tipo	Long.	Descripción
SqSQL_ide	Text	64	Guarda el nombre de la Consulta a crear.
SqStr_sql	Text	64	Almacena el String SQL generado por la consulta creada.

Tabla: IC_SQL_Atr

Almacena datos de atributos incluidos en consultas

Detalle de Campos:

Campos	Tipo	Long.	Descripción
SaSQL_ide	Text	64	Almacena el nombre de la Consulta
SaEnt_ide	Text	64	Guarda la Entidad que es parte de la consulta.
SaAtr_ide	Text	64	Almacena el Atributo de la Entidad que forma parte de la consulta
SaUubicaci	Integer		Almacena la posición del Atributo (Si se encuentra en la lista superior izquierda o en la superior derecha), para efectos de recuperar los valores.

Tabla: IC_SQL_Rel

Almacena datos de relaciones incluidas en consultas

Detalle de Campos:

Campos	Tipo	Long.	Descripción
SqSQL_ide	Text	64	Almacena el nombre de la Consulta
SqRel_ide	Text	64	Almacena el nombre de la relación, para consulta de tablas relacionadas.

3.2 IDENTIFICACION DE PROCEDIMIENTOS

3.2.1 DISEÑO DE SALIDAS

Las salidas a obtener con el sistema de automatización **Integral CASE** son las siguientes:

- a) Diagrama Entidad Relación (modelo físico)
- b) Base de datos en el gestor seleccionado: MS Access, MS SQL Server o MySQL
- c) Proyecto de Visual Basic conteniendo: Código para conexión a base de datos generada, formularios de consulta y mantenimiento a tablas, Reportes sencillos de información contenida en tablas

3.2.2 DISEÑO DE ENTRADAS

Para la obtención de las salidas planteadas en el apartado anterior, se necesita de las siguientes entradas de datos:

- a) Definición de entidades, atributos y relaciones por parte del usuario
- b) Definición del gestor en el cual será generada la base de datos a partir del diagrama de E-R.
- c) - Conexión ODBC creada una vez que la base de datos ha sido generada.
 - Diseño de pantallas de mantenimiento a partir de las entidades y atributos definidos en el diagrama

- Para la creación de reportes, la base de datos creada por la herramienta a partir del E-R.

3.2.3. DISEÑO DE PROCESOS

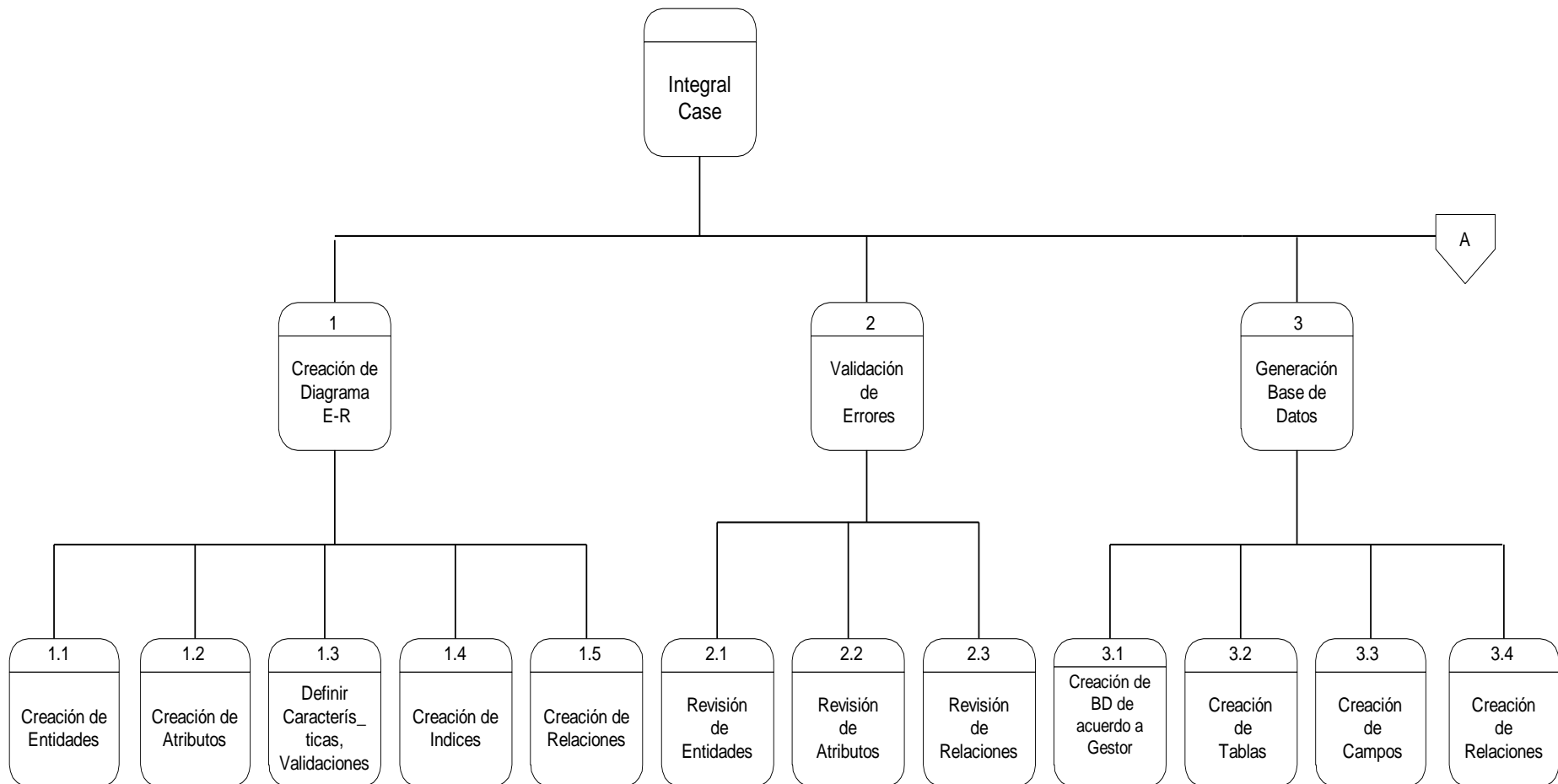
Los grandes procesos a realizar en Integral CASE, representan operaciones que ayudan al usuario en la automatización de tareas de desarrollo de sistemas, estos son los siguientes:

- a) Creación de Diagramas E-R
- b) Diseño de Pantallas
- c) Diseño de Consultas
- d) Generación de Bases de Datos
- e) Validación de Errores
- f) Conexión a Base de Datos
- g) Generación de Proyecto en Visual Basic

3.2.4. DIAGRAMAS DE ANÁLISIS

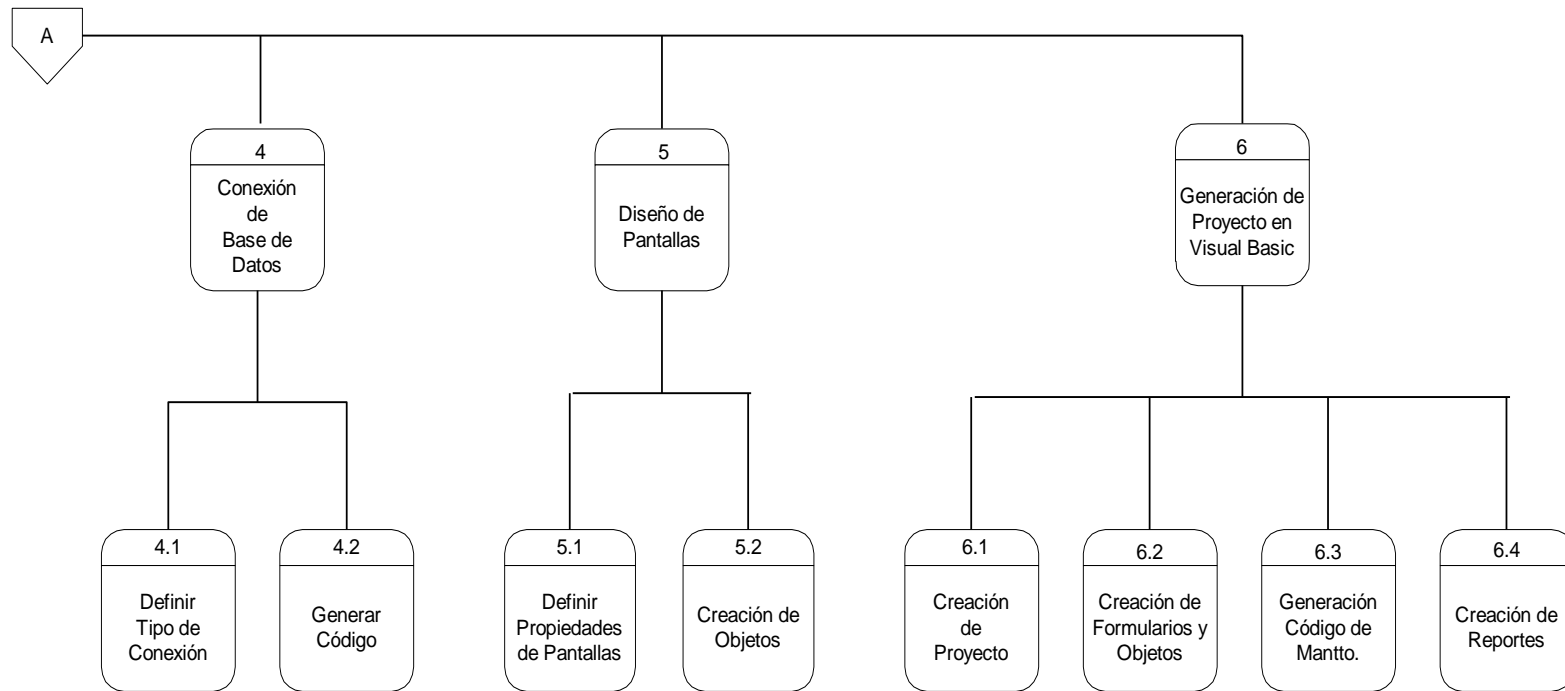
Los procesos planteados en el apartado anterior, se muestran en detalle (subprocesos) en la sección 3.2.4.1, a través del diagrama de jerarquía de procesos; a la vez en la sección 3.2.4.2 se definen las relaciones existentes y los flujos de información entre éstos, con los DFD (Diagramas de Flujo de Datos), los cuales han sido diseñados utilizando el enfoque de Gane y Sarson (**Ver Pág. 9**).

3.2.4.1 DIAGRAMA DE JERARQUIA DE PROCESOS PARA LA HERRAMIENTA CASE “INTEGRAL CASE”



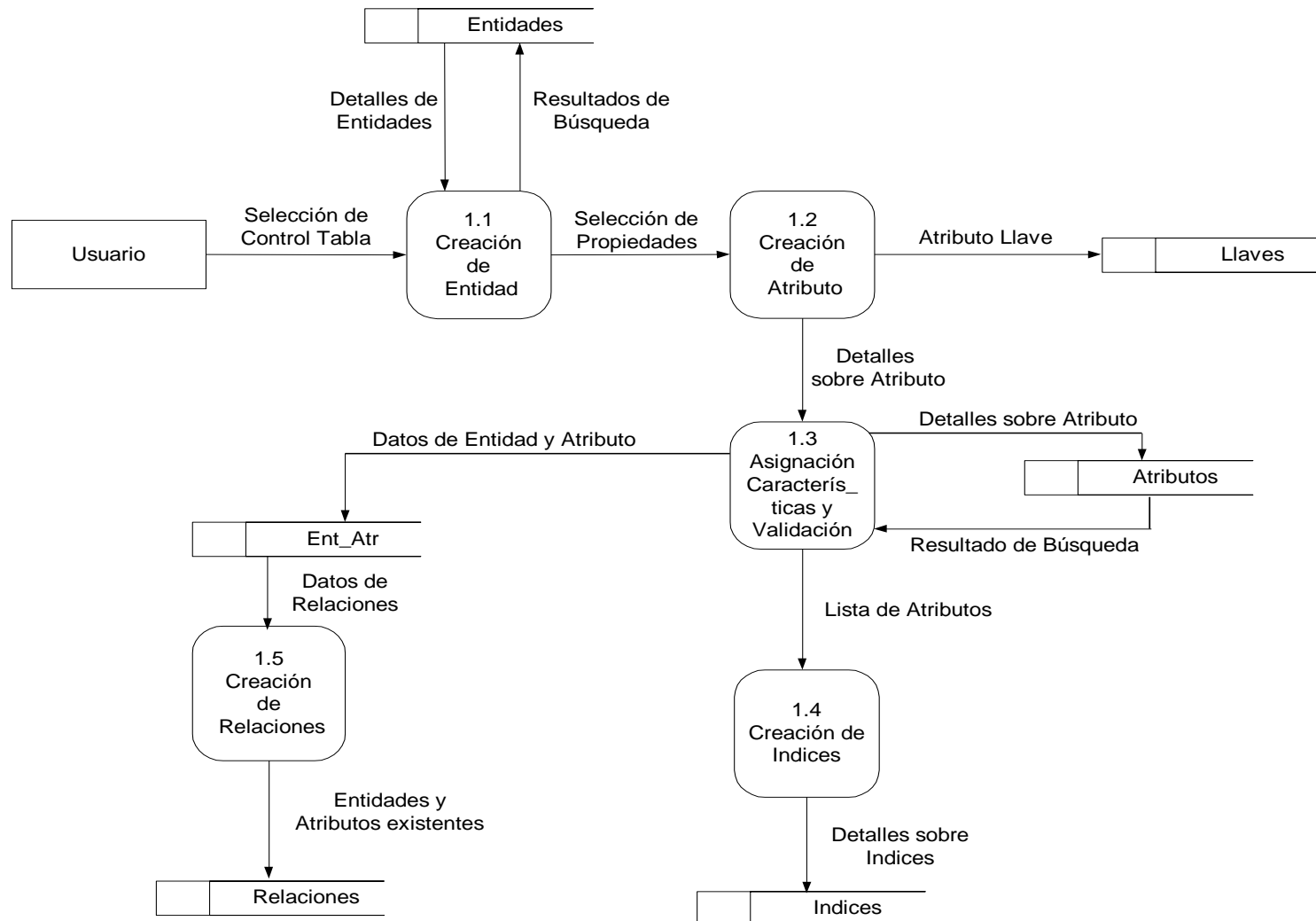
CONTINÚA....

DIAGRAMA DE JERARQUIA DE PROCESOS PARA LA HERRAMIENTA CASE “INTEGRAL CASE”

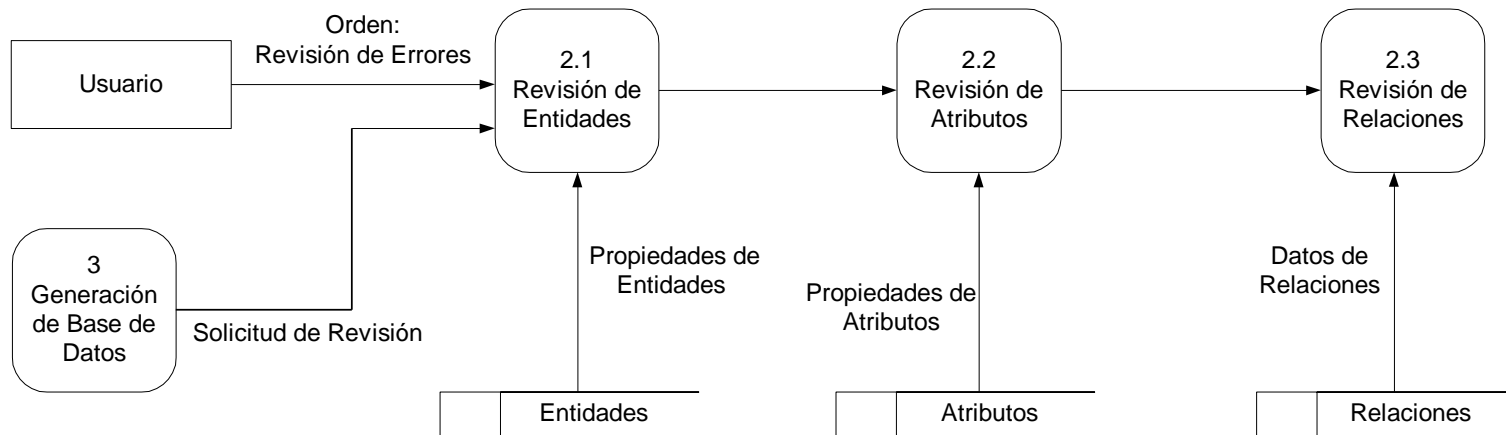


3.2.4.2 DFD DE PROCEDIMIENTOS DE LA HERRAMIENTA CASE “INTEGRAL CASE”

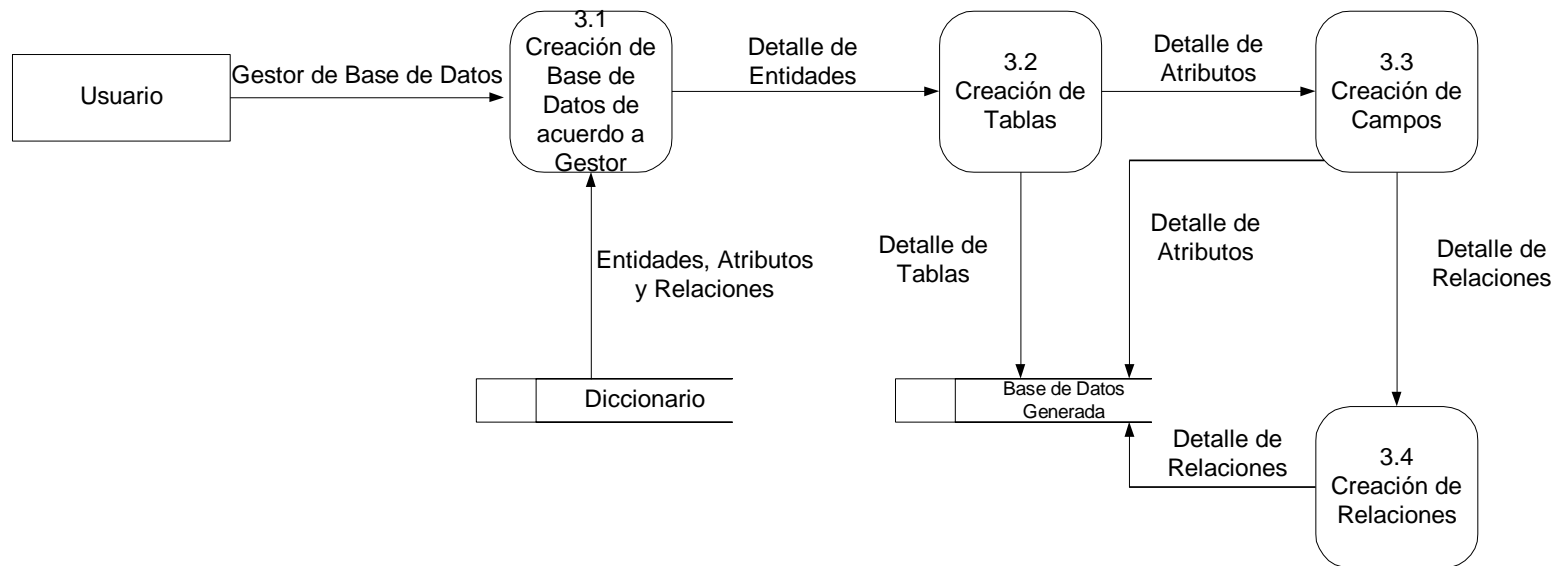
DFD Nivel 2: Creación de Diagramas E-R



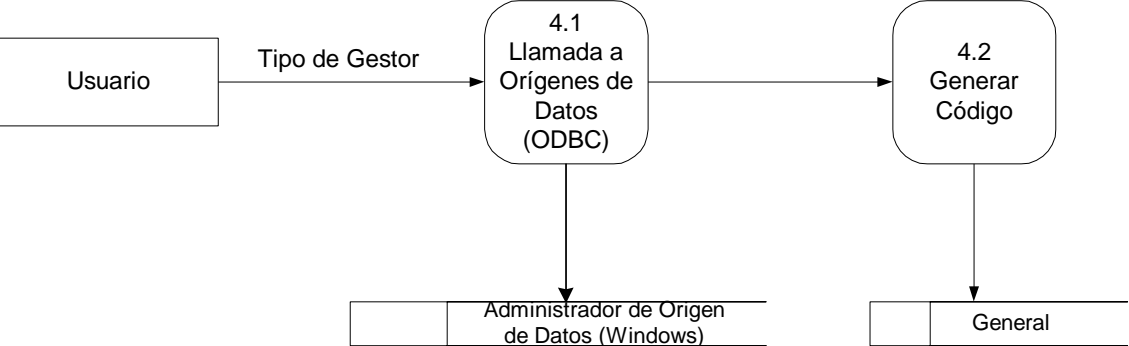
DFD Nivel 2: Validación de Errores



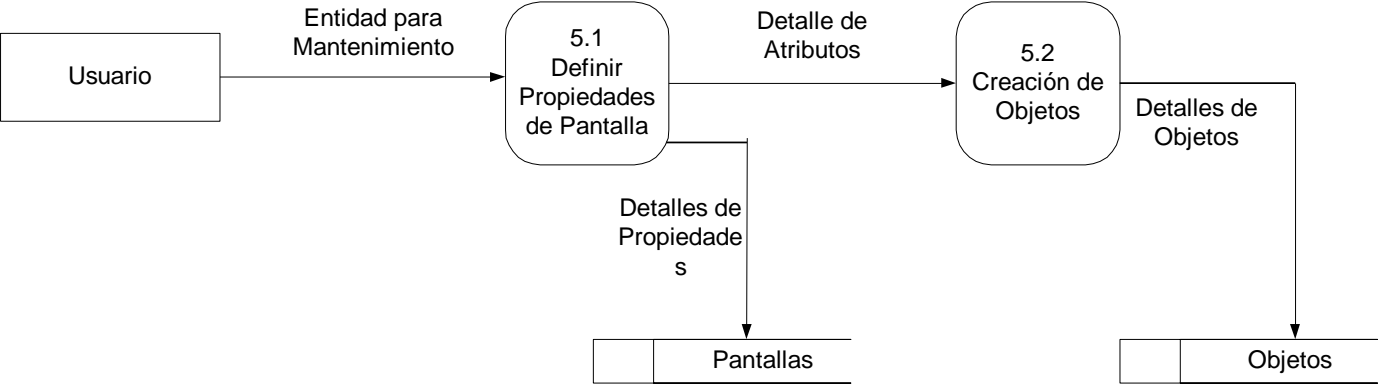
DFD Nivel 2: Generación de Base de Datos



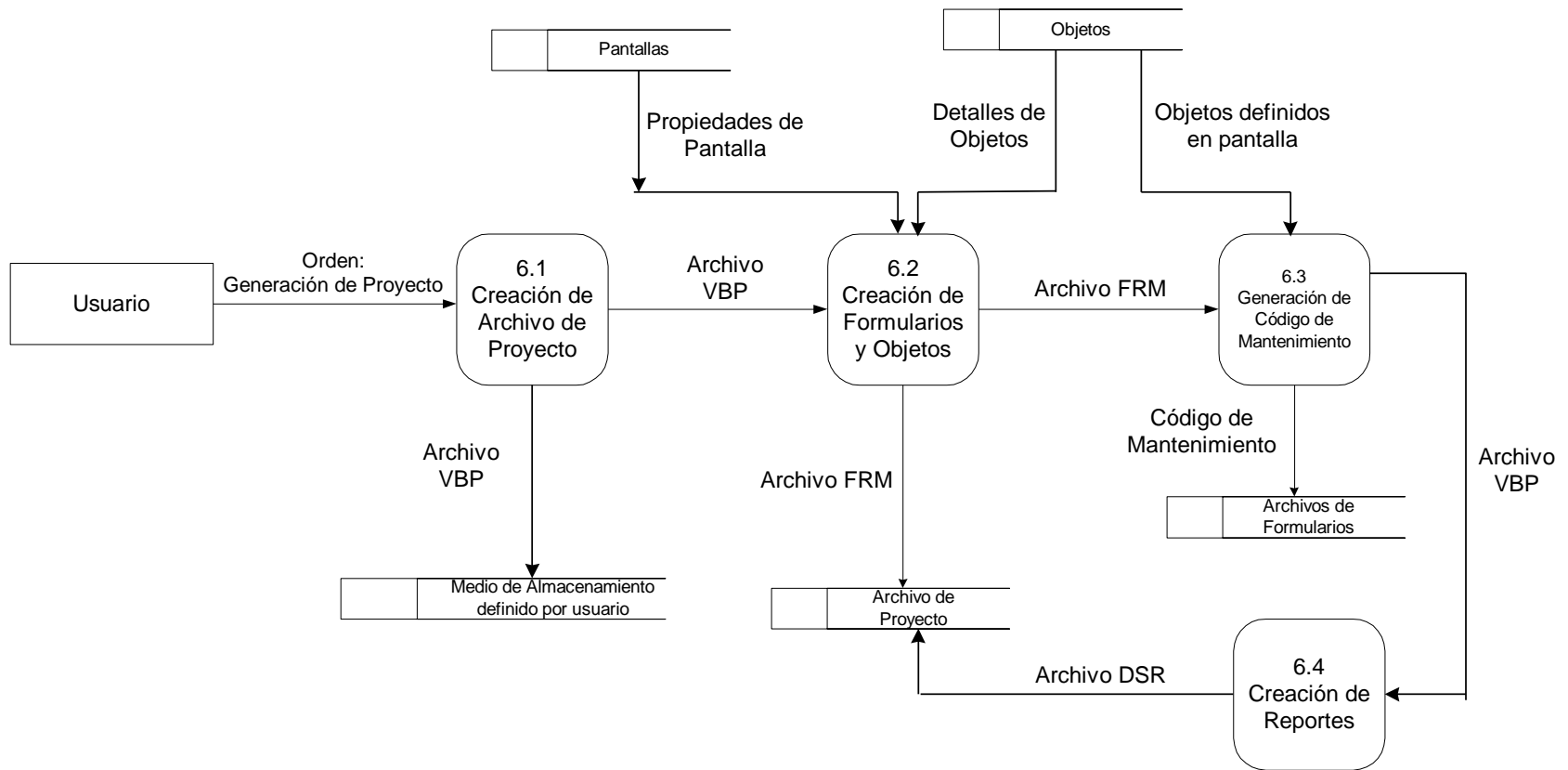
DFD Nivel 2: Conexión de Base de Datos (ODBC)



DFD Nivel 2: Diseño de Pantallas



DFD Nivel 2: Generación de Proyecto en Visual Basic



CAPITULO IV

DESARROLLO DE Integral CASE

4.1. DISEÑO DE CONTROLES E INTERFACES

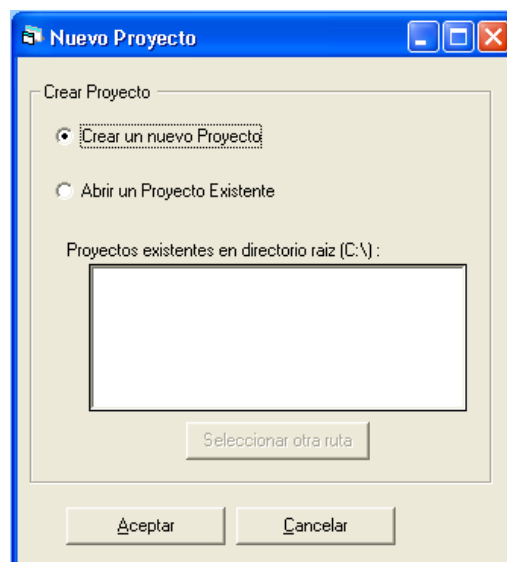
En esta sección se muestran las interfaces y controles diseñados para desarrollar los diferentes procedimientos con la herramienta Integral CASE, además la descripción de dichas interfaces y las opciones disponibles en el menú principal y en las barras de diseño, que obedecen a la ejecución de procesos planteados en el análisis y diseño de la herramienta.

4.1.1. Creación de interfaz de usuario de la herramienta Integral CASE: entorno de diseño, que se utiliza como interfaz para el usuario; en esta se encuentran disponibles las opciones para la creación de diagramas E-R, depuración de errores en el diagrama, generación automática de bases de datos y del prototipo de la aplicación de usuario.

4.1.1.1 Ventanas de Ingreso:

Crear o Abrir un Proyecto:

Permite definir el nombre con que se creará el proyecto o abrir uno existente, Integral CASE asignará la extensión **ddp**, para identificar sus archivos de proyecto.



Para abrir o crear un nuevo proyecto se realiza la llamada a los cuadros de diálogo **Guardar como** y **Abrir** de Windows utilizando el objeto **CommonDialog** y sus propiedades **ShowSave** y **ShowOpen**. Cuando se selecciona la opción Abrir un Proyecto Existente, se muestran los proyectos con extensión ddp almacenados en el directorio raíz; utilizando el objeto **File** asignando las siguientes propiedades:

```
File1.Path = "C:\\"
```

```
File1.Pattern = "*.ddp"
```

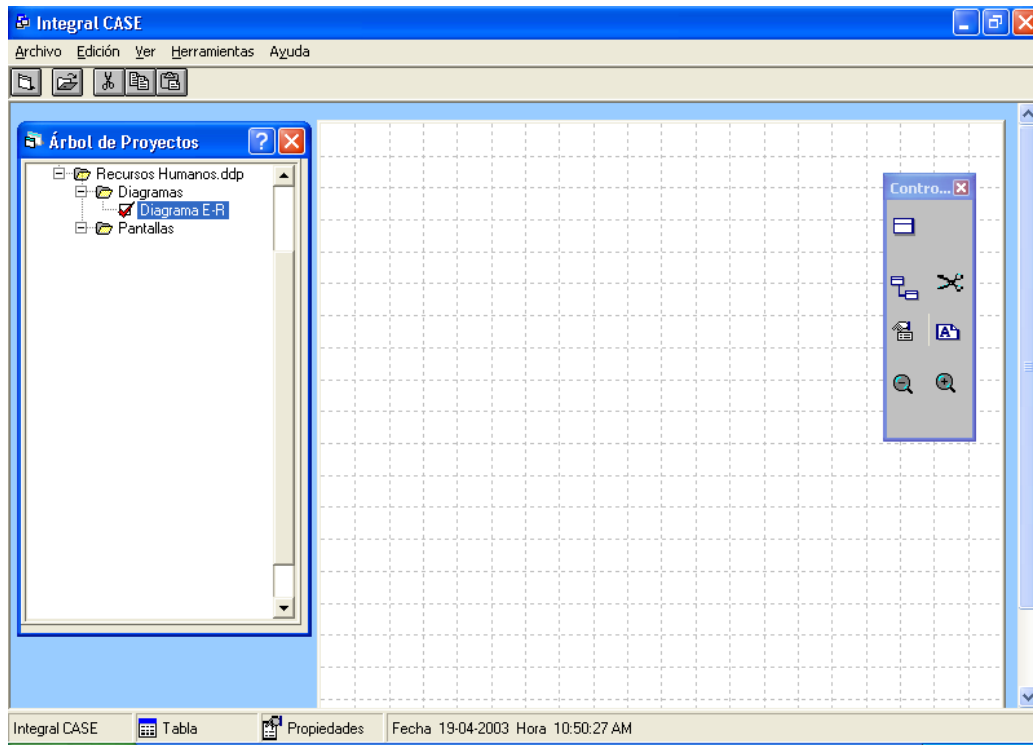
Además se habilita el botón **Seleccionar ruta** en caso que se desee buscar el archivo en otra ubicación.

Selección del gestor a utilizar para diseño de diagramas y generación de base de datos:



El cuadro **Tipo de Base de Datos** es visualizado cuando se crea un nuevo proyecto una vez que se le ha asignado un nombre. En la tabla de datos Generales del diccionario de datos existen un campo de tipo entero destinado a manejar el tipo de base de datos del proyecto, el valor asignado a este campo depende del gestor seleccionado en esta ventana: **1** para Access, **2** para Microsoft SQL Server, **3** para MySQL.

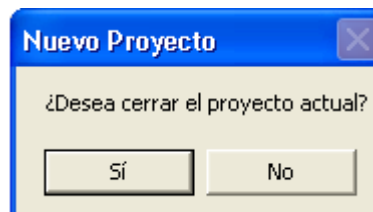
Área de Diseño:



4.1.1.2. Opciones del Menú Principal:

Menú archivo El menú archivo incluye las siguientes órdenes:

- **Crear un nuevo proyecto.** Si se encuentra abierto algún proyecto, esta opción muestra el cuadro de diálogo siguiente:



Si la respuesta a la pregunta es afirmativa se cierra el proyecto actual y a continuación se muestra el cuadro de diálogo Guardar como, en el cual se

almacenará el nombre y la ruta del proyecto a crear. Si la respuesta es No, se mantiene abierto el proyecto actual.

Si no existe ningún proyecto abierto al momento de la selección de esta opción, inmediatamente se muestra el cuadro de diálogo Guardar como.

- **Abrir un proyecto existente.** En esta opción, si se encuentra un proyecto abierto, muestra el cuadro de diálogo en el cual se pregunta si se desea cerrar el proyecto actual, si la respuesta es afirmativa, se muestra el cuadro de diálogo Abrir, en el cual se especifica la ruta y el nombre del proyecto que se desea abrir. Si no se encuentra abierto ningún proyecto al momento de seleccionar esta opción, solamente se muestra el cuadro de diálogo Abrir.
- **Guardar y Guardar como.** Esta orden permite guardar todos los archivos del proyecto que se encuentra abierto.
- **Imprimir.** Permite imprimir el Diagrama E-R creado por el usuario. Al seleccionar esta opción se abre el cuadro de diálogo Imprimir, el cual permite seleccionar la impresora y configurar el tipo de papel y su orientación.
- **Propiedades.** Muestra como su nombre lo indica las propiedades del proyecto actual, entendiéndose como propiedades la fecha de creación, el tamaño del archivo, etc.

- **Cerrar.** Esta opción cierra el proyecto actual.
- **Salir.** Permite salir de la Herramienta Integral Case, es decir que es el equivalente de presionar el botón de la esquina derecha de la barra de título.

Menú Edición

El menú edición contiene las opciones que se detallan a continuación:

- **Deshacer.** Orden que deshace la última acción realizada por el usuario.
- **Rehacer.** Permite que se repita la última acción que el usuario deshizo.
- **Cortar.** Esta opción permite eliminar texto y objetos creados, y lo coloca en el portapapeles de Windows.
- **Copiar.** Orden que crea una copia del texto y objetos seleccionados y lo coloca en el portapapeles.
- **Pegar.** Toma lo que haya en el portapapeles y lo pega en la aplicación actual.
- **Eliminar.** Esta opción suprime la información y controles seleccionados, pero no coloca una copia en el portapapeles.

Menú Ver

Contiene las opciones que se describen a continuación:

- **Barra de Controles.** Orden que permite visualizar la ventana que contiene los controles necesarios para creación de diagramas.
- **Árbol de Proyecto.** Permite mostrar la ventana de explorador de proyectos, es decir el árbol en el cual se detallan todos los componentes del proyecto actual.

Menú Herramientas.

El menú Herramientas permite visualizar las ventanas que forman parte del entorno principal de la Herramienta Integral Case, las opciones de las que consta se detallan a continuación:

- **Detección de errores en diagrama.** Opción que realiza una verificación de errores en la estructura del diagrama E-R, al finalizar muestra un reporte en el cual se detalla si existen o no errores en el diseño.
- **Generar base de datos.** Orden que crea la base de datos tomando como base el diagrama E-R realizado y el gestor seleccionado al momento de crear el proyecto actual (debe tenerse en cuenta que solo se generará la base de datos si el diagrama E-R se encuentra libre de errores).

- **Generar conexión ODBC.** Realiza una llamada al Administrador de Orígenes de Datos de Windows
- **Generar proyecto en Visual Basic.** Esta opción toma las pantallas creadas por el usuario en la Herramienta Integral Case y genera una nueva aplicación en Visual Basic. Dicha aplicación contendrá formularios para mantenimiento a tablas y reportes sencillos de la información contenida en éstas.

Menú Ayuda.

El Menú ayuda, brinda información adicional al usuario, consta de las siguientes opciones:

- **Temas de Ayuda.** Opción que muestra toda la información concerniente al sistema, como pueden ser controles, funciones, forma de operar, etc.

Los temas de ayuda han sido diseñados utilizando tecnología HTML, de tal manera que resulte fácil y amigable al usuario la búsqueda de información.

La forma en que se muestra la ayuda es a través de Libros en Pantalla, que contienen índices de contenidos y enlaces a las páginas HTML correspondientes al tema.

Constituye una de las herramientas más novedosas de Integral CASE, con el propósito de hacer más fácil y rápido el trabajo dentro de la herramienta.

- **Acerca de Integral CASE.** Orden que muestra las especificaciones y aspectos legales de Integral CASE.

4.1.1.3. Barra de Herramientas

La barra de herramientas está compuesta por los siguientes íconos, mediante los cuales se tiene un acceso rápido a ciertas opciones del Menú Principal.



Nuevo Proyecto.



Abrir un Proyecto Existente.



Cortar.



Copiar.



Pegar.

4.1.1.4. Árbol de Proyectos

La Herramienta Integral Case organiza las aplicaciones en Proyectos, el nombre de éste es introducido por el usuario al iniciar la aplicación. Cada Proyecto está formado por subcarpetas, en la primera de éstas se almacena el Diagrama E-R creado por el usuario, en la segunda las pantallas generadas y en la tercera las consultas creadas por el usuario. El árbol de proyectos aparece en la parte superior derecha de la pantalla.

También puede accederse al Árbol de Proyectos mediante la opción Árbol del Proyecto del Menú Ver.

El árbol de proyectos es manipulado a través de un control ActiveX llamado AXTree.ocx, que admite propiedades para generación de árboles de carpetas y archivos.

4.1.1.5. Barras de Controles

La barra de controles utilizada para el diseño de diagramas E-R consta de:



Regresa el puntero a su forma estándar y anula la última opción elegida.



Este control permite crear una nueva entidad dentro del área de diseño de la Herramienta Integral Case.



Permite crear relaciones entre entidades.



Modifica una relación entre entidades.



Muestra las propiedades del objeto.



Permite crear una etiqueta para que el usuario coloque información adicional dentro del área de diseño.



Reduce en un porcentaje el tamaño de la ventana de diseño junto con sus componentes.



Incrementa en un porcentaje el tamaño de la ventana de diseño junto con sus componentes.

La barra de controles para el diseño de pantallas consta de:



Regresa el puntero a su forma estándar y anula la última opción elegida.



Etiqueta, permite colocar rótulos de texto en la pantalla



Permite agregar cuadros de texto en el diseño de la pantalla



Utilizada para dibujar líneas en la pantalla



Agrega un control que muestra la hora del sistema



Permite colocar botones de comando en el diseño de la pantalla



Agrega un control que muestra la fecha del sistema



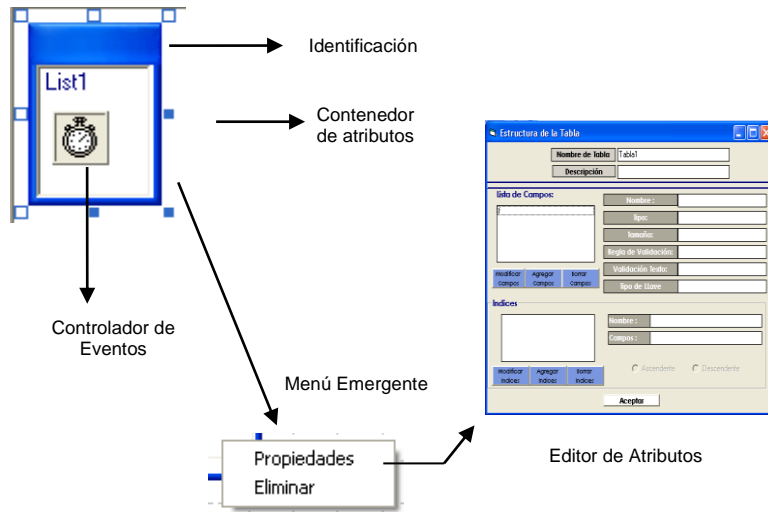
Utilizado para insertar un control combo

4.1.2. Diseño y creación de clases base:

Las clases base utilizadas como controles insertables para el diseño de los diagramas E-R son:

Control Tabla

Utilizado para la creación de entidades, compuesto por varios objetos:

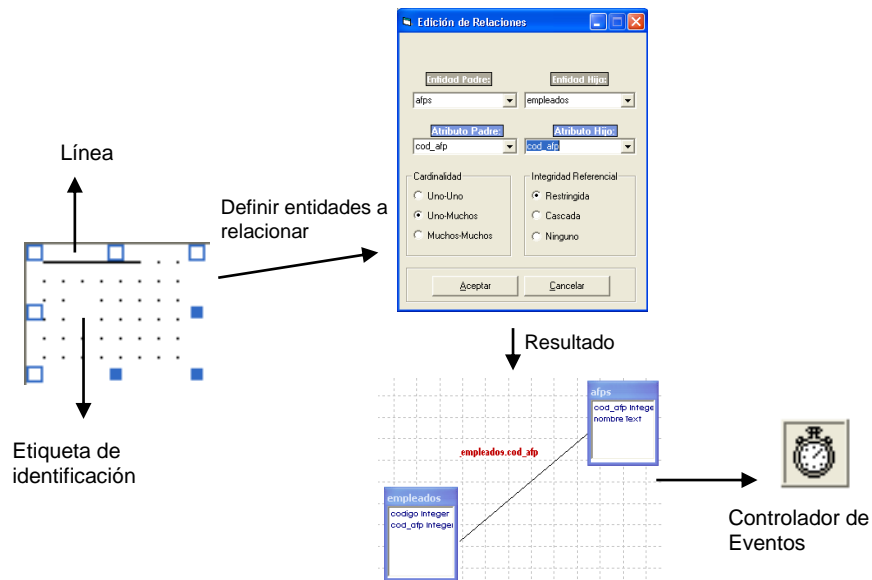


La barra de **Identificación** muestra el nombre asignado a la Entidad en las ventana de propiedades; el **Contenedor de atributos** está formado por un cuadro de lista con barras de desplazamiento, el cual almacena todos atributos definidos para la entidad junto con su tipo de dato; el **Controlador de Eventos** está constituido por un objeto Timer de Visual Basic, el cual despierta a intervalos de tiempo ejecutando una rutina de verificación de propiedades (posición de la entidad, longitud, etc), compara las propiedades anteriores con las actuales (al momento que despierta el Timer) y si se han realizado cambios actualiza con los nuevos valores la tabla de Entidades del diccionario de datos.

Control Relación

Utilizado para definir relaciones entre entidades, consiste en un control ActiveX formado por una línea y una etiqueta de identificación (label). Cuando se crea una relación o se cambian de posición las entidades involucradas en la relación se alteran ciertas propiedades del control ActiveX que permiten direccionar y dimensionar la línea de relación.

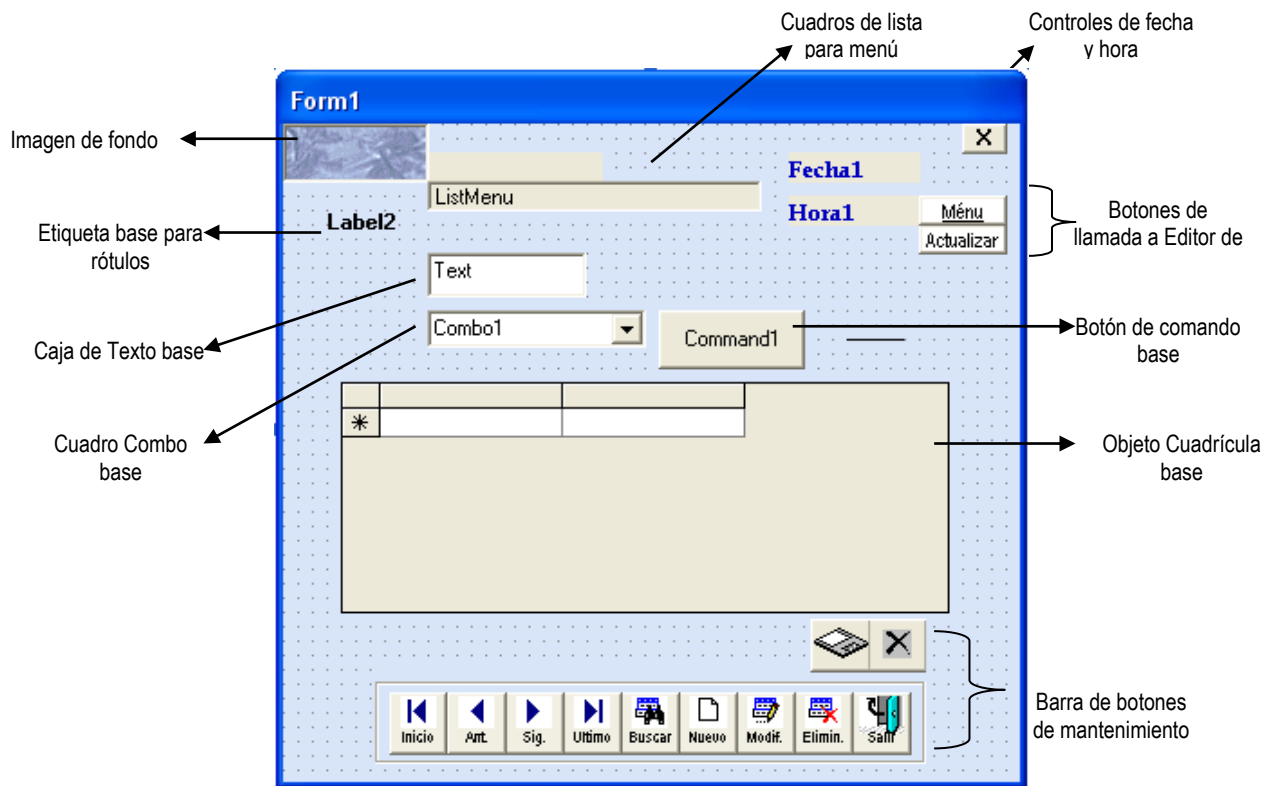
Cuando la relación se ha creado, este control interactúa con el Controlador de Eventos del control tabla para mantener permanentemente la unión de las entidades a través de la línea de relación.



Las clases base utilizadas para el diseño de pantallas son las siguientes:

Formulario de clase

A partir de este formulario y sus controles se crean instancias para nuevas pantallas y objetos, de acuerdo al tipo de pantalla. Las características y propiedades de cada instancia son almacenadas en el diccionario de datos. Incluye controles como la barra de mantenimiento (utilizada para pantallas de este tipo), cajas de texto y etiquetas base, botones de comando base, etc.



Operaciones para insertar controles en el área de diseño o en pantallas:

Cuando el usuario selecciona un control de la barra de controles, se activa una variable de memoria que indica que debe crearse una nueva instancia del objeto deseado, en el momento que se crea la instancia, las características iniciales se almacenan en el diccionario de datos en la tabla correspondiente (entidades, atributos, relaciones, objetos de pantallas, etc.)

4.2. DISEÑO DE ALGORITMOS

A continuación se describen los algoritmos empleados para la ejecución de todos los procesos de Integral CASE, tales como: creación del diccionario de datos, algoritmo para apertura de proyectos, algoritmo de validación de errores en diagramas E-R, algoritmo

para generación automática de bases de datos, algoritmo para conexión a base de datos generada, algoritmo para diseño de pantallas de interfaz y mantenimiento, algoritmo para definición de consultas, algoritmo para generación de proyecto en Visual Basic.

4.2.1. Diseño de base de datos para diccionario de datos

La base de datos se crea cada vez que se inicia con un nuevo proyecto, funciona con el motor de bases de datos Microsoft Jet, es creada mediante instrucciones de referencias DAO en Visual Basic y se le asigna la extensión “.**ddp**”. Este diccionario constituye el archivo de proyecto, en el cual se almacena toda la información referente a los objetos que conforman el proyecto creado por el usuario. Las instrucciones de código del algoritmo crean el diccionario en el orden que sigue:

- Construcción de la Base de Datos con extensión .ddp, en la ruta que el usuario indica al inicio de la creación del proyecto, para ello se utiliza el método CreateDatabase
- Construcción de cada tabla del diccionario con sus respectivos campos (ver sección 3.1), se utiliza el método CreateTableDef y la colección Fields, con la propiedad Append para definición de campos.
- Definición de llaves primarias de tablas
- Se hace uso del método CreateIndex para creación de índices de búsqueda de tablas
- Creación de relaciones entre tablas utilizando el método CreateRelation

4.2.2. Apertura de Proyectos

Al momento de abrir un proyecto existente, se ejecuta el algoritmo de apertura; el cual se encarga de recuperar todos los objetos que hayan sido creados hasta el momento en el proyecto, respetando todas sus características como posiciones, dimensiones, contenidos, etc.

Toda la información necesaria para este procedimiento se encuentra almacenada en el archivo de proyecto ddp (diccionario de datos), el orden que se sigue para la recuperación es la siguiente:

- **Creación de Recordset para cada una de las tablas del diccionario:** se realiza la conexión a la base de datos del proyecto y la apertura de las tablas.
- **Recuperación de Entidades y Atributos:** se buscan todas las características definidas en las tablas de Entidades, Entidades – Atributos, Atributos. A partir de estos datos se crean instancias del objeto Control Tabla, tantas como entidades se hayan almacenado en el diccionario de datos; asimismo para cada entidad se toman los datos de atributos almacenados.
- **Recuperación de pantallas:** para cada pantalla almacenada en la tabla de Pantallas del diccionario se crea una instancia del Formulario Clase para pantallas, y a la vez se agregan los ítems de pantallas al árbol de proyectos. En este mismo paso se recorre paralelamente la tabla de Objetos del diccionario para asignar a la pantalla todos los objetos que se colocaron la última vez que se hicieron cambios.

- **Recuperación de consultas:** se realiza la búsqueda de todas las consultas definidas en la tabla de Consultas y se agregan los ítems correspondientes en el árbol de proyectos. Los campos, tablas involucrados y la orden Select generada se recuperan al momento de abrir el generador de consultas.

4.2.3. Validación de errores en el diseño de diagramas E-R.

La validación de errores incluye un análisis de la estructura del diagrama, tanto en tiempo de creación como al ejecutar la opción de revisión de errores correspondiente; este análisis es posible mediante el apoyo del diccionario de datos del proyecto, la manipulación y control de los diferentes objetos contenidos en el diagrama, así como de los eventos relacionados con éstos.

Con la validación, la herramienta CASE determina si el diagrama E-R obedece a una estructura coherente, mediante mensajes de advertencia mostrados en pantalla durante la creación y el despliegue de un reporte con el estado actual del diagrama al ejecutar la opción de revisión.

Al ejecutar la opción de validación de errores correspondiente (en el menú principal) se realiza lo siguiente:

- a) Comprobar que cada entidad posea al menos un atributo: para lo cual se hace una búsqueda en la tabla de Entidades-Atributos del diccionario para determinar si las entidades definidas en el diagrama poseen atributos en esta tabla.
- b) Que a cada atributo le hayan sido asignadas las características: tipo de dato, longitud, tipo de llave.

- c) Que una entidad posea al menos un atributo clave (atributo principal) que identifique a cada una de sus instancias: para comprobarlo se realizan búsquedas de atributos (para cada entidad específica) en la tabla de definición de Llaves del diccionario de datos.
- d) Procurar que cada entidad se encuentre relacionada con una o más entidades y así evitar, en cuanto sea posible, la existencia de islas dentro del diagrama, esto puede ser determinado si se busca en la tabla de Relaciones del diccionario, que cada entidad se encuentre al menos en un registro de relación ya sea como entidad padre o como entidad hija.

4.2.4. Generación automática de bases de datos.

Este algoritmo permite la generación automática de base de datos de acuerdo a las especificaciones realizadas en el diagrama. Las instrucciones que se utilizan para la generación son:

- Código SQL, en el caso de SQL Server y MySQL
- Instrucciones de Referencias DAO en el caso de Access.

La generación de base de datos se hace atendiendo lo siguiente:

- a) El nombre del proyecto se asigna como nombre a la base de datos; para la creación se utilizan las instrucciones CreateDatabase (en el caso de Access) y la sentencia SQL: CREATE DATABASE (en el caso de SQL Server y MySQL); en este último caso, se realiza antes una conexión a la base de datos MASTER de SQL Server, para determinar si la base de datos ya existe y sobrescribirla con la nueva estructura.

- b) Cada entidad se convierte en una tabla de la base de datos; se recorre la tabla de definición de Entidades del diccionario de datos para generar una a una las tablas en el Gestor seleccionado. Para Access se utiliza la instrucción CreateTableDef y para SQL Server y MySQL la instrucción CREATE TABLE
- c) Los atributos y sus propiedades (tipo de dato, longitud, tipo de llave) pasan a ser los campos de cada tabla; se realiza una concatenación de sentencias para definir cada campo de las tablas planteadas en el literal anterior, mientras se recorre la tabla de definición de Atributos del diccionario.
- d) Los atributos definidos como llaves primarias en la tabla de Llaves del diccionario se incluyen en las tablas generadas como campos principales, utilizando instrucciones como la propiedad Primary (para Access) y Add Primary Key (para SQL Server y MySQL)
- e) Los índices definidos en la tabla de Indices del diccionario se incluyen como índices a través de las sentencias CreateIndex y CREATE INDEX
- f) Las relaciones definidas en el diagrama E-R, constituyen las relaciones correspondientes entre tablas, y son construidas con las sentencias CreateRelation y ADD CONSTRAINT / FOREIGN KEY

Este proceso no se realiza si se detecta (mediante el algoritmo de validación de errores) que el diagrama aún posee errores en el diseño.

4.2.5. Conexión a bases de datos

En este algoritmo se muestran al usuario las herramientas para definir la conexión ODBC que utilizará como acceso a la base de datos desde Visual Basic.

El algoritmo hace una llamada al Administrador de Orígenes de Datos de Windows; en el cual se define: la base de datos a la cuál se conectará la aplicación, el nombre de la conexión ODBC, el nombre del servidor, la forma en que se iniciará la autenticación y la definición del usuario y contraseña para la conexión.

Antes de iniciar con la generación de la conexión, Integral CASE determina si la base de datos del proyecto actual ya ha sido creada; en caso contrario, no es posible ejecutar este algoritmo.

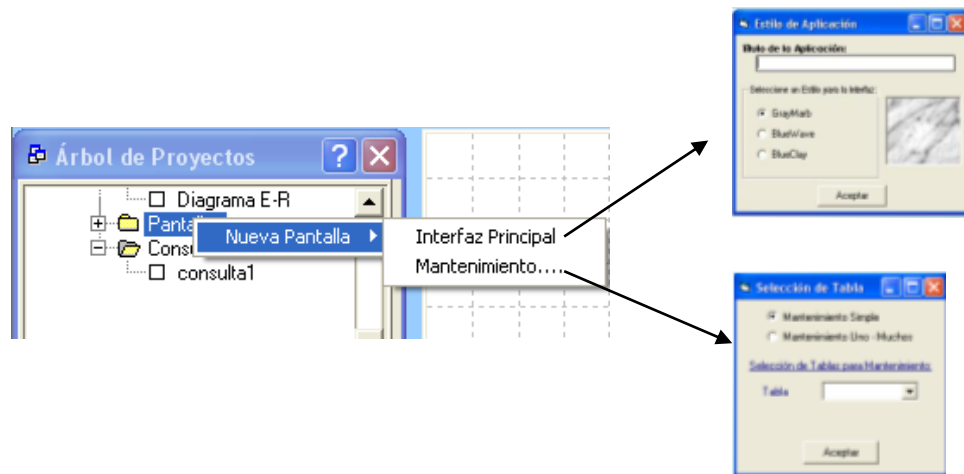
Al finalizar la creación de la ODBC a través del Administrador de Orígenes de Datos, se produce como resultado una cadena de conexión, la cual es almacenada en el diccionario de datos del proyecto actual.

Para conseguir lo anterior se utiliza el método OpenConnection acompañado de la constante dbDriverPrompt, que muestra la ventana de ODBC de Windows.

4.2.6. Diseño de pantallas de interfaz y mantenimiento

El diseño de pantallas se realiza a través de una llamada en el menú contextual mostrado en la carpeta “Pantallas” del árbol de proyectos; dicha llamada puede hacerse con dos opciones:

- Nueva pantalla de Interfaz principal
- Nueva pantalla de Mantenimiento



Con la opción **“Interfaz principal”** se muestra un cuadro de diálogo en el que se define el título de la aplicación y una imagen de fondo; mientras que la opción **“Mantenimiento”** proporciona dos alternativas en la creación de mantenimientos: Mantenimiento Simple y Mantenimiento uno-muchos. En el primer caso, solamente se indica la tabla para la cual se creará la pantalla; en el segundo debe definirse una tabla padre y una tabla hija relacionadas entre sí, de no estar debidamente definida una relación entre ellas, no es aceptada la petición de creación de la pantalla.

Una vez completadas las opciones anteriores, el algoritmo procede a generar de forma automática la pantalla correspondiente, colocando los controles necesarios para la interfaz o para entradas de datos a los campos en el caso de mantenimientos; en este último, la distribución se realiza de forma vertical colocando al pie de la pantalla una barra de botones para mantenimiento de datos (primer registro, adelante, atrás, último registro, nuevo, modificar, guardar y cancelar).

Para permitir la edición de los controles por el usuario, se hace uso de llamadas a las **API de Windows**, en donde se definen funciones para editar tamaño, posición, etc. Algo relevante en la edición de pantallas de **Interfaz**, es la generación automática del menú principal de la aplicación, en el cual se incluyen las opciones de llamada a las pantallas de mantenimiento definidas en el proyecto, a los reportes de datos y a las consultas; adicionalmente, el usuario puede definir sus propias opciones a ser incluidas en el menú, haciendo uso del editor de menú de Integral CASE.

Cuando el algoritmo finaliza la generación, muestra como resultado la pantalla en formato de diseño, en donde el usuario puede realizar cambios, modificar la distribución y colocar nuevos objetos desde la barra de controles (ver Pág. 83).

4.2.7. Definición de consultas

Las consultas en Integral CASE son creadas a través del menú contextual mostrado en la carpeta “Consultas” del árbol de proyectos. La generación de éstas requiere que el usuario indique las tablas y campos que estarán involucradas, con lo que la herramienta es capaz de diseñar una orden **Select** como resultado; para ello verifica las tablas y campos involucrados en la consulta, realiza búsquedas de las relaciones que existen entre tablas (a través del diccionario de datos); de esta manera involucra las sentencias **INNER JOIN** para definir relaciones y la orden de comparación para establecer la relación.

Todo lo anterior se realiza en el editor de consultas que se muestra a continuación:



Las tablas, campos involucrados en la consulta y la orden Select que se produce como resultado, son almacenados en el diccionario de datos del proyecto, en las tablas de Consultas correspondientes, para ser utilizados al momento de generar el producto final en Visual Basic.

4.2.8. Generación de proyecto en Visual Basic

El algoritmo de generación de proyecto constituye el proceso final en el desarrollo de proyectos en Integral CASE; con este paso la herramienta se encarga de transformar las pantallas y consultas en objetos reales de un proyecto en Visual Basic, los cuales interactúan entre sí mediante el código generado para cada formulario. Para la ejecución de este algoritmo será necesario haber definido pantallas de interfaz, de mantenimiento o consultas de tablas. Para detallar la complejidad del algoritmo se describen a continuación los subprocessos realizados en éste:

- **Distribución de archivos de imagen:** consiste en realizar una copia de los archivos de imagen necesarios para los formularios, botones, fondos, etc. de la aplicación a generar. Dichos archivos se distribuyen con la aplicación Integral CASE en una carpeta denominada **bmpps**, cuyo contenido se replica al directorio del proyecto del usuario mediante instrucciones FileSystemObject de Visual Basic.
- **Creación de archivo de proyecto .VBP:** edición del archivo de proyecto, el cual contiene las referencias a controles ActiveX, formularios y módulos; se le asigna el nombre del proyecto **ddp** creado con la herramienta CASE, para generarlo se abre un archivo (de extensión VBP) para escritura utilizando la instrucción **Open**; para la definición de sentencias en dicho archivo se ase uso de instrucciones como **Print #**.
- **Creación de módulos:** el algoritmo construye dos archivos con extensión **.bas** (módulos de Visual Basic), en los cuales se definen variables públicas como por ejemplo las utilizadas para conexiones y funciones públicas para la emisión de reportes.
- **Creación de formulario de interfaz:** a partir de la pantalla de interfaz definida en el proyecto de Integral CASE, se creará en este algoritmo un formulario principal (archivo con extensión **.frm**) que contendrá el fondo definido y el título de la aplicación. Además, se incluye el menú principal, con las opciones

de llamada a otros formularios, consultas, reportes y otras opciones definidas por el usuario; dicho menú podrá ser editado con el Editor de Menú de Visual Basic.

- **Creación de formularios de mantenimiento:** la creación de los formularios de acceso a datos de las tablas se realiza a partir de las pantallas diseñadas en Integral CASE, respetando la distribución que el usuario definió la última vez que realizó cambios. El algoritmo construye un archivo de formulario de Visual Basic (**.frm**) para cada pantalla definida en el proyecto de usuario; creando todos los controles involucrados para el mantenimiento y los adicionales que se han incluido en el diseño. Además, se genera de forma automática el código en lenguaje de Visual Basic para manipular dichos controles y para visualizar, adicionar, modificar y eliminar datos de las tablas de la base de datos.

En esta parte, se adiciona el código para realizar conexiones a la base de datos, utilizando la **sentencia de conexión** obtenida cuando el usuario creó la ODBC correspondiente; lo que significa que si el Sistema detecta que no se ha creado la ODBC, no es posible proceder con la generación de formularios de mantenimiento.

Las sentencias para conexión a la base de datos generadas se muestran como la siguiente:

```
str_con = "DSN=Planillas;APP=Visual Basic;WSID=XP;DATABASE _  
=Planillas;Trusted_Connection=Yes"  
  
cn.Open str_con
```

Las sentencias para apertura de una tabla son las siguientes:

```
rs.CursorType = adOpenDynamic  
rs.LockType = adLockPessimistic  
rs.Open "Select * from afp", cn, adOpenDynamic, adLockPessimistic
```

Las sentencias generadas para mantenimiento de datos de tablas se muestran de la siguiente forma:

```
- Ir al inicio          : rs.MoveFirst  
-Ir al final           : rs.MoveLast  
-Siguiete registro    : rs.MoveNext  
-Registro anterior    : rs.MovePrevious  
-Eliminar              : rs.Delete  
-Guardar Nuevo        : cn.Execute "insert into afp values ('" & _  
                        RTrim(txtcodigo_afp.Text) & "','" & _  
                        RTrim(txtnombre_afp.Text) & "','" & _  
                        RTrim(txtret_empleado.Text) & "','" & _  
                        RTrim(txtret_patrono.Text) & "','" & _  
                        RTrim(txtlimite.Text) & "')"   
  
-Modificar            :  
                        rs!codigo_afp.Value=RTrim(txtcodigo_afp.Text)
```

Además de las órdenes de mantenimiento mostradas, se generan validaciones para cada una de las operaciones; en las que se verifica la Integridad

Referencial, de los datos; dando a conocer al usuario los errores mediante mensajes como:

MsgBox (" Error al Guardar, se necesita un registro relacionado en otra Tabla")

- **Creación de formularios de consultas:** como se mencionó en la sección 4.2.6, en Integral CASE es posible realizar el diseño de consultas; el resultado de ellas es la generación de la instrucción **Select** que es almacenada en el diccionario de datos; a partir de ello el algoritmo de generación del proyecto en Visual Basic elabora un formulario por cada consulta diseñada; los cuales tienen la finalidad de mostrar a través de una cuadrícula los datos obtenidos de dicha consulta, tomando como fuente de datos la sentencia **Select** definida.

Algunas instrucciones que se generan en los formularios de consulta son las siguientes:

```
Adodc1.ConnectionString = "DSN=Planillas;APP=VisualBasic;WSID=XP;  
                           DATABASE=Planillas;Trusted_Connection=Yes"
```

```
Adodc1.RecordSource = "Select empleados.nombres,empleados.apellidos,  
                       empleados.cod_cargo,cargos.nombre_cargo FROM  
                       cargos INNER JOIN empleados ON  
                       cargos.cod_cargos= empleados.cod_cargo"
```

```
Adodc1.Refresh
```

```
Set DataGrid1.DataSource = Adodc1
```

- **Creación de reportes:** el último paso del algoritmo para generación del proyecto consiste en la elaboración de código para emisión de reportes básicos, los cuales podrán ser enviados a Microsoft Excel para su visualización. Para ello se construyen las instrucciones necesarias de creación de objetos del tipo “libro de trabajo”, necesarios para emitir hojas de cálculo (**archivos con extensión .xls**) desde Visual Basic. Es notable que esta generación será útil en el caso que el usuario posea acceso a Microsoft Excel para visualizar el resultado. Para realizar este proceso se utilizan sentencias de Visual Basic para Aplicaciones; en las cuales se crean objetos para aplicaciones de Office, en este caso se generan sentencias para creación de hojas de Excel, como por ejemplo:

```
Set objExcel = New Excel.Application  
objExcel.Visible = False  
objExcel.SheetsInNewWorkbook = 1  
objExcel.Workbooks.Add
```

El algoritmo de generación de proyecto en Visual Basic, concluye con la emisión de un mensaje en pantalla indicando la realización satisfactoria de todos los procesos anteriores.

4.3. OPCIONES DE DOCUMENTACIÓN

Como una opción de ayuda al usuario, se han incluido en Integral CASE opciones para la emisión de reportes, los cuales contienen documentación referente al diseño del diagrama Entidad – Relación; son cinco los reportes que pueden emitirse:

- Descripción General: que muestra un listado de todas las entidades creadas y sus respectivos atributos
- Reporte de Entidades: listado de entidades del diagrama
- Reporte de Atributos: listado de atributos con sus propiedades definidas
- Reporte de Índices: reporte de datos de índices de tablas
- Relaciones: resumen de relaciones definidas entre entidades.

El objetivo principal de la documentación es proporcionar al usuario una herramienta de registro de actividades dentro de la herramienta CASE.

Para generar esta documentación se han diseñado los reportes correspondientes haciendo uso del Data Report incluido con Visual Basic; por lo tanto, pueden ser visualizados en pantalla e impresos si se desea:

Ejemplo de Reportes:



Descripción General del "Diccionario de Datos"

Nombre de la Entidad y sus respectivos Atributos:

Entidad: atp
 codigo_atp
 nombre_atp
 ret_employado
 ret_patrono
 limite

Entidad: cargos
 cod_cargos

CAPITULO V

VALIDACION DE RESULTADOS

5.1. PROCEDIMIENTO DE VALIDACIÓN DE RESULTADOS

Para comprobar el buen funcionamiento de la herramienta CASE diseñada se realizó una validación, procediendo de la siguiente manera:

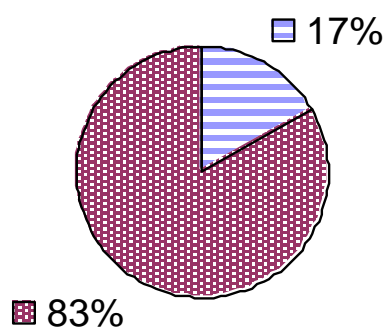
1. Se seleccionó la empresa **AVAL CARD**, la cual posee un área destinada al desarrollo de software.
2. Dentro de dicha empresa se seleccionó seis Analistas y Programadores de Sistemas, como usuarios de la herramienta.
3. Se efectuaron encuestas a dichos usuarios para conocer su metodología de programación actual.
4. Posteriormente, los usuarios realizaron una prueba de desarrollo de sistemas, utilizando la herramienta CASE como generadora.
5. A continuación se efectuaron encuestas a dichos usuarios sobre los resultados obtenidos al utilizar la aplicación “Integral CASE”, para comprobar la eficiencia en el desarrollo al utilizar la herramienta CASE.
6. Como último paso se realizó un análisis comparativo entre la encuesta previa a la prueba y la encuesta posterior a la misma.

5.2. ANALISIS DE RESULTADOS DE ENCUESTAS

Encuesta sobre Metodologías de Programación Actuales

1. ¿Qué herramientas de documentación de análisis estructurado emplea para el desarrollo de sus aplicaciones?

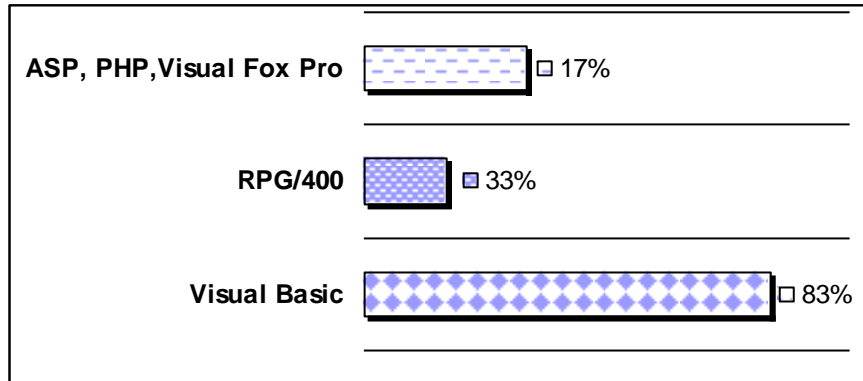
Resultado:



El 83% de los usuarios no utiliza ninguna herramienta de documentación de análisis estructurado, mientras que el 17% si utiliza alguna herramienta de documentación; como son: DFD y Diagramas E-R.

2. ¿Qué herramienta (as) de software utiliza para el desarrollo de sus aplicaciones?

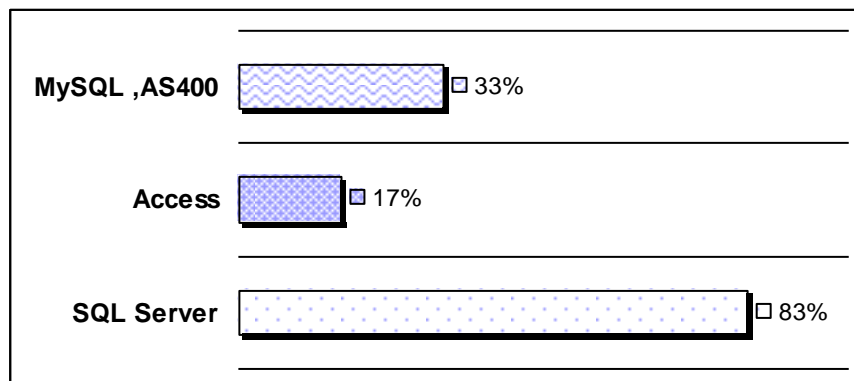
Resultado:



Entre las múltiples opciones que poseen los usuarios para el desarrollo de software el 83% utiliza Visual Basic, el 33% utiliza Visual FoxPro , ASP y PHP, mientras que un 17 % utiliza RPG/400.

3. ¿Qué gestor (es) de BDD utiliza, para el almacenamiento de su información?

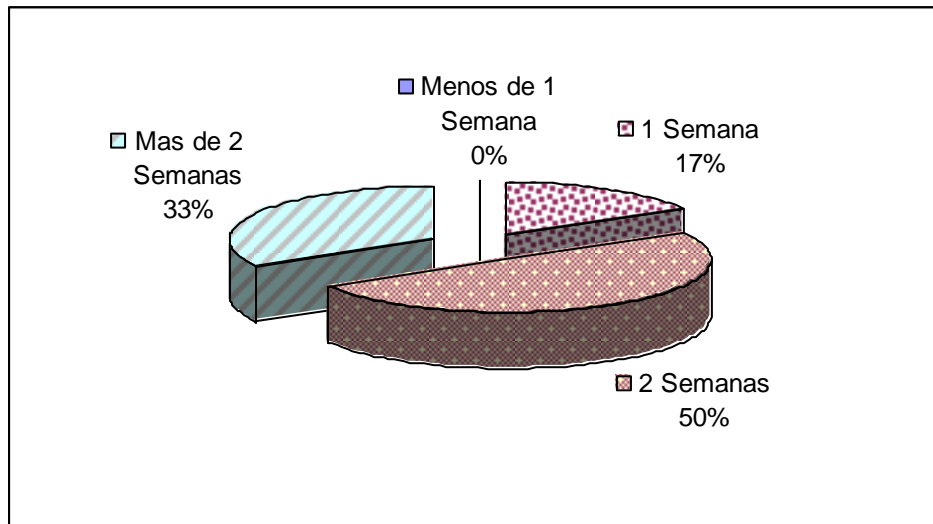
Resultado:



Entre las múltiples opciones que poseen los usuarios para el almacenamiento de la información el 83% utiliza MS SQL Server, el 33% utiliza MySQL y AS400, mientras que un 17% utiliza Access como Gestor de base de Datos

4. ¿Cuál es el tiempo promedio que emplea para el diseño de la Base de Datos y la creación de un prototipo inicial de una aplicación simple, que pueda crear Mantenimientos y Consultas?

Resultado:



El tiempo promedio que los usuarios emplean para la creación de un prototipo inicial de una aplicación y el diseño de la Base de Datos. El 50% necesita 2 Semanas, el 33% necesita más de 2 semanas, mientras el 17% necesita 1 semana.

Encuesta sobre Utilización de Integral CASE

1. De acuerdo a la utilización de “Integral CASE”, ¿Cuáles considera que son las ventajas que se obtienen con el empleo de esta herramienta, en comparación con las metodologías de programación actuales?

Resultado:

- × Optimización de tiempo empleado para el diseño de sistemas, creación de base de datos, para el desarrollo y mantenimiento de información.
- × Crea un prototipo inicial de una aplicación
- × Interfaz amigable
- × Documentación de Sistemas

2. ¿Considera que al utilizar Integral CASE se mejora la eficiencia en el desarrollo?

Resultado:

El 100% respondió que al utilizar Integral CASE mejora la eficiencia en el desarrollo de sistemas, ya que:

- × La herramienta permite la verificación de errores en el diseño de Bases de Datos.
- × Es una herramienta generadora de código que facilita la manipulación de información contenida en bases de datos.
- × Permite la optimización de tiempo.
- × Permite emitir consultas de forma rápida y sencilla.

3. Especifique las actividades de desarrollo que considera automatizables al utilizar Integral CASE:

Resultado:

- × Generar Base de Datos
- × Construir mantenimientos de Tablas
- × Generar Consultas
- × Generar Reportes
- × Generar Menús de forma automática

5.3. RESUMEN COMPARATIVO DE RESULTADOS DE VALIDACIÓN

De acuerdo con los resultados obtenidos, se verificó que la implantación de la herramienta CASE diseñada constituye un instrumento eficiente para la automatización de procesos en el desarrollo de sistemas.

Los usuarios coinciden en que su uso minimiza tiempo para el diseño y desarrollo de sistemas, en comparación con el método tradicional de programación; así como también posee una interfaz amigable, y es posible obtener documentación inmediata del diseño.

CAPITULO VI

PLAN DE IMPLANTACION

6.1. METODOLOGÍA DE IMPLANTACIÓN

La metodología de implantación sugerida para el buen funcionamiento de Integral CASE comprende cinco etapas fundamentales:

6.1.1. Plan de Mercadeo:

El proceso inicial para la implantación de Integral CASE consiste en dar a conocer la herramienta y los beneficios que pueden obtenerse con su utilización. Para ello se plantean los siguientes métodos:

- a) Promoción a través de conferencias sobre el uso de Integral CASE y su repercusión en la fábrica de software, en las cuales se proporcione una demostración práctica de la herramienta dando a conocer todos los procesos y beneficios que incluye.
- b) Publicidad a cargo de la Universidad Don Bosco, en donde se promuevan a través de los medios de comunicación los proyectos de tesis que aporten herramientas útiles a las empresas del país.
- c) Realización de visitas y demostraciones del producto al personal encargado de toma de decisiones en el área de desarrollo de software en empresas dedicadas a este rubro.
- d) Elaboración de una versión de evaluación de Integral CASE válida por 30 días, en la cual los usuarios puedan realizar pruebas de forma gratuita que les permitan conocer las ventajas que posee.

- e) Diseño de un Portal Web que muestre de forma sistemática el funcionamiento de Integral CASE y a la vez contenga el enlace para descargar la versión de evaluación de la herramienta.
- f) Distribución de afiches en empresas que posean áreas de desarrollo de sistemas, en los que se refleje la descripción general de la herramienta Integral CASE.
- g) Investigación de direcciones de correo electrónico de empresas que posean áreas de desarrollo de sistemas, para el envío de mensajes publicitarios que contengan información sobre Integral CASE.

6.1.2. Plan de Capacitación a usuarios:

Implica en primer lugar, la educación de los usuarios en cuanto al uso de herramientas CASE en general, ya que la metodología de análisis y diseño al utilizar este tipo de herramientas, difiere en cierta medida de las técnicas tradicionales de programación. Es necesario dar a conocer los diferentes tipos de herramientas CASE, los beneficios que se obtienen con su uso así como sus limitantes y hacer énfasis en que, si bien es cierto, las herramientas CASE automatizan tareas de análisis y diseño, siempre es necesaria la intervención del analista o programador en ellas.

En segundo lugar se debe capacitar a los usuarios para la utilización de la herramienta Integral CASE, haciendo uso del Manual de Usuario (Ver anexo E); en el que se detallan los diferentes procesos que la herramienta es capaz de realizar.

6.1.3. Plan de Instalación:

Para la instalación de Integral CASE debe hacerse uso del CD de instalación, el cual contiene un Asistente para Instalación, que le guiará en el proceso. Antes de proceder, debe asegurarse de que el equipo en que se instalará Integral CASE cumple los siguientes requisitos mínimos de hardware y software:

Requisitos mínimos de hardware:

- × Microprocesador Pentium de 233 MHZ. o superior (o equivalente)
- × 64 MB de RAM (como mínimo), se recomienda **128 MB** de RAM
- × 1 GB de espacio libre en el disco duro
- × Monitor VGA o Super VGA
- × Teclado
- × Mouse
- × Unidad de CD-ROM

Requisitos de software

- × El sistema deberá instalarse sobre plataforma de Windows, por lo que se requiere licencias de Windows 95/ 98 /ME /2000/NT ó XP
- × Además el usuario deberá poseer el lenguaje de programación Visual Basic para generar su aplicación en extensión VBP y hacer uso total de las funciones de la herramienta.
- × Según el gestor de Base de Datos a utilizar: deberá poseer licencia de MS SQL Server (si se desea generar bases de datos en SQL Server)

MS Access (no indispensable) o poseer el gestor de Base de Datos MySQL (se adquiere completamente gratis)

El proceso de instalación es sumamente sencillo a través de los siguientes pasos:

- × Doble clic en el archivo **Instalar.exe** que se encuentra en el CD de instalación de *Integral CASE*.
- × Seleccionar Aceptar, para iniciar el proceso de instalación.
- × Seguir los pasos del Asistente, hasta llegar a la ventana Finalizar

Una vez instalada la aplicación, para ingresar a Integral CASE, seleccione el Menú Inicio de Windows/ Programas/ Integral CASE

6.1.4. Plan de Conversión: consiste en definir todas las actividades que deben ocurrir para implantar el nuevo sistema (Integral CASE) y ponerlo en operación. Debe existir una persona encargada de identificar y asignar las tareas que pueden llevarse a cabo haciendo uso de la herramienta.

6.1.5. Revisión de resultados: luego de hacer ciertos ejercicios con la herramienta Integral CASE debe hacerse una revisión para determinar si el sistema cumple con las expectativas y evaluar la correcta utilización de la misma por parte de los usuarios.

6.2. NUEVO ENTORNO DE DESARROLLO DE SOFTWARE

Con la implantación de Integral CASE surge una nueva concepción en el entorno de la ingeniería de software:

El valor agregado indudable de utilizar Integral CASE es el aumento de productividad en las actividades soportadas por la herramienta.

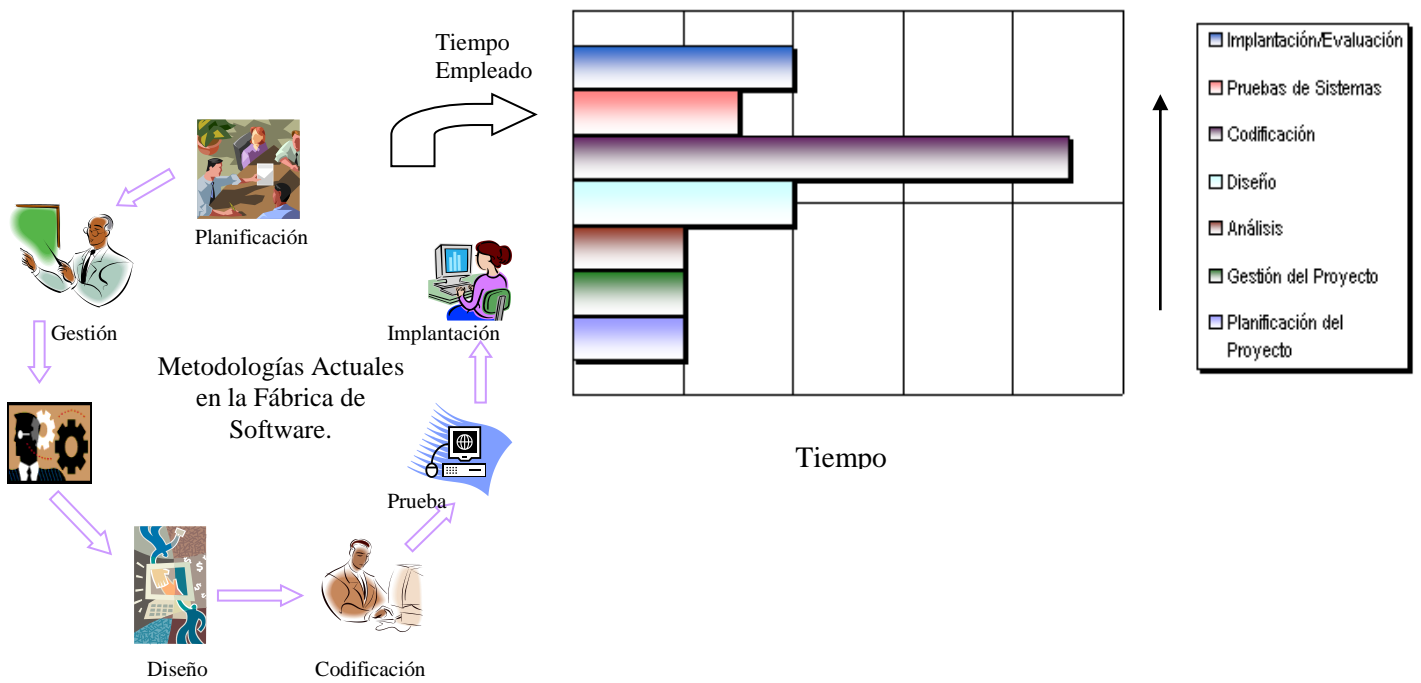
El énfasis en el análisis y diseño promovido por la herramienta tiene un fuerte impacto y recompensa en la mejora de la calidad del producto obtenido y en el aumento de productividad (disminución de tiempos, costes y esfuerzos) en las actividades de desarrollo.

El tiempo que se disminuye en estos procesos, puede ser empleado en otras áreas que de alguna manera son descuidadas pero de suma importancia, ya que de no proporcionar el tiempo adecuado terminan en repercusiones posteriores a lo largo de todo el proceso; tal es el caso de la Planificación, Gestión y Análisis del proyecto (etapas primarias del desarrollo).

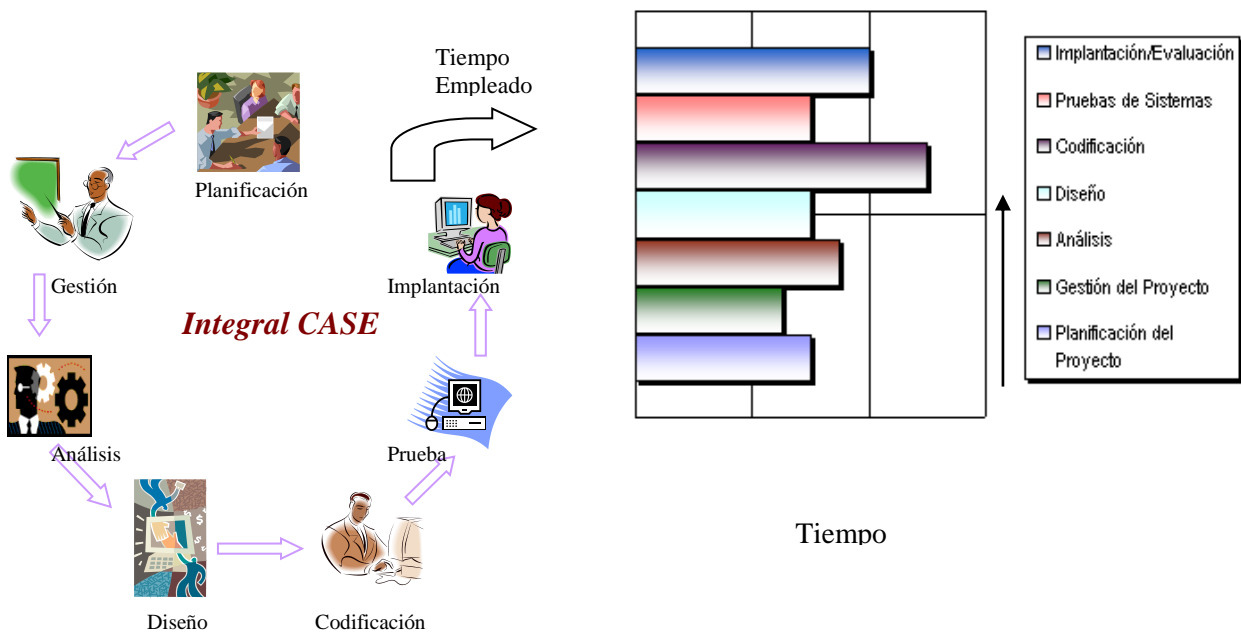
De la forma anterior, Integral CASE viene a integrarse al proceso completo que se da en una organización dedicada a la fabricación de software, contribuyendo a minimizar algunos problemas de retrasos en procedimientos que incluso pueden llevar al fracaso la finalización de los proyectos por la falta de tiempo para poder llevar a cabo un buen trabajo de planificación y análisis.

Lo anterior se basa en la comparación de las actividades realizadas previas a la implantación de Integral CASE y posteriores a ella:

ACTIVIDADES PREVIAS A LA IMPLANTACION



ACTIVIDADES POSTERIORES A LA IMPLANTACION



CONCLUSIONES

- Teniendo como base la investigación realizada, se concluye que la mayoría de las empresas e instituciones del área de San Salvador dedicadas al desarrollo de software no utilizan Herramientas CASE para crear nuevos sistemas. Por tal motivo invierten más tiempo en el diseño utilizando solamente Lenguajes de Programación, pues el empleo de Herramientas de Ingeniería de Software Asistida por Computadora (CASE) permite agilizar el proceso de Diseño de aplicaciones.
- La mayoría de Herramientas CASE que se encuentran disponibles en el mercado solamente proporcionan herramientas para diagramación y generación de bases de datos, por lo que la adquisición de Integral CASE para empresas orientadas al desarrollo de sistemas, constituye una buena inversión ya que ésta proporciona adicionalmente a las herramientas anteriores diseño de pantallas, construcción de consultas, emisión de reportes y generación de código en Visual Basic.
- La mayoría de programadores y analistas entrevistados coinciden en que no utilizan Herramientas de Diagramación para el Análisis y Diseño de aplicaciones, lo cual implica que podrían invertir mayor tiempo depurando la aplicación al intentar diseñarla en primera instancia en un lenguaje de programación, que al utilizar una herramienta de diagramación; lo cual permitiría depurarla previamente.

- La utilización de un Diccionario de Datos para el diseño de Integral CASE ha sido de vital importancia, ya que éste tiene como objetivo facilitar el control de cada uno de los objetos que se manejan en la aplicación lo que permite generar automáticamente todas las tablas definidas, pantallas, reportes, consultas y código en Visual Basic.

RECOMENDACIONES

- Se recomienda que las empresas que poseen áreas de desarrollo de software en nuestro país, se interesen cada día más en la adquisición de Herramientas CASE para realizar los procesos de diseño con mayor eficiencia.
- Incentivar a las empresas dedicadas al desarrollo de software, a utilizar Metodologías Estructuradas de Análisis y Diseño de Sistemas, como herramientas de diagramación, ya que las herramientas con representaciones gráficas y documentación mejoran la productividad en el desarrollo.
- Para que la implantación de Integral CASE sea un éxito, es necesario que los usuarios de la herramienta posean experiencia sobre análisis y diseño de sistemas, a fin de que ésta sea utilizada adecuadamente para reducir el tiempo en el desarrollo de aplicaciones eficientes.
- Para visualizar completamente los resultados de los procesos realizados por el sistema, es necesario contar con Visual Basic, y con gestores de bases de datos como Microsoft Access, Microsoft SQL Server o MySQL.

GLOSARIO TÉCNICO

Append

Agrega un nuevo objeto DAO a una colección.

Base de Datos

Una base de datos es una recopilación de información relativa a un asunto o propósito particular, como el seguimiento de pedidos de clientes o el mantenimiento de una colección de música.

Campo

Elemento de una tabla que contiene un elemento de información específico, por ejemplo un apellido, un campo se representa como una columna de una tabla en una base de datos.

Clave Externa

Uno o más campos (columnas) de una tabla que hacen referencia al campo o campos de clave principal de otra tabla. Una clave externa indica como están relacionadas las tablas; los datos en los campos clave externa deben coincidir, aunque los nombre de los campos no sean los mismos.

Clave Principal

Uno o más campos (columnas) cuyo valor o valores identifican de manera única cada registro de una tabla. Una clave principal no puede permitir valores Null (Nulo) y debe

tener siempre un índice único. Una clave principal se utiliza para relacionar una tabla con claves externas de otras tablas.

Control ActiveX

Un control ActiveX, al igual que un control integrado(los cuales aparecen en el cuadro de herramientas) es un objeto que se inserta en un formulario, informe o página de acceso a datos para permitir o mejorar la interacción entre el usuario y una aplicación.

Consulta

Se utilizan consultas para ver, modificar y analizar datos de formas diferentes. También pueden utilizarse como el origen de registros para formularios, informes y páginas de acceso a datos.

CreateDatabase

Crear un nuevo objeto Database (Base de Datos), guarda la base de datos en el disco y devuelve un objeto Database abierto (sólo espacio de trabajo Microsoft Jet).

Sintaxis:

Set basedatos = espaciodetrabajo.CreateDatabase (nombre, escenario, opciones)

CreateField

Crea un nuevo objeto Field (Campo).

Sintaxis:

Set campo = objeto.CreateField (nombre, tipo, tamaño)

CreateIndex

Crea un nuevo objeto Index (Índice).

Sintaxis:

Set índice = TableDef.CreateIndex (nombre)

CreateRelation

Crea un nuevo objeto Relation (Relación).

Sintaxis:

Set relación = basededatos.CreateRelation (nombre, tabla, tablaexterna, atributos)

CreateTableDef

Crea un nuevo objeto TableDef (Tabla).

Sintaxis:

Set TableDef = basededatos.CreateTableDef (nombre, atributos, origen, conectar)

Field (Objeto)

Un objeto Field representa una columna de datos con un tipo de datos y una serie de propiedades comunes.

ForeignName

Establece o devuelve un valor que especifica en una relación el nombre del objeto Field de la tabla externa que corresponde a un campo de la tabla principal (sólo espacios de trabajo Microsoft Jet).

Formulario

Objeto en el que se pueden colocar controles para realizar acciones o para introducir, mostrar y modificar los datos de los campos de una base de datos.

Índice

Devuelve un valor o la referencia a un valor en una tabla.

Integridad Referencial

Sistema de reglas para garantizar que las relaciones entre los registros de tablas relacionadas son válidas y que no se eliminan ni modifican accidentalmente datos relacionados.

Interfaz de programación de aplicaciones de Windows (API)

Funciones, mensajes, estructuras de datos, tipos de datos e instrucciones que se pueden utilizar para crear aplicaciones que se ejecuten bajo Microsoft Windows 95 o posterior. Las partes de la interfaz API que se utilizan mayoritariamente son los elementos de código incluidos para llamar a funciones de la API desde Windows. Incluyen declaraciones de procedimientos (para las funciones de Windows), definiciones de tipos definidos por el usuario y declaraciones de constantes.

MoveFirst(Método)

Mueven al registro primero de un objeto Recordset y lo convierten en el registro activo.

MoveLast(Método)

Mueven al registro último de un objeto Recordset y lo convierten en el registro activo.

MoveNext(Método)

Mueven al registro siguiente de un objeto Recordset y lo convierten en el registro activo.

MovePrevious (Método)

Mueven al registro anterior de un objeto Recordset y lo convierten en el registro activo.

Nulo

Valor que indica que en un campo faltan datos o son desconocidos. Los valores nulo se pueden especificar en campos cuya información se desconoce. Los campos definidos como clave principal no pueden tener un valor nulo.

ODBC (Conectividad abierta de bases de datos)

Protocolo estándar que permite a las aplicaciones que se conecten con diferentes servidores de bases de datos o archivos externos. Los controladores ODBC que utiliza el motor de base de datos Microsoft® Jet permiten tener acceso a Microsoft SQL Server y a muchas otras bases de datos externas.

La interfaz de programación de aplicaciones ODBC (API) también se puede utilizar para tener acceso a los controladores ODBC y a las bases de datos con las que se conectan sin utilizar el motor de base de datos Microsoft Jet.

OpenRecordset

Crea un nuevo objeto Recordset y lo añade a la colección Recordsets.

Sintaxis:

Para los objetos Connection y Database:

Set variable = objeto.OpenRecordset (origen, tipo, opciones, bloquear modificaciones)

RecordCount

Devuelve el número de registros accedidos en un objeto Recordset o el número total de registros de un objeto Recordset o TableDef.

Relación

Una asociación establecida entre campos comunes (columnas) de dos tablas. Una relación puede ser de Uno a Uno, Uno a Muchos y Muchos a Muchos.

Relation (Objeto)

Un objeto Relation representa una relación entre campos de tablas o consultas (sólo bases de datos Microsoft Jet).

Relación Uno a Uno

Asociación entre dos tablas en la que el valor de clave principal de cada registro de la tabla principal se corresponde con el valor del campo o los campos coincidentes de un registro y sólo uno de la tabla relacionada.

Relación Uno a Varios

Asociación entre dos tablas en la que el valor de clave principal de cada registro de la tabla principal se corresponde con el valor del campo o los campos coincidentes de varios registros de la tabla relacionada.

Reporte

Método eficaz de presentar los datos en formato impreso. Además se tiene el control sobre el tamaño y el aspecto de todo el informe, por lo que se puede mostrar la información en la manera que desea verla.

Tabla

Una tabla es una colección de datos sobre un tema específico, como productos o proveedores. La utilización de una tabla diferente para cada tema significa que se almacenan los datos sólo una vez, lo cual hace aumentar la eficacia de la base de datos, y reduce errores de entrada de datos.

Workspace

Un objeto Workspace define una sesión para un usuario. Contiene las bases de datos abiertas y proporciona mecanismos para realizar transacciones y, en espacios de trabajo Microsoft Jet admite asegurar grupos de trabajo. También controla si utiliza el motor de base de datos Microsoft Jet o ODBCDirect para tener acceso a datos externos.

BIBLIOGRAFIA

- Cornell,Gary: “Visual Basic 6.0 Manual de Referencia”, España, McGRAW – HILL, 1999.
- Martin,James: “Organización de las bases de datos”, México, PRENTICE-HALL, 1977.
- Senn, James A.: “Análisis y Diseño de Sistemas de Información”, 2º edición, México, McGRAW – HILL, 1992.
- Cruz, Luis: “Diseño de un sistema mecanizado de apoyo a la gestión de las asociaciones cooperativas de ahorro y crédito de El Salvador”, Tesis, Universidad de El Salvador, 1993
- León, Ana: “Definición del rol profesional del gestor de información en El Salvador. Investigación y estudio de casos en empresas salvadoreñas”, Tesis, Universidad Don Bosco, 1999
- Joyanes,Luis: “Microsoft Visual Basic 6.0”, 1ª edición, España, McGRAW – HILL, 1999.
- Burgos, Guadalupe: “Desarrollo de aplicaciones utilizando metodología auxiliada por herramientas CASE”, Tesis, Universidad don Bosco, 1996
- Ramalho: “Microsoft SQL Server 7 Iniciación y Referencia”, 1ª. edición, Colombia, McGRAW HILL, 1999.

Referencias de Internet:

- <http://www.amazon.com>
- <http://www.monografias.com>
- <http://www.embarcadero.com>
- <http://www.programacion.com>
- <http://www.linkoteca.sytes.net>
- <http://www.webexperto.com>
- <http://www.mysql.com>
- <http://www.dhn.mil.pe>
- <http://www.civ.cl>
- <http://www.colomhelp.8m.com/tiposdecase.html>
- www.desarrolloweb.com/articulos/981.php
- www3.uniovi.es/%7epsi/depto_psicologia/metodos/tutor.7/p3.html
- www.iscmolina.com/Herramientas%20CASE.html
- www.lafacu.com/apuntes/informatica/ inge_soft/isw1/default.htm
- ji.ehu.es/teaching/has/Traspas-html%5C1-CASE%5CApuntesCASE.htm
- www.ciat.org/doc/buen/metodologias_y_estandares_en_el_desarrollo_de_sistemas.pdf
- www.monografias.com/trabajos14/herramicase/herramicase.shtml
- www.geocities.com/nagc/case.htm
- www.informatica.uda.cl/pcontreras/HD/recursos/Hd1.pdf

A N E X O S

ANEXO A

Precios de Herramientas CASE

La siguiente tabla muestra los precios de algunas herramientas CASE en el mercado actual.

Herramienta CASE	Costos de licencias*	Fuente
ERWIN 3.5.2	\$164.95	http://www.monografias.com
Power Designer 9	\$4,208.73	
ER/Studio 4.02a	\$2,495.00	http://www.embarcadero.com
PowerBuilder Enterprise 8.0.1 Update	\$ 1,495.00	http://www.amazon.com
Powerbuilder Enterprise 8.0.1	\$2,669.99	
PowerBuilder Professional 8.0	\$1,059.99	

*Correspondientes a Enero del 2003.

ANEXO B

Cálculo del tamaño de la muestra

A la hora de determinar el tamaño que debe alcanzar una muestra hay que tomar en cuenta varios factores: el tipo de muestreo, el parámetro a estimar, el error muestral admisible, la varianza poblacional y el nivel de confianza. Por ello antes de presentar algunos casos sencillos de cálculo del tamaño muestral delimitemos estos factores.

Parámetro. Son las medidas o datos que se obtienen sobre la población.

Estadístico. Los datos o medidas que se obtienen sobre una muestra y por lo tanto una estimación de los parámetros.

Error Muestral, de estimación o standard. Es la diferencia entre un estadístico y su parámetro correspondiente. Es una medida de la variabilidad de las estimaciones de muestras repetidas en torno al valor de la población, nos da una noción clara de hasta dónde y con qué probabilidad una estimación basada en una muestra se aleja del valor que se hubiera obtenido por medio de un censo completo. Siempre se comete un error, pero la naturaleza de la investigación nos indicará hasta qué medida podemos cometerlo (los resultados se someten a error muestral e intervalos de confianza que varían muestra a muestra). Varía según se calcule al principio o al final. Un estadístico será más preciso en cuanto y tanto su error es más pequeño. Podríamos decir que es la desviación de la distribución muestral de un estadístico y su fiabilidad.

Nivel de Confianza. Probabilidad de que la estimación efectuada se ajuste a la realidad. Cualquier información que queremos recoger está distribuida según una ley de probabilidad (Gauss o Student), así llamamos nivel de confianza a la probabilidad de que el intervalo construido en torno a un estadístico capte el verdadero valor del parámetro.

Varianza Poblacional. Cuando una población es más homogénea la varianza es menor y el número de entrevistas necesarias para construir un modelo reducido del universo, o de la

población, será más pequeño. Generalmente es un valor desconocido y hay que estimarlo a partir de datos de estudios previos.

Tamaño de muestra para estimar la media de la población

Veamos los pasos necesarios para determinar el tamaño de una muestra empleando el muestreo aleatorio simple. Para ello es necesario partir de dos supuestos: en primer lugar el nivel de confianza al que queremos trabajar; en segundo lugar, cual es el error máximo que estamos dispuestos a admitir en nuestra estimación. Así pues los pasos a seguir son:

1.- Obtener el tamaño muestral imaginando que $N \rightarrow \infty$:

$$n_{\infty} = \frac{z_{\alpha/2}^2 \sigma^2}{e^2}$$

Donde:

$z_{\alpha/2}$: z correspondiente al nivel de confianza elegido

σ^2 : varianza poblacional

e: error máximo

2.- Comprobar si se cumple

$$N > n_{\infty}(n_{\infty} - 1)$$

si esta condición se cumple el proceso termina aquí, y ese es el tamaño adecuado que debemos muestrear.

Si no se cumple, pasamos a una tercera fase:

3.- Obtener el tamaño de la muestra según la siguiente fórmula:

$$n = \frac{n_{\infty}}{1 + \frac{n_{\infty}}{N}}$$

Veamos un ejemplo: La Consejería de Trabajo planea un estudio con el interés de conocer el promedio de horas semanales trabajadas por las mujeres del servicio doméstico. La muestra será extraída de una población de 10000 mujeres que figuran en los registros de la Seguridad Social y de las cuales se conoce a través de un estudio piloto que su varianza es de 9.648. Trabajando con un nivel de confianza de 0.95 y estando dispuestos a admitir un error máximo de 0,1, ¿cuál debe ser el tamaño muestral que empleemos?.

Buscamos en las tablas de la curva normal el valor de $z_{\alpha/2}$ que corresponde con el nivel de confianza elegido: $z_{\alpha/2} = \pm 1.96$ y seguimos los pasos propuestos arriba.

1.-

$$n_{\infty} = \frac{1.96^2 \cdot 9.648}{0.1^2} = 3706$$

2.- Comprobamos que no se cumple $N > n_{\infty}(n_{\infty} - 1)$, pues en este caso

$$10000 < 3706(3706 - 1); 10000 < 13730730$$

3.-

$$n = \frac{3706}{1 + \frac{3706}{10000}} = 2704$$

Tamaño de muestra para estimar la proporción de la población

Para calcular el tamaño de muestra para la estimación de proporciones poblacionales hemos de tener en cuenta los mismos factores que en el caso de la media. La fórmula que nos permitirá determinar el tamaño muestral es la siguiente:

$$n = \frac{N z_{\alpha/2}^2 P(1-P)}{(N-1)e^2 + z_{\alpha/2}^2 P(1-P)}$$

Donde

$z_{\alpha/2}$: z correspondiente al nivel de confianza elegido

P: proporción de una categoría de la variable

e: error máximo

N: tamaño de la población

Siguiendo con el estudio planteado en el punto anterior, supongamos que tratamos de estimar la proporción de mujeres que trabajan diariamente 10 horas o más. De un estudio piloto se dedujo que $P=0.30$, fijamos el nivel de confianza en 0.95 y el error máximo 0.02.

$$n = \frac{10000 \cdot 1.96^2 \cdot 0.30(1-0.30)}{(10000-1) \cdot 0.02^2 + 1.96^2 \cdot 0.30(1-0.30)} = 1678$$

ANEXO C

Formulario de Encuesta

PARTE I: Herramientas CASE

Objetivo:

Determinar la utilización de herramientas automatizadas para procesos de desarrollo de sistemas informáticos.

1. ¿Cuáles herramientas de documentación de análisis estructurado emplea para el desarrollo de sus aplicaciones?

- Diagrama de Flujo de Datos (DFD)
- Flujogramas
- Diagramas de Transición de Estados (DTE)
- Diagrama Entidad – Relación
- Otros: _____
- Ninguna

2. ¿Utiliza alguna herramienta de ingeniería de software asistida por computadora (CASE)?

- Si No

3. Si actualmente NO utiliza una herramienta CASE para el desarrollo de software ¿Estaría interesado en utilizar una? ¿Porque?

4. ¿Cuál es el nombre de la herramienta CASE que utiliza?

5. Indique si la herramienta CASE que utiliza posee los siguientes componentes:

- a) Herramientas para Diagramación
- b) Diccionario de Datos (Metadatos)
- c) Generador de Base de Datos
- d) Generador de Interfaces
- e) Generador de código
- f) Herramientas de Administración

6. Si la herramienta CASE que utiliza posee herramientas para diagramación, indique cuales de las siguientes:

- Diagrama de Flujo de Datos (DFD)
- Flujogramas
- Diagramas de Transición de Estados (DTE)
- Diagrama Entidad – Relación
- Otros: _____
- Ninguna

7. Si la herramienta CASE que utiliza incluye generadores de bases de datos, seleccione el gestor(es) con los cuales interactúa la herramienta:

- Microsoft SQL Server
- Microsoft Access
- MySQL
- Oracle
- Postgress
- Visual FoxPro
- Otros: _____

8. Si la herramienta CASE que utiliza posee generadores de código, indique el lenguaje de programación en que se realiza la codificación:
- C/C++
 - Visual Basic
 - Visual FoxPro
 - Delphi
 - Java
 - Otros: _____
9. ¿La herramienta CASE que utiliza contiene herramientas para generación de reportes?
- Si No
10. ¿Los reportes generados por la herramienta CASE muestran información cruzada a partir de tablas relacionadas?
- Si No

PARTE II: Herramientas de Desarrollo

*Nota: Por favor contestar esta parte si actualmente **NO utiliza** herramientas CASE para desarrollo de sistemas.*

Objetivo:

Conocer cuáles herramientas de desarrollo de software utilizan las empresas en el área de San Salvador

11. ¿Qué herramientas de software utiliza para el diseño de sus aplicaciones informáticas?
- Visual Basic
 - Visual FoxPro
 - Delphi
 - Java
 - Otros: _____

12. ¿Qué gestor de base de datos utiliza para almacenar y administrar la información de los sistemas que desarrolla?

- Microsoft SQL Server
- Microsoft Access
- MySQL
- Oracle
- Visual FoxPro
- Otros: _____

13. ¿Por qué razón(es) utiliza ésta(s) herramienta(s)?

- Minimiza el tiempo de desarrollo de la aplicación
- Satisfacción del usuario incrementada por el producto
- Costo accesible a usuarios
- Facilidad de uso
- Otros: _____

Formulario de Entrevista

Objetivo:

Determinar la utilización y beneficios que se obtienen con el empleo de herramientas de automatización de procesos, en el desarrollo de sistemas informáticos.

Entrevistado: _____

Nombre de Empresa: _____

Cargo: _____

1. ¿Qué concepto tiene de una herramienta CASE?

2. ¿Cuáles son los beneficios que se obtienen al utilizar una herramienta CASE?

3. ¿Qué debilidades observa en las herramientas CASE?

4. De las herramientas CASE que usted conoce, ¿cuál es la que ofrece mayores beneficios en el mercado actual? ¿Por qué?

5. ¿Considera que es más eficiente y efectivo utilizar una herramientas CASE en el desarrollo de sistemas, que realizar los procesos con el método tradicional? ¿Por qué?

6. ¿A cuánto asciende (en promedio) el costo de las herramientas CASE existentes en el mercado actual?

7. ¿Puede justificarse el costo de una herramienta CASE con los beneficios que se obtienen de ella? ¿Por qué?

8. ¿De qué manera podría garantizarse el éxito de implantación de una herramienta CASE?

ANEXO D

LISTA DE EMPRESAS ENCUESTADAS

INDUSTRIA	COMERCIO	SERVICIO	CONSTRUCCION
MOLSA	DIDEA	Banco Cuscatlán	Inversiones Roble
MOORE de Centroamérica	Grupo Q	INET	Constructora Arco Ingenieros
CETRON	Supermercado Selectos	Fync Data Computer System	Linares Constructora
Alimentos y Turismo	Supermercado La Despensa	TELECOM	
El Dorado	La Curacao	CONSISA	
Diana	Hotel Terraza	Superintendencia de Pensiones	
Industrias La Constancia		Biggest	
		CNR	
		Millennium Consulting	

ANEXO E

Manual de Usuario

Integral CASE

INDICE

Introducción	
Interfaz de usuario de Integral CASE.....	1
Entorno de diseño.....	1
Ventanas de Ingreso.....	1
Área de Diseño.....	2
Menú Principal.....	3
Archivo.....	3
Edición.....	4
Ver.....	5
Herramientas.....	6
Ayuda.....	8
Barra de Herramientas.....	9
Árbol de Proyectos.....	10
Secciones del Árbol e Proyectos.....	10
Diagrama Entidad-Relación.....	10
Pantallas.....	10
Consultas.....	13
Barra de Controles.....	15
Barra de Controles para Diagrama E-R.....	15
Barra de Controles para el Diseño de Pantallas.....	16
Estructura de la Entidad.....	17
Crear Relaciones.....	21

INTRODUCCIÓN

“Integral CASE” es una herramienta CASE orientada a automatizar las primeras fases del desarrollo de aplicaciones; inicialmente proporcionando plantillas para la creación de diagramas Entidad-Relación (E-R) utilizando el modelo físico.

A partir del diagrama E-R creado, la herramienta proporciona la posibilidad de generar automáticamente la base de datos, además el usuario tiene la opción de utilizar como gestor de bases de datos a Microsoft Access, MySQL y Microsoft SQL Server.

Una vez generada la base de datos, para manipular la información contenida en ésta, Integral CASE permite que el usuario pueda crear un prototipo inicial de una aplicación en Microsoft Visual Basic, es decir pueden desarrollarse Formularios de mantenimiento a tablas, Reportes, y Consultas.

En el presente manual se detallan todas las opciones necesarias para lograr lo anteriormente descrito, describiendo todos los elementos del Entorno de Integral CASE, así como la manera de obtener Ayuda a través de la Herramienta.

Interfaz de usuario de Integral CASE

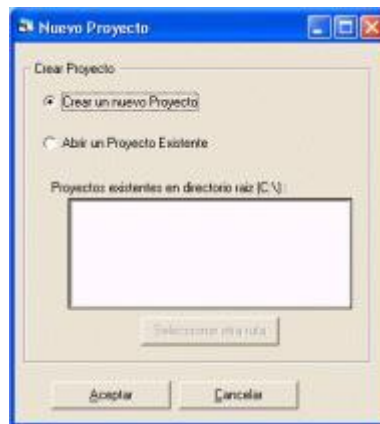
Entorno de diseño

En la interfaz de usuario se encuentran disponibles las opciones para la creación de diagramas E-R, depuración de errores en el diagrama, generación automática de bases de datos y del prototipo de la aplicación de usuario.

Ventanas de Ingreso

Crear o Abrir un Proyecto

Define el nombre con que se creará el proyecto o abrirá uno existente, Integral CASE asigna la extensión **ddp**, para identificar sus archivos de proyecto.



Al escoger la opción de Crear un Nuevo Proyecto, se activa la ventana siguiente, la cual permite elegir el Gestor de Base de Datos a utilizar en el proyecto, además para cualquiera de las dos opciones debe elegirse una ruta y un nombre de proyecto.

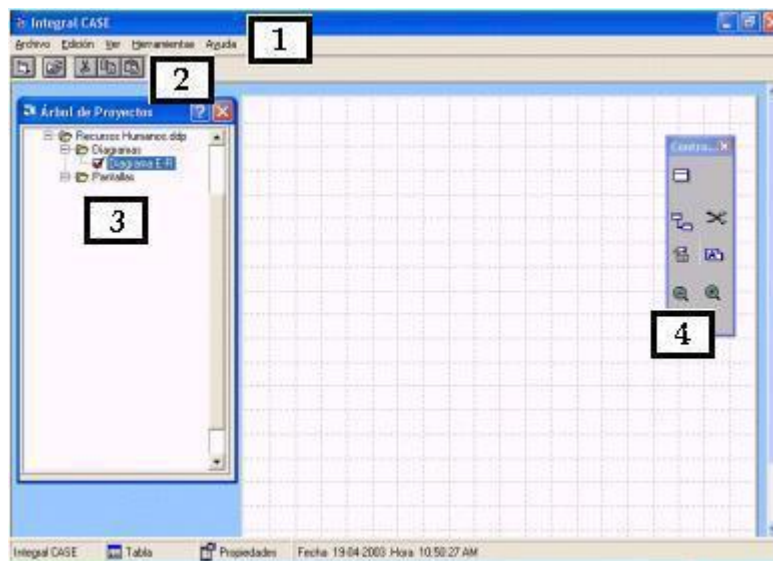


Una vez presionado el botón Aceptar, se presenta el Entorno para el diseño de diagramas Entidad-Relación de **Integral CASE**

Área de Diseño

Se encuentra constituida por:

- 1 Menú Principal
- 2 Barra de Herramientas
- 3 Árbol de Proyecto
- 4 Barra de Controles



Menú Principal

Opciones del Menú Principal:



- *Archivo*
- *Edición*
- *Ver*
- *Herramientas*
- *Ayuda*

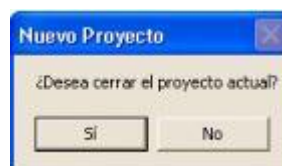
Menú archivo

El menú archivo incluye las siguientes órdenes:



- **Crear un nuevo proyecto.**

Si se encuentra abierto algún proyecto, esta opción muestra el cuadro de diálogo siguiente:



Si la respuesta a la pregunta es afirmativa se cierra el proyecto actual y a continuación se muestra el cuadro de diálogo Guardar como, en el cual se almacenará el nombre y la ruta del proyecto a crear. Si la respuesta es No, se mantiene abierto el proyecto actual.

Si no existe ningún proyecto abierto al momento de la selección de esta opción, inmediatamente se muestra el cuadro de diálogo Guardar como.

- **Abrir un proyecto existente.**

En esta opción, si se encuentra un proyecto abierto, muestra el cuadro de diálogo en el cual se pregunta si se desea cerrar el proyecto actual, si la respuesta es afirmativa, se muestra el cuadro de diálogo Abrir, en el cual se especifica la ruta y el nombre del proyecto que se desea Abrir. Si no se encuentra abierto ningún proyecto al momento de seleccionar esta opción, solamente se muestra el cuadro de diálogo Abrir.

- **Guardar y Guardar como.**

Esta orden permite guardar todos los archivos del proyecto que se encuentra abierto.

- **Imprimir.**

Permite imprimir el Diagrama E-R creado por el usuario.

Al seleccionar esta opción se abre el cuadro de diálogo Imprimir, el cual permite seleccionar la impresora y configurar el tipo de papel y su orientación.

- **Propiedades.**

Como su nombre lo indica, muestra las propiedades del proyecto actual, entendiéndose como propiedades la fecha de creación, el tamaño del archivo, etc.

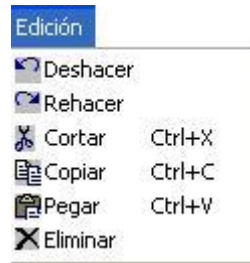
- **Cerrar.**

Esta opción cierra el proyecto actual.

- **Salir.**

Permite salir de la herramienta Integral CASE, es decir que es el equivalente de presionar el botón de la esquina derecha de la barra de título.

Menú Edición



El menú edición contiene las opciones que se detallan a continuación:

- **Deshacer.**

Orden que deshace la última acción realizada por el usuario.

- **Rehacer.**

Permite que se repita la última acción que el usuario deshizo.

- **Cortar.**

Esta opción permite eliminar texto y objetos creados, y lo coloca en el portapapeles de Windows.

- **Copiar.**

Orden que crea una copia del texto y objetos seleccionados y lo coloca en el portapapeles.

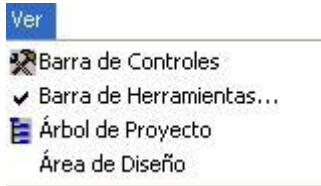
- **Pegar.**

Toma lo que haya en el portapapeles y lo pega en la aplicación actual.

- **Eliminar.**

Esta opción suprime la información y controles seleccionados, pero no coloca una copia en el portapapeles.

Menú Ver



Permite visualizar las ventanas que forman parte del entorno principal de la Herramienta Integral CASE. Contiene las opciones que se describen a continuación:

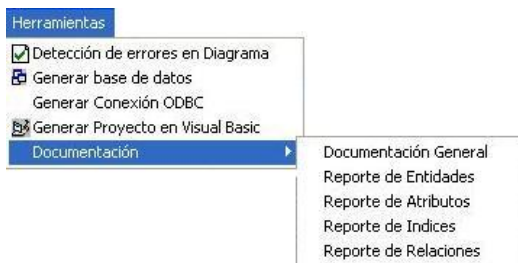
- **Barra de Controles.**

Orden que permite visualizar la ventana que contiene los controles necesarios para creación de diagramas.

- **Árbol de Proyecto.**

Permite mostrar la ventana de explorador de proyectos, es decir el árbol en el cual se detallan todos los componentes del proyecto actual.

Menú Herramientas.



El menú Herramientas contiene órdenes para la ejecución de procesos de automatización, sus opciones son las siguientes:

- **Detección de errores en diagrama.**

La validación de errores incluye un análisis de la estructura del diagrama, tanto en tiempo de creación como al ejecutar la opción de revisión de errores correspondiente.

Integral CASE determina si el diagrama E-R obedece a una estructura coherente, mediante mensajes de advertencia mostrados en pantalla durante la creación y el despliegue de un reporte con el estado actual del diagrama al ejecutar la opción de revisión.

Al ejecutar la opción de validación de errores correspondiente (en el menú principal) se verifica lo siguiente:

- Comprobar que cada entidad posea al menos un atributo
- Que a cada atributo se le haya asignado las características: tipo de dato, longitud, tipo de llave.
- Que una entidad posea al menos un atributo clave (atributo principal) que identifique a cada una de sus instancias
- Procurar que cada entidad se encuentre relacionada con una o más entidades y así evitar, en cuanto sea posible, la existencia de islas dentro del diagrama



- **Generar base de datos.**

Orden que crea la base de datos tomando como referencia el diagrama E-R diseñado y el gestor seleccionado al momento de crear el proyecto actual (debe tenerse en cuenta que solo se generará la base de datos si el diagrama E-R se encuentra libre de errores).

Es decir, esta opción permite la generación automática de base de datos de acuerdo a las especificaciones realizadas en el diagrama.

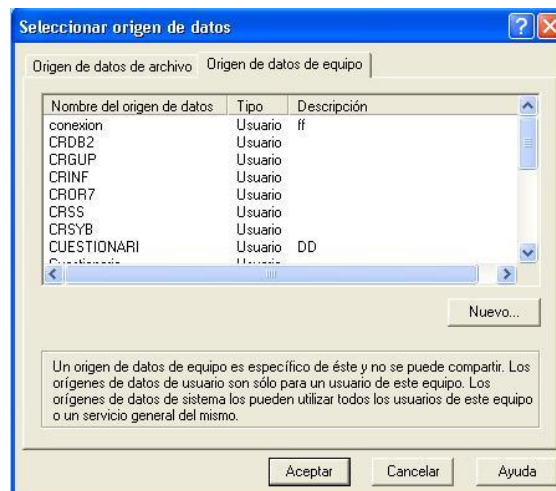
La generación de base de datos se hace atendiendo lo siguiente:

- El nombre del proyecto se asigna como nombre a la base de datos.
- Cada entidad se convierte en una tabla de la base de datos.
- Los atributos y sus propiedades (tipo de dato, longitud, tipo de llave) pasan a ser los campos de cada tabla.
- Las relaciones definidas en el diagrama E-R, constituyen las relaciones correspondientes entre tablas.

Este proceso no se realiza si se detecta, mediante el algoritmo de validación de errores, que el diagrama aún posee errores en el diseño.

- **Generar conexión ODBC.**

Opción que coloca en primer plano al asistente para crear Conexiones ODBC de Windows, en ésta se especifica el Gestor a utilizar, la base de datos y debe el nombre de la ODBC.



- **Generar proyecto en Visual Basic.**

Esta opción toma las pantallas creadas por el usuario en la Herramienta Integral CASE y genera una nueva aplicación en Visual Basic. Dicha aplicación contendrá formularios para mantenimiento a tablas, consultas y reportes sencillos de datos.

- **Documentación.**

Muestra Reportes de Datos Generales, o específicos como de las Entidades que forman parte del Diagrama Entidad-Relación, de los Atributos, Índices o reporte de todas las Relaciones en dicho Diagrama.

Menú Ayuda.



El Menú ayuda, brinda información adicional al usuario, consta de las siguientes opciones:

- **Temas de Ayuda.**

Opción que muestra toda la información concerniente al sistema, como pueden ser controles, funciones, forma de operar, etc.

- **Acerca de Integral CASE.**

Orden que muestra las especificaciones y aspectos legales de Integral CASE.

Barra de Herramientas

La barra de herramientas está compuesta por los siguientes íconos, mediante los cuales se tiene un acceso rápido a ciertas opciones del Menú Principal.



Nuevo Proyecto



Crea un nuevo proyecto definido por el usuario. Este archivo posee la extensión .ddp

Abrir un Proyecto Existente



Abre un Proyecto .ddp en la ruta especificada por el usuario.

Cortar.



Esta opción permite eliminar texto y objetos creados, y lo coloca en el portapapeles de Windows.

Copiar



Orden que crea una copia del texto y objetos seleccionados y lo coloca en el portapapeles.

Pegar



Toma lo que haya en el portapapeles y lo pega en la aplicación actual.

Deshacer



Orden que deshace la última acción realizada por el usuario

Rehacer



Permite que se repita la última acción que el usuario deshizo.

Mostrar cuadro de Controles.



Utilice esta opción para mostrar o traer a primer plano la ventana cuadro de controles para el diseño del Diagrama E-R.



Mostrar Arbol del Proyecto.

Activa la Ventana Arbol del Proyecto, si ésta se encuentra oculta.



Chequear Errores.

Activa la Ventana Chequear Errores, en donde se realiza un análisis de la estructura del Diagrama E-R.



Generar Base de Datos.

Opción que genera la Base de Datos a partir del Diagrama E-R. El tipo Base de Datos que se genere dependerá del Gestor seleccionado por el Usuario.



Mostrar Ayuda.

Muestra en primer plano la Ayuda del Sistema.

Árbol de Proyectos

La Herramienta Integral CASE organiza las aplicaciones en Proyectos, Cada Proyecto puede contener un Diagrama E-R, varias Pantallas, Reportes y Consultas. El Árbol del Proyecto aparece en la parte superior izquierda del Entorno de Integral CASE. El nombre del Proyecto es especificado por el Usuario al iniciar la aplicación.

También puede accederse al Árbol de Proyectos mediante la opción Árbol del Proyecto del Menú Ver.



SECCIONES DEL ARBOL DE PROYECTO

- **DIAGRAMAS**

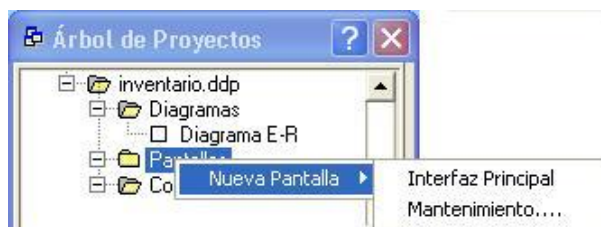
En esta sección se almacena el Diagrama E-R creado por el usuario.

- **PANTALLAS**

Contiene todas las Pantallas generadas.

Para crear una nueva Pantalla:

Clic derecho sobre la carpeta de Pantallas, a continuación aparece un Menú Emergente con dos opciones.



La opción Interfaz Principal:

Crea una Pantalla que permite crear un Menú, el cual contiene acceso al resto de pantallas a crear (Pantallas de Mantenimiento a Tablas).

Al Elegir esta opción se muestra la ventana Estilo. En donde puede elegir un fondo a su aplicación, ya sea los estilos que se muestran o accedando a un archivo de imagen que se encuentre almacenado en la unidad de disco.



Opción Mantenimiento:

Con esta opción el usuario crea Pantallas para proporcionar mantenimiento a Entidades del Diagrama E-R.

Existen dos tipos de Mantenimiento.

- De una sola Entidad
- De Entidades Relacionadas

Cualquiera de las opciones anteriores puede elegirse en la ventana siguiente:



La primera opción permite crear un Mantenimiento para una sola Entidad, la cual debe elegirse de todas las Entidades que se encuentran en el Diagrama Entidad-Relación, éstas aparecen disponibles en el Combo.

La siguiente opción habilita un segundo Combo, en el cual se especifican las Tablas que se encuentran relacionadas, con la Entidad seleccionada en el primer Combo. Una vez seleccionada una de éstas opciones, presionar el botón Aceptar para crear la Pantalla.

Eliminar Pantallas

Para eliminar una pantalla creada, selecciónela y presione clic derecho sobre ella. A continuación clic en Eliminar...



Botones de Mantenimiento a Tablas



Inicio

Muestra el primer registro de la tabla



Anterior

Mueve la posición del registro actual de la tabla, un registro hacia atrás.



Siguiente

Mueve la posición del registro actual de la tabla, un registro hacia adelante.



Ultimo

Mueve la posición del registro actual, al último registro de la tabla.



Buscar

Busca un registro en la tabla.



Nuevo

Crea un nuevo registro en la tabla.



Modificar

Acción que permite modificar un registro específico



Eliminar

Borra o Elimina un registro específico de la Base de Datos.



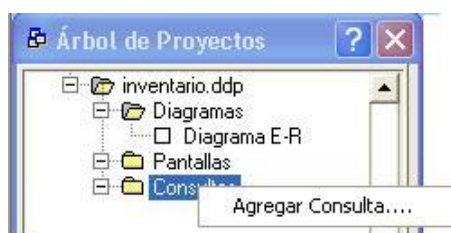
Salir

Cierra la Ventana actual.

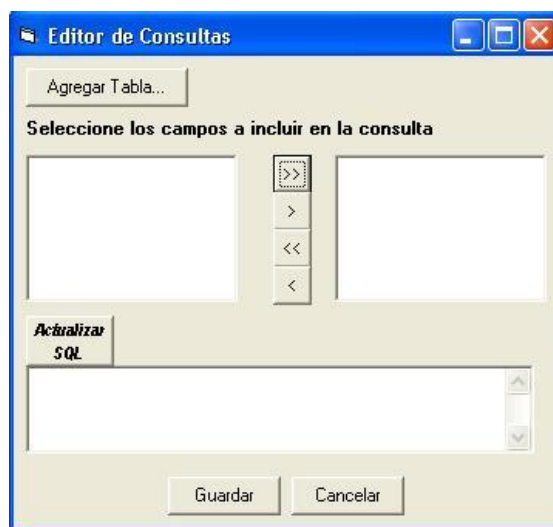
- **CONSULTAS**

Las Consultas son creadas a partir de Entidades y sus Atributos. A partir de las Consultas el usuario puede organizar la información para mostrarla en pantalla o través de un Reporte, según convenga a su aplicación.

Para crear una nueva Consulta, clic derecho sobre la carpeta Consultas del Árbol de Proyectos. Esta acción desplegará un menú, clic en **Agregar Consulta**.









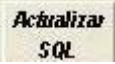
A continuación se visualizará en primer plano la ventana **Editor de Consultas**:



El cuadro de lista de la parte superior derecha muestra todos los Atributos que forman parte de las Entidades que fueron seleccionadas para formar la consulta

El cuadro de lista a la izquierda permite visualizar los Atributos que fueron seleccionados para la Consulta

Elementos:

	<p>Despliega la ventana Agregar Tabla, en la cual se muestra una lista de todas las Entidades que conforman el Diagrama Entidad-Relación y que se encuentran disponibles para crear Consultas. Una vez seleccionada la Entidad presionar el botón Agregar.</p> 
	<p>Permite seleccionar todos los Atributos incluidos para formar la Consulta.</p>
	<p>Agrega un elemento seleccionado de la Lista de Atributos disponibles para ser agregado a la Consulta.</p>
	<p>Quita un Atributo de la lista que seleccionado con anterioridad para ser parte de la Consulta.</p>
	<p>Elimina todos los Atributos seleccionados para que formen una Consulta.</p>
	<p>Permite visualizar el string de la Consulta generada, a partir de las Entidades y los Atributos seleccionados</p>

Para guardar la Consulta creada, presionar el botón **Guardar**, a continuación debe asignarle un nombre.

Barras de Controles

Barra de controles para diagramas E-R



Regresa el puntero a su forma estándar y anula la última opción elegida.



Este control permite crear una nueva entidad dentro del área de diseño de la Herramienta Integral CASE.



Permite crear relaciones entre entidades.



Modifica una relación entre entidades.



Muestra las propiedades del objeto.



Permite crear una etiqueta para que el usuario coloque información adicional dentro del área de diseño.



Incrementa en un porcentaje el tamaño de la ventana de diseño junto con sus componentes.



Reduce en un porcentaje el tamaño de la ventana de diseño junto con sus componentes.

Barra de controles para el diseño de pantallas



Regresa el puntero a su forma estándar y anula la última opción elegida.



Etiqueta, permite colocar rótulos de texto en la pantalla



Permite agregar cuadros de texto en el diseño de la pantalla



Utilizada para dibujar líneas en la pantalla



Agrega un control que muestra la hora del sistema



Permite colocar botones de comando en el diseño de la pantalla



Agrega un control que muestra la fecha del sistema



Utilizado para insertar un control combo.

Estructura de la Entidad

The screenshot shows a dialog box titled "Estructura de la Tabla". It has a close button in the top right corner. The dialog is divided into several sections. At the top, there are two input fields: "Nombre de Tabla" containing the text "Tabla2" and "Descripción". Below this, there are two main sections: "Lista de Campos" and "Indices". Each section has a list box on the left, a "Agregar" button, and a "Tomar" button. To the right of the "Lista de Campos" section, there are five input fields: "Nombre:", "Tipo:", "Tamaño:", "Regla de Validación:", "Validación Texto:", and "Tipo de Llave". To the right of the "Indices" section, there are two input fields: "Nombre:" and "Campos:". Below the "Indices" section, there are two radio buttons labeled "Ascendente" and "Descendente". At the bottom center of the dialog is an "Aceptar" button.

La Entidad se encuentra constituida por los siguientes Elementos:

- **Nombre de la Tabla**

Nombre de Tabla

Almacena el nombre que el usuario otorga a la Entidad. Si no se define un nombre específico, Integral CASE almacena el texto “Tabla” seguido de un valor auto numérico.

- **Descripción**

Descripción:

Opcional. Puede introducirse una breve descripción de las características de la Entidad que está creando.

- **Modificar Campos**

Este botón despliega la pantalla que permite Modificar Atributos ya creados a una Entidad específica, debe tenerse en cuenta que esta ventana es mostrada solo si se ha seleccionado el Atributo a modificar.

En la ventana que aparece en primer plano se muestran los datos introducidos al Atributo seleccionado, para que puedan ser modificados.



The image shows a dialog box titled "Agregar Campo" with a close button in the top right corner. It contains several input fields: "Nombre:" (text), "Descripción:" (text), "Tipo:" (dropdown menu), "Tamaño:" (text), "Regla de Validación:" (text with a small icon to the right), and "Mensaje de Validación:" (text). Below these is a section for "Tipo de llave:" with two checkboxes: "Principal" (unchecked) and "Permitir Valores Nulos" (checked). At the bottom are two buttons: "Aceptar" and "Cerrar".

Una vez realizados los cambios, Presionar el botón Aceptar para que éstos sean almacenados.

- **Agregar Campos**

Para agregar Atributos a una Entidad, Presionar el botón **Agregar Campos**. Esta acción envía en primer plano la Pantalla:



This is an identical screenshot of the "Agregar Campo" dialog box as shown above, featuring the same fields for Name, Description, Type, Size, Validation Rule, Validation Message, Key Type (with "Permitir Valores Nulos" checked), and "Aceptar"/"Cerrar" buttons.

En donde:

Nombre : Nombre que se proporciona al atributo, el cual forma parte de la Entidad.

Descripción: La descripción del campo es opcional. Ayuda a describir las características del campo que está creando.

Tipo : Determina la clase de valores que el usuario puede guardar en el campo. Los tipos de valores permitidos varían de acuerdo al gestor de base de datos que se haya seleccionado.

Tamaño : Máximo número de caracteres que se puede introducir en el campo.

Regla de Validación : Expresión que limita los valores que pueden introducirse en el campo.

Para introducir una expresión al Atributo, clic sobre Regla de Validación, esta acción desplegará la ventana:



Luego de haber descrito la expresión que se aplicará al atributo, presionar clic en Aceptar.

Mensaje de Validación : Es el Mensaje de Error que aparece cuando se introduce un valor prohibido por la regla de validación

Tipo de llave:

Principal

Identificador único para cada registro de una tabla. Cada tabla debe incluir un campo o conjunto de campos que únicamente identifiquen cada registro almacenado en la tabla.

Permitir Valores Nulos

Determinar si el atributo que se está creando permitirá valores nulos.

- **Eliminar Campos**

Para Eliminar un Atributo, éste debe ser seleccionado de la lista que muestra los Atributos correspondientes a la Entidad.

- **Modificar Índices**

Para modificar un Índice, es necesario seleccionar el nombre del Índice a modificar y a continuación presionar el botón Modificar Índice, el cual despliega la ventana Índice, con los elementos del Índice seleccionado. Una vez realizados los cambios, presionar el botón Aceptar, para que éstos sean almacenados.



- **Agregar Índices**

Esta opción permite agregar un nuevo índice a la Entidad, para ello despliega la **Ventana Índices**



Elementos de la Ventana Índices:

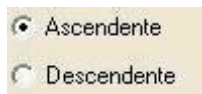
Nombre : Nombre que hace referencia al índice que se está creando.

Campos Indexados : En esta sección se muestra una lista de los Atributos que forman parte del Índice que se está creando.

Campos Disponibles : Lista de todos los Atributos que forman parte de la Entidad, y que se encuentran disponibles para ser agrupados en el Índice.



Opción deshacer. Permite eliminar de Campos Indexados, el Atributo que se haya seleccionado



Mediante estas opciones puede especificarse el orden del Índice (Ascendente o Descendente). Por defecto se presentan los índices en un Orden Ascendente.

- **Eliminar Índices**

Mediante esta acción puede eliminar un índice creado con anterioridad, para realizar este proceso, seleccione el índice a borrar y luego clic en **Eliminar Índices**.

Crear Relaciones

Para crear una relación, es necesario que existan Entidades creadas con anterioridad.

En cuanto a las Entidades a relacionar, ambas deben poseer al menos un atributo con las mismas características de tipo y longitud.

Al presionar el botón relación que se encuentra en la barra de controles para el diseño de Diagrama Entidad-Relación

La ventana que se visualiza en primer plano es la que se presenta a continuación



En el primer y segundo combo de la parte superior de la ventana se muestran todas las Entidades creadas en el diagrama Entidad-Relación.

En los combos de la parte inferior, deben escogerse los atributos que se desean relacionar, de acuerdo a las Entidades seleccionadas como Padre e Hija.

Además debe determinarse si la relación a crear será de Uno a Uno o de Uno a Muchos y el tipo de Integridad que se aplicará.