

REPOSITORIO ACADÉMICO DIGITAL INSTITUCIONAL

Sistemas de evaluación para exámenes de calidad

Autor: Elías Domínguez Maldonado

**Tesis presentada para obtener el título de:
Ing. en Sistemas Computarizados [sic]**

**Nombre del asesor:
Héctor Guerrero Guadarrama**

Este documento está disponible para su consulta en el Repositorio Académico Digital Institucional de la Universidad Vasco de Quiroga, cuyo objetivo es integrar, organizar, almacenar, preservar y difundir en formato digital la producción intelectual resultante de la actividad académica, científica e investigadora de los diferentes campus de la universidad, para beneficio de la comunidad universitaria.

Esta iniciativa está a cargo del Centro de Información y Documentación "Dr. Silvio Zavala" que lleva adelante las tareas de gestión y coordinación para la concreción de los objetivos planteados.

Esta Tesis se publica bajo licencia Creative Commons de tipo "Reconocimiento-NoComercial-SinObraDerivada", se permite su consulta siempre y cuando se mantenga el reconocimiento de sus autores, no se haga uso comercial de las obras derivadas.





UNIVERSIDAD VASCO DE QUIROGA

LICENCIATURA EN SISTEMAS COMPUTARIZADOS

SISTEMA DE EVALUACION
PARA EXAMENES DE CALIDAD

TESIS

QUE PARA OBTENER EL TITULO DE
LICENCIADO EN SISTEMAS COMPUTARIZADOS

PRESENTA
ELIAS DOMINGUEZ MALDONADO

CLAVE 16PSU0014Q
ACUERDO 952006

ASESOR DE TESIS:
ING HECTOR GUERRERO GUADARRAMA



MORELIA, MICH.

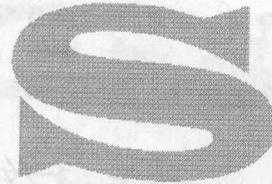
DICIEMBRE DE 1996



M.R.

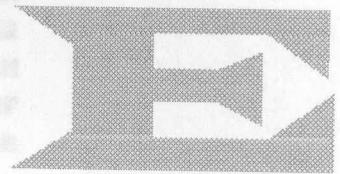
UNIVERSIDAD VASCO DE QUIROGA

LICENCIATURA EN SISTEMAS COMPUTARIZADOS



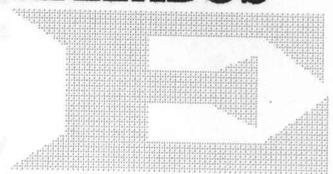
SISTEMA DE EVALUACION PARA EXAMENES DE CALIDAD

TESIS

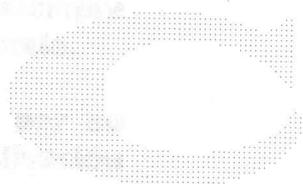


QUE PARA OBTENER EL TITULO DE

LICENCIADO EN SISTEMAS COMPUTARIZADOS



PRESENTA



ELIAS DOMINGUEZ MALDONADO

CLAVE 16PSU0014Q
ACUERDO 952006



ASESOR DE TESIS: ING. HECTOR GUERRERO GUADARRAMA

MORELIA, MICH.

DICIEMBRE, 1996

AGRADECIMIENTOS

Muchas personas han contribuido en mi preparación, me han brindado gran parte de su tiempo y apoyo, me han motivado constantemente hacia una superación personal y académica, siempre con la idea que fuese una persona útil e íntegra.

Es por eso que les agradezco infinitamente a mis padres, que fueron mi guía y mi motivación, que desde que me dieron la vida siempre me han dado su amor, su comprensión y su apoyo y que sin escatimar en desvelos y consejos han hecho de mí una persona autosuficiente.

A todos mis maestros que han contribuido para que llegara a esta etapa de mi formación como una persona capaz de ayudar y brindar mis conocimientos con ética profesional.

A Faby, mi novia, por apoyarme con cariño y entusiasmo en todo lo que estuvo a su alcance.

A mi tía Margarita, por brindarme siempre sus consejos y ayuda desinteresadamente.

A mis hermanos Mary, Ara y Dani, por su amor y confianza y por ser una motivación más para alcanzar lo que se desea.

Y al Ing. Hector Guerrero, por brindarme sus conocimientos, paciencia y amistad como profesor en la licenciatura y como asesor en la tesis.

INTRODUCCIÓN..... i

PRELIMINARES..... 1

ANTECEDENTES 1

OBJETIVOS..... 2

METAS 3

LOS SISTEMAS COMPUTARIZADOS 4

EN LA BÚSQUEDA DE SISTEMAS DE CALCULO 4

LA COMPUTACIÓN 5

POR QUE UN SISTEMA COMPUTARIZADO..... 7

ESTRATEGIAS PARA EL DESARROLLO DE SISTEMAS..... 8

Características de las estrategias: 9

Ciclo de vida clásico del desarrollo de sistemas 11

Método de desarrollo por análisis estructurado..... 12

Elección del método para el desarrollo del S.E.E.C...... 13

ESTUDIO Y ANÁLISIS DEL SISTEMA..... 15

IDENTIFICACIÓN DE LAS CARACTERÍSTICAS DEL SISTEMA 15

Solicitud del proyecto 16

Características de los exámenes 16

Características del cuestionario: 19

Importancia del problema..... 19

INVESTIGACIÓN PRELIMINAR..... 20

DETERMINACIÓN DE LOS REQUERIMIENTOS DEL SISTEMA..... 23

Funciones del sistema..... 23

Determinación de los datos de entrada y salida..... 25

Recopilación y análisis de hechos..... 28

 Entrevistas 28

 Cuestionarios 31

 Árboles de decisión 32

 Español estructurado o pseudocódigo..... 35

Categorías de usuarios 37

 Identificación de usuarios del S.E.E.C. 38

 Ubicación de categorías de los usuarios del S.E.E.C. 39

DISEÑO DEL SISTEMA..... 41

CARACTERÍSTICAS DEL DISEÑO 42

Preparando el diseño lógico 42

Accesibilidad de la información..... 42

 o Predeterminar información 43

 o Facilitar el acceso a la información y a los procesos 45

 o Eliminar la necesidad de procesamiento..... 45

Complejidad..... 46

 • Simplificación 46

- División 46
- Cambios en la secuencia 47
- UN SISTEMA FÁCIL DE UTILIZAR 47**
 - Proporcionar especificaciones detalladas para el desarrollo de software* 48
 - Estándares de diseño* 49
 - Estándares para datos.- 49
 - Estándares de codificación.- 50
 - Estándares estructurales.- 51
 - Estándares de documentación.- 52
- ELEMENTOS DEL DISEÑO..... 52**
- INICIANDO EL PROCESO DEL DISEÑO FÍSICO O SOFTWARE..... 63**
 - Fundamentos del diseño fisico* 67
 - Abstracción 67
 - Refinamiento 75
 - Modularidad..... 76
 - Arquitectura del software 77
 - Jerarquía de control 80
 - Estructura de datos 82
- DISEÑO MODULAR 82**
 - Diseño de la entrada* 91
 - Diseño de la salida* 95
 - Diseño de salidas en pantalla..... 95
 - Diseño de las salidas impresas..... 95
 - Diseño de archivos* 97
 - Diseño de interacciones con las bases de datos*..... 98
 - Diseño de Controles*..... 100
 - Diseño de Procedimientos*..... 101
 - Procedimientos para entrada de datos*..... 101
 - Procedimientos durante la ejecución* 104
 - Procedimientos para el manejo de errores*..... 105
 - Procedimientos de seguridad y respaldo*..... 106
- DESARROLLO DEL SOFTWARE..... 109**
- CONCLUSIONES Y RECOMENDACIONES 116**
- APENDICE (Codigo fuente) 118**
- BIBLIOGRAFIA 150**

INTRODUCCIÓN

Al observar a nuestro alrededor podemos darnos cuenta que la tecnología computarizada se encuentra en la mayoría de los lugares que frecuentamos cotidianamente, en los centros comerciales, centros médicos, servicios públicos como: luz, agua, telecable, bancos, instituciones educativas e incluso en los hogares.

¿De donde viene esto? Se ha convertido en una pregunta obligada al estar adquiriendo o utilizando los servicios que dicha tecnología nos ofrece. En estos últimos años hemos visto ampliarse explosivamente la variedad de opciones para satisfacer cada necesidad. Aumentó también la variabilidad de niveles y usos específicos. Es el avance científico y tecnológico que constantemente nos ayuda a realizar nuestros objetivos con mayor facilidad y perfección.

La tecnología computarizada se encuentra creciendo a pasos gigantescos, que dentro de poco las revistas ofrecerán anuncios o inserciones personalizadas, algunos transportes estarán equipados con teléfonos y pantallas interactivas conectadas a satélites para programar la compra y la entrega de productos y servicios al llegar a nuestro destino o al llegar a nuestros hogares. Los hornos de microondas, las videocaseteras y los televisores responderán a nuestros comandos hablados, teniendo acceso a decenas de cientos de canales de televisión, muchos de los cuales serán sistemas interactivos haciendo posible el que los usuarios puedan ordenar un producto o servicio con el sólo "toque" de un botón, en fin un ambiente tecnológico que seguirá modificando nuestras formas de vida.

La pregunta es si: ¿Estamos preparados culturalmente para recibir y desarrollar eficientemente estos elementos?. Además de la tecnología computarizada se requiere de profesionistas que desempeñen con calidad sus funciones y así mismo obtener mejores resultados día con día.

Es por eso que la necesidad de tener profesionistas con mayor capacidad y preparación ha hecho que diferentes organismos cuenten con diversos programas de evaluación para los profesionistas, uno de estos organismos es el Centro Nacional de Evaluación para la Educación Superior, A.C. (CENEVAL), quien ha desarrollado un Examen General de Calidad Profesional en Contaduría (EGCP-C) y que tiene como propósito brindar información que apoye la preparación correspondiente; se trata de un prueba de cobertura nacional que evalúa el nivel de conocimientos y aptitudes académicas de los recién egresados de la licenciatura en contabilidad y que con los resultados que se obtienen, los sustentantes del EGCP-C conocen sus niveles de habilidades y conocimientos, que coadyuva para que las instituciones educativas evalúen los resultados de su proceso de enseñanza aprendizaje.

Con el fin de apoyar éste proceso, la Universidad Vasco de Quiroga en las licenciaturas de Administración y Contabilidad, determinó llevar acabo Exámenes de Calidad de forma periódica, en los cuales podrán detectar los niveles de conocimientos y de habilidades que tienen los alumnos de éstas áreas, logrando identificar los puntos en los que hay que reafirmar el proceso de enseñanza, para que los alumnos logren alcanzar un nivel mas alto.

La importancia de los exámenes, es detectar en el alumno la calidad de conocimientos que éste ha adquirido en el transcurso de su carrera, así el alumno puede reconocer qué áreas domina y cuáles no, y así rectificar y empeñarse en las que así lo requieran, formando un profesionista de primer nivel, como lo demanda nuestra sociedad, además de abrirse las puertas para laborar en el extranjero, si así lo desea, ya que el nivel será de competitividad; cumpliéndose con las metas y objetivos de la aplicación de exámenes de calidad, que la Universidad se ha propuesto llevar acabo.

Los exámenes de Calidad se conforman con las licenciaturas de Administración y Contabilidad en las áreas de: Finanzas, Costos, Fiscal, Fundamentos de Administración, Mercadotecnia, Auditoría, Contabilidad, Administración de Personal, entre otras.

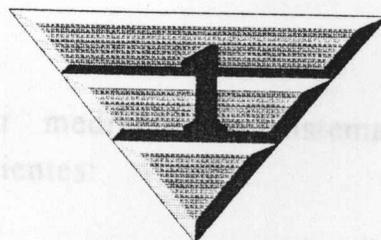
PRELIMINARES

Para lograr tanto metas como objetivos se desea hacer uso de la tecnología computarizada, facilitando y agilizando las acciones, además de perfeccionar el desarrollo de los exámenes como la aplicación y la obtención de resultados. Es por ello que un sistema computarizado de evaluación que lleve acabo la realización y el control de los exámenes que se aplicarán a los alumnos, facilitará el trabajo y se obtendrá un mejor desempeño así como una eficiencia en tiempo, ya que el sistema será orientado hacia la agilización y creación de dichos exámenes por medio de bases de datos, que servirán como antecedente de documentación y cuestionamientos en las áreas antes mencionadas, además de obtener un resultado eficientemente y con la ponderación que el profesor crea conveniente a los cuestionamientos.

El sistema evaluará el rendimiento de los sustentantes con respecto a los conocimientos básicos y necesarios de las licenciaturas, es decir, evaluará el dominio que poseen de los conocimientos fundamentales de sus disciplinas y será aplicable a alumnos entre séptimo y noveno semestre.

Dicho sistema estará sujeto por estructuras que contengan desde formatos de captura, procedimientos de confiabilidad de evaluación, disponibilidad de opciones o de guión en las áreas a evaluar, captura de resultados, obtención de reportes tanto de revisión como de enunciados, reportes de respuestas y reportes de evaluación final, entre otras.

PRELIMINARES



ANTECEDENTES

Una de las funciones de las instituciones educativas es proporcionar profesionistas con un alto grado de calidad y con un criterio amplio; es por ello que aproximadamente a mediados del 1993 surgió una opción mas con fundamentos para lograr estos principios. Fue creada por la Asociación Nacional de Facultades y Escuelas de Contabilidad y Administración (ANFECA), que pretende llevar a cabo la tarea de evaluar cada tres generaciones a los alumnos egresados de las áreas correspondientes, con el objetivo de identificar el grado de calidad de los alumnos egresados y logrando una integración de profesionistas en una coexistencia competitiva.

De esta práctica de evaluaciones creada por la ANFECA, la Universidad Vasco de Quiroga tomó la esencia, tratando de aplicarla a sus alumnos de las licenciaturas de Administración y Contabilidad, es por eso que los directivos de ésta escuela se propusieron planear e implantar evaluaciones que ayuden tanto a los alumnos como a los profesores a identificar el grado de calidad con el cual las generaciones que cursan estas licenciaturas puedan detectar el nivel de desempeño y que sirva de retroalimentación para poder corregir las fallas detectadas.

En la actualidad la dirección de Administración y Contabilidad ha optado por llevar acabo esta práctica de exámenes, los cuales se realizarán con el apoyo de las ciencias computarizadas generando un "Sistema de Evaluación para Exámenes de Calidad"; que cubrirá las expectativas deseadas.

OBJETIVOS

Los objetivos que se pretenden alcanzar mediante un sistema computarizado de exámenes de calidad, son los siguientes:

Objetivo General:

Realizar un sistema computarizado integrando los procesos de interactividad y emisión de reportes para la evaluación de exámenes de calidad en las áreas de Administración y Contabilidad de la Universidad Vasco de Quiroga.

Objetivos Específicos:

1. Ofrecer un sistema interactivo y formal con los conceptos esenciales relativos a las áreas de evaluación.
2. Facilitar a los profesores la creación, rectificación, evaluación y emisión de los exámenes.
3. Evaluar en el tiempo indicado sin tener demoras de resultados.
4. Proporcionar mayor versatilidad en la aplicación de los exámenes.

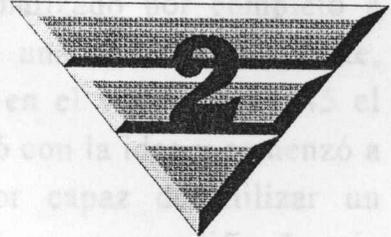
METAS

LOS SISTEMAS COMPUTARIZADOS

Las metas que se pretenden cumplir con la elaboración del sistema son:

1. Proporcionar la sistematización de los exámenes de calidad.
2. Reducir considerablemente el tiempo para la creación y la evaluación de los exámenes (85% aproximadamente).
3. Conocer los resultados de las evaluaciones automáticamente al finalizar el examen.
4. Generar reportes de los exámenes así como del resultado de las evaluaciones.
5. Que la interactividad del sistema facilite las evaluaciones y la operación del mismo.

LOS SISTEMAS COMPUTARIZADOS



EN LA BÚSQUEDA DE SISTEMAS DE CALCULO

La idea de utilizar medios mecánicos o semiautomáticos para realizar cálculos aritméticos es muy antigua. El origen del ábaco se pierde en la antigüedad. Blaise Pascal, en el siglo XVII, construyó un mecanismo para realizar operaciones aritméticas. Gottfried Willheim von Leibnitz, uno de los creadores de la lógica matemática, coinventor del cálculo infinitesimal, desarrolló un programa para lo que podría llamarse pensamiento automatizado.

El primer hombre que ideó un computador general, completo y provisto de un esquema de programación flexible y de unidades de memoria, fue Charles Babbage, de Inglaterra, que en 1833 diseñó la máquina analítica. Babbage pasó el resto de su vida y gastó gran parte de su fortuna tratando de construirla.

Entre los contribuyentes más importantes a la tecnología de la computación moderna se encuentra el Ing. eléctrico J. Presper Eckert, el físico John W. Mauchly, y uno de los matemáticos más importantes de este siglo, John Von Neumann.

En 1944, Eckert y Mauchly trabajaban en el desarrollo de una máquina denominada ENIAC, iniciales de Electronic Numerical Integrator and Computer. Diseñada para calcular tablas de disparos de artillería para el Army Ordnance Department, el ENIAC fue finalizado por completo a finales de 1945. Estaba diseñado para realizar una sucesión diferente, había que cambiar sus conexiones. Al conocer, en el verano de 1945 el proyecto del ENIAC, Von Neumann se entusiasmó con la idea y comenzó a desarrollar el diseño lógico de un computador capaz de utilizar un programa almacenado; un programa que pudiera ser modificado sin necesidad de cambiar los circuitos de la computadora. Se habla generalmente del año de 1946 como el comienzo de la era moderna del ordenador digital.

LA COMPUTACIÓN

Todo los intentos que se realizaron por obtener los medios adecuados y los instrumentos apropiados que facilitaran las tareas que desarrollaba el hombre, lo llevaron a la computación.

Actualmente, las computadoras son instrumentos de trabajo casi imprescindibles en algunas esferas de actividad humana. Se utiliza en todos los ámbitos: en la administración pública, en las pequeñas y grandes empresas, e incluso en el hogar. Nos ayudan a regular el tráfico de las grandes ciudades, a dirigir una cadena de montaje, explorar clínicamente a un paciente, y a un sinfín de tareas más.

En la forma más general una computadora es una máquina que recibe información, la elabora por medio de programas y sistemas y proporciona unos resultados. Su propiedad más característica consiste en tratar la información a gran velocidad y con dispositivos auxiliares en almacenar grandes volúmenes de información. Tareas que a un ser humano le ocuparían meses e incluso años de trabajo, la computadora las hace en unos momentos.

Los modos de indicar a la computadora cómo debe procesar los datos que recibe se denominan programas y sistemas.

Un programa es un conjunto de instrucciones que la computadora entiende y que le indican cómo debe computar los datos. Los programas constituyen el software de la computadora la cual es el hardware.

- Se conoce como software al conjunto de programas que el fabricante ofrece, para facilitar el uso de la computadora, auxilian en todas las áreas de trabajo especializándose éste en cada una de ellas; existen dos tipos, software del sistema, que es el que realiza funciones comunes para todos los usuarios y software específico, utilizado en aplicaciones particulares del usuario.

- El hardware son todos los elementos físicos necesarios para poder escribir, almacenar, y ejecutar los programas que componen el software, estos son monitores, unidad central de procesamiento (UCP), discos duros, lectoras de discos ópticos, impresoras, graficadores, etc.

Un sistema es un conjunto de componentes, cuyo comportamiento depende tanto del comportamiento de cada uno de sus partes como de la forma en que éstas interactúan entre sí.

Cuando los componentes principales se encuentran dirigidos a programas que interactúan entre sí con la finalidad de obtener resultados de los datos que se operan se le denomina sistema de información.

POR QUE UN SISTEMA COMPUTARIZADO

El comienzo de la evolución tecnológica proporciona datos fehacientes sobre la importancia del tratamiento de la información en el entorno del hombre del siglo XX. En las últimas décadas se han producido y continúan produciéndose mejoras tecnológicas en el campo informático a una velocidad que ninguna ciencia anterior había desarrollado.

El vertiginoso avance de las investigaciones ha producido en multitud de ocasiones el desconcierto en muchos usuarios de computadoras, que no han tenido tiempo de asimilar las nuevas técnicas debido a la velocidad con que éstas aparecen. Esta situación origina una inadecuada e insuficiente utilización de las computadoras.

Son abundantes los documentos, revistas libros y congresos sobre el tema para que una persona pueda, no sólo asimilar, sino tener acceso a toda la información que surge. Los cambios tecnológicos son precursores de las sucesivas generaciones de computadoras, además y con gran importancia del software, que trata de cumplir satisfaciendo las necesidades de los usuarios en forma general, por áreas y en forma especializada.

El desarrollo de un sistema experto o especializado incorpora el conocimiento de personas expertas en un tema, estando este conocimiento representado como estructura de información procesable de forma separada del procedimiento para manipularlo.

El preguntarse por que dirigir un sistema para satisfacer las necesidades de los usuarios, nos lleva a comentar algunas de las ventajas que ofrece un sistema:

Versatilidad de aplicaciones conjuntas.

Ahorro de tiempo al no tener la necesidad de ocupar un programa para cada necesidad en forma independiente.

Mayor facilidad de operatividad.

Ahorro de espacio de almacenamiento al tener todas las aplicaciones integradas.

Mejora de los cálculos al integrar los procesos dentro de un mismo ambiente.

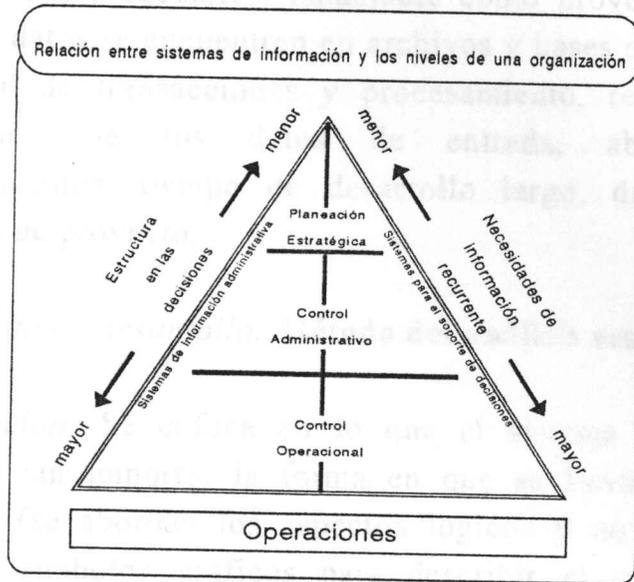
Resultados exactos y en menor tiempo que por programas independientes.

Mejor aprovechamiento de los recursos del hardware.

El obtener las herramientas conjuntas para satisfacer las diversas necesidades del usuario hace preferir un sistema, aunque el desarrollo de un sistema por lógica es mas complejo, ya que requiere de un análisis mas profundo por el hecho de ser un conjunto de necesidades que cubrir.

ESTRATEGIAS PARA EL DESARROLLO DE SISTEMAS

Los sistemas de información basados en computadora sirven para diversas finalidades que van desde el procesamiento de transacciones de una empresa, hasta proveer de la información necesaria para decidir sobre asuntos que se presentan con frecuencia, asistencia a los altos funcionarios con la formulación de estrategias difíciles y la vinculación entre la información de las oficinas y los datos de toda la corporación. En algunos casos los factores que deben considerarse en un proyecto de sistemas de información, tales como el aspecto más apropiado de la computadora o la tecnología de comunicaciones que se va a utilizar, el impacto del nuevo sistema sobre los empleados de la empresa y las características específicas que el sistema debe tener, se pueden determinar de una manera secuencial. En otros casos, debe ganarse experiencia por medio de la experimentación conforme el sistema evoluciona por etapas.



Todas las situaciones están representadas por tres distintos enfoques al desarrollo de sistemas por computadora:

1. Método del ciclo de vida para el desarrollo de sistemas.
2. Método del desarrollo del análisis estructurado.
3. Método del prototipo de sistemas.

Características de las estrategias:

Estrategia de desarrollo: **Método del ciclo de vida para el desarrollo de sistemas**

Descripción: Incluye las actividades de investigación preliminar, determinación de requerimientos, diseño del sistema, desarrollo del software, prueba de sistemas e implantación.

Características de aplicación: Requerimientos del sistema de información predecibles, manejable como proyecto, requiere que los datos se encuentren en archivos y bases de datos, gran volumen de transacciones y procesamiento, requiere de la validación de los datos de entrada, abarca varios departamentos, tiempo de desarrollo largo, desarrollo por equipos de proyecto.

Estrategia de desarrollo: **Método del análisis estructurado.**

Descripción: Se enfoca en lo que el sistema o aplicación realizan sin importar la forma en que se llevan a cabo su función (se abordan los aspectos lógicos y no los físicos). Emplea símbolos gráficos para describir el movimiento y procesamiento de datos. Los componentes importantes incluyen los diagramas de flujo de datos.

Características de aplicación: Adecuado para todo tipo de aplicaciones, Mayor utilidad como complemento de otros métodos de desarrollo.

Estrategia de desarrollo: **Método del prototipo de sistemas.**

Descripción: Desarrollo iterativo o en continua evolución donde el usuario participa directamente en el proceso.

Características de aplicación: Condiciones únicas de la aplicación donde los encargados del desarrollo tienen poca experiencia o información, o donde los costos y riesgos de cometer un error pueden ser altos, Asimismo, útil para probar la factibilidad del sistema, identificar los requerimientos del usuario, evaluar el diseño de un sistema o examinar el uso de una aplicación.

¿Qué método de desarrollo es el más apropiado?

No existe ningún método correcto para desarrollar un sistema de información, pero sí existen diferentes formas para producir el sistema correcto para una aplicación. Algunos métodos tienen más éxito que otros y esto depende de cuándo se emplean, cómo se aplican y de los participantes en el proceso de desarrollo.

En ciertas ocasiones el único método será un enfoque paso por paso, comparable con el ciclo de vida de desarrollo de un sistema. En otros casos, el desarrollo de prototipos es el único método que tiene sentido. En otras situaciones se combinan los métodos y, además, los usuarios desarrollan parte de la aplicación, quizá utilizando hojas electrónicas de cálculo y una computadora personal.

El indicador definitivo del éxito de un método de desarrollo en particular es aquel que se refiere a los resultados obtenidos y no a la "precisión" teórica del método.

Ciclo de vida clásico del desarrollo de sistemas

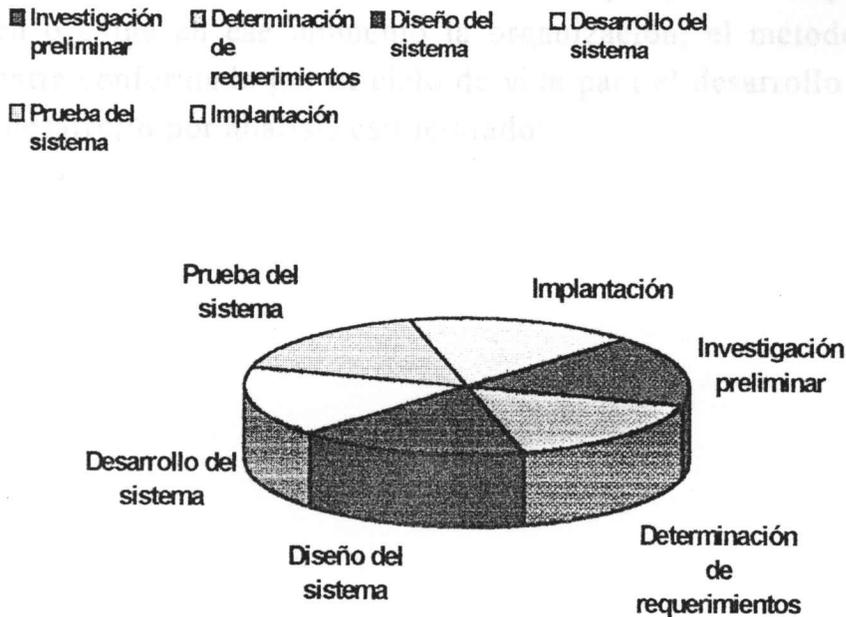
El desarrollo de sistemas, un proceso formado por las etapas de análisis y diseño, comienza cuando la administración o algunos miembros del personal encargado de desarrollar sistemas, detectan que la empresa necesita un sistema o mejoras a un sistema.

En la mayor parte de las situaciones dentro de una empresa todas las actividades están muy relacionadas, en general son inseparables, y quizá sea difícil determinar el orden de los pasos que se siguen para efectuarlas. Las diversas partes del proyecto pueden encontrarse al mismo tiempo en distintas fases de desarrollo; algunos componentes en la fase de análisis mientras que en otros en etapas avanzadas de diseño.

Este método consta de las siguientes actividades:

- I. Investigación preliminar
- II. Determinación de los requerimientos del sistema
- III. Diseño del sistema
- IV. Desarrollo de software
- V. Prueba de los sistemas
- VI. Implantación y evaluación

Como se muestra en la siguiente figura:



Método de desarrollo por análisis estructurado

Muchos especialistas de información reconocen la dificultad de comprender de manera completa sistemas grandes y complejos. El método de desarrollo por análisis estructurado tiene como finalidad superar esta dificultad por medio de: 1) la división del sistema en componentes y 2) la construcción de un modelo del sistema. El método incorpora elementos tanto de análisis como de diseño.

El análisis estructurado se concentra en especificar lo que se requiere que haga el sistema o la aplicación, los elementos esenciales del análisis estructurado son símbolos gráficos, diagramas de flujo y el diccionario centralizado de datos.

Elección del método para el desarrollo del S.E.E.C.

Adquirir un panorama global de todas las actividades y operaciones, distinguir y comprender las relaciones entre las diversas funciones de los diferentes departamentos y considerar, a lo largo del proceso de desarrollo, el impacto que éstas tienen en toda la organización, conduce a crear los sistemas de información más útiles; útiles porque se adaptan a los que existen o exige en ese momento la organización; el método siguiente se encuentra conformado por el ciclo de vida para el desarrollo de sistemas y el de desarrollo por análisis estructurado:

III. Diseño del sistema

1. Requisitos de el desarrollo de sistema
2. Especificación de requisitos
3. Descripción de estructura de datos de flujo de software
4. Diagrama de flujo de datos
5. Diseño de algoritmos
6. Diseño de la interfaz
7. Implementación de algoritmos
8. Diseño de la salida de datos
9. Diseño de la salida de datos
10. Diseño de la salida de datos
11. Diseño de la salida de datos
12. Diseño de la salida de datos
13. Diseño de la salida de datos

IV. Desarrollo de software

V. Pruebas del sistema

Método de desarrollo por análisis estructurado del S.E.E.C.

I) Estudio y Análisis del Sistema

1. *Solicitud del proyecto*
2. *Investigación preliminar*
 - 2.1 *Aclaración de la solicitud*
 - 2.2 *Estudio de factibilidad*
 - 2.2.1 *Factibilidad técnica*
 - 2.2.2 *Factibilidad económica*
 - 2.2.3 *Factibilidad operacional*
 - 2.3 *Aprobación de la solicitud*

II) Determinación de los requerimientos del sistema

1. *Funciones del sistema*
2. *Determinación de los datos de entrada y salida*
3. *Recopilación y análisis de hechos*
4. *Categorías de usuarios*

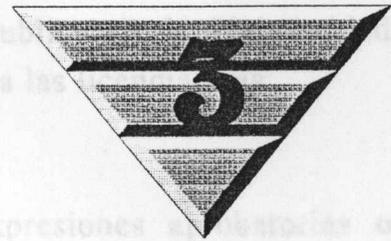
III) Diseño del sistema

1. *Preparando el diseño lógico*
2. *Elementos del diseño*
3. *Iniciando el proceso de diseño físico o del software*
4. *Fundamentos del diseño físico*
5. *Diseño modular*
6. *Diseño de la entrada*
7. *Diseño de la salida*
 - 7.1 *Diseño de las salidas en pantalla*
 - 7.2 *Diseño de las salidas impresas*
8. *Diseño de archivos*
9. *Diseño de interacciones con las bases de datos*
10. *Diseño de Controles*
11. *Diseño de Procedimientos*
12. *Funciones y responsabilidades del personal*

IV) Desarrollo de software

V) Prueba del sistema

ESTUDIO Y ANÁLISIS DEL SISTEMA



El estudio y análisis del sistema se refiere al proceso de examinar el problema con el propósito de utilizar los métodos y procedimientos más adecuados, especificando qué es lo que debe hacer el sistema y la factibilidad de implantación del sistema.

El análisis no está sujeto de ninguna manera a la improvisación, se realizó mediante una labor de investigación minuciosa, que llevó a la investigación misma a formar parte de un proceso más formal, sistemático e intensivo y con la finalidad de llevar acabo el método científico del análisis, o como nos lo presenta Arias Galicia¹ en una de sus definiciones "la investigación puede ser definida como una serie de métodos para resolver problemas cuyas soluciones necesitan ser obtenidas a través de una serie de operaciones lógicas tomando como punto de partida datos objetivos".

IDENTIFICACIÓN DE LAS CARACTERÍSTICAS DEL SISTEMA

Conforme a lo anterior para la recopilación de la información se utilizaron cuestionarios, entrevistas, además de folletos, trípticos y documentos del Centro Nacional de Evaluación para la educación Superior, A.C., obteniendo como resultado lo siguiente :

¹ Arias Galicia, Fernando. *Introducción a la técnica de Investigación en Psicología*. Pág. 28

Solicitud del proyecto

¿Cuál es el problema?

Realizar exámenes a los alumnos del 1° al 9° semestre, de las licenciaturas de Administración de Empresas y Contador Publico de la Universidad Vasco de Quiroga, en las áreas correspondientes a las licenciaturas.

Detalles del problema

La obtención de los resultados no tienen expresiones aprobatorias o reprobatorias; es una prueba válida estadísticamente que evalúa la posición relativa de los alumnos, respecto a los demás sustentantes del examen, así como sus conocimientos y habilidades en relación al perfil general y específico, aprobados por las escuelas de Administración y Contabilidad.

Características de los exámenes

Además de tener en cuenta los resultados que se desean del sistema, otro punto esencial es identificar cuáles son las características de los exámenes. Las cuales se encuentran definitivamente integradas a los resultados que se requieren del sistema, ya que representan los resultados finales; dichas características son:

Objetivo:

Es un instrumento de evaluación que permite de forma efectiva y eficiente, medir el nivel académico de los alumnos.

Aplicable a:

Los alumnos entre séptimo y noveno semestre de las licenciaturas en Administración y Contabilidad.

Contenido:

Debe contener los conocimientos básicos de las áreas esenciales de cada licenciatura.

Áreas de conocimiento:

Las áreas, materias y semestres que integran los exámenes son las siguientes:

Licenciatura en Administración :

Fundamentos de Administración :

Teoría Administrativa	(1°)
Proceso Administrativo	(2°)
Laboratorio de Administración	(3°)

Finanzas :

Matemáticas Financieras I	(4°)
Matemáticas Financieras II	(5°)
Finanzas I	(7°)
Finanzas II	(8°)
Administración de Finanzas	(9°)

Mercadotecnia :

Administración de Ventas	(5°)
Mercadotecnia	(6°)
Investigación de Mercados	(7°)
Publicidad	(8°)
Estrategia de Mercados	(9°)

Producción e Investigación de Operaciones :

Producción I	(7°)
Producción II	(8°)
Producción III	(9°)
Investigación de Operaciones	(9°)

Administración de Personal :

Psicología Industrial	(4°)
Relaciones Humanas	(7°)
Relaciones Industriales I	(8°)
Relaciones Industriales II	(9°)
Laboratorio de Personal	(9°)

Lic. Contabilidad : Contabilidad y Costos:

Contabilidad y Costos:

- Contabilidad I (1°)
- Contabilidad II (2°)
- Contabilidad III (3°)
- Contabilidad IV (4°)
- Costos I (5°)
- Costos I (6°)
- Costos I (8°)

Finanzas :

- Matemáticas Financieras I (4°)
- Matemáticas Financieras II (5°)
- Finanzas I (7°)
- Finanzas II (8°)
- Finanzas III (9°)

Fiscal :

- Legislación Fiscal I (4°)
- Legislación Fiscal II (5°)
- Legislación Fiscal III (6°)
- Seguridad Social (7°)

Auditoría :

- Auditoría I (8°)
- Auditoría II (9°)

Características del cuestionario:

- ♦ Examen cerrado por opción múltiple que provenga del 100% del programa académico (4 opciones de solución por reactivo).
- ♦ Grado de profundidad esencial.
- ♦ Máximo 10 reactivos por programa.
- ♦ En total se aceptan 110 reactivos los cuales se preparan en 2 evaluaciones.
- ♦ Tiene la tendencia a concretar la teoría en aspectos prácticos.
- ♦ Claridad en los planteamientos.
- ♦ El alumno podrá hacer uso de calculadora, formularios básicos y leyes específicas.

Importancia del problema

Permite medir de forma efectiva y eficiente el nivel académico de los alumnos, con la finalidad de poder identificar el grado de conocimientos y, poder tomar las acciones pertinentes para reafirmar el método de enseñanza-aprendizaje.

¿Cuál cree el usuario que es la solución?

Crear un sistema de evaluación computarizado.

¿En qué forma será de ayuda un sistema de información?

1. Apoyará interactiva y formalmente los conceptos esenciales y relativos a las áreas de evaluación.
2. Facilitará a los profesores la creación, rectificación, evaluación y emisión de los exámenes.
3. Evaluará en el tiempo indicado sin tener demoras de resultados.
4. Proporcionará mayor versatilidad en la aplicación de los exámenes.

Investigación preliminar

La solicitud para recibir ayuda de un sistema de información puede originarse por varias razones; sin importar cuáles sean éstas, el proceso se inicia siempre con la petición de una persona - administrador, empleado o especialista en sistemas -..

Cuando se formula la solicitud comienza la primera actividad de sistemas: la *investigación preliminar*. Esta actividad tiene tres partes: aclaración de la solicitud, estudio de la factibilidad y aprobación de la solicitud.

Aclaración de la solicitud

Muchas solicitudes que provienen de empleados y usuarios no están formuladas de manera clara. Por consiguiente, antes de considerar cualquier investigación de sistemas la solicitud de proyecto debe examinarse para determinar con precisión lo que el solicitante desea. Si éste tiene una buena idea de lo que necesita pero no está seguro cómo expresarlo, entonces bastará con afinar los ajustes, ya sea con una llamada telefónica o con el apoyo de cuestionarios. Por otro lado si el solicitante pide ayuda sin saber qué es lo que está solicitando, la aclaración del mismo se vuelve más difícil. En cualquier caso, antes de seguir adelante, la solicitud de proyecto debe estar claramente planteada.

Correspondiente a ésta etapa el problema se tenía claro y preciso, únicamente se tenían algunas incógnitas que debían ser aclaradas para contemplar con exactitud lo que se deseaba del sistema, para esto sirvió de apoyo el siguiente cuestionario con los resultados correspondientes:

1.- ¿Materias o áreas que se evaluarán:?

R = Son 8 áreas, Fundamentos de administración, Finanzas, Mercadotecnia, Producción e Investigación de Operaciones, Administración de personal, Contabilidad y Costos, Fiscal y Auditoría.

2.- *¿Número de opciones de solución para cada pregunta?*

R = 4

3.- *¿Número de reactivos para los exámenes?*

R = Máximo 110 reactivos.

4.- *¿Tipos de reportes?*

R = Reportes de usuarios, de los reactivos y de los exámenes.

5.- *¿Datos para los exámenes?*

R = Nombre del alumno, semestre, licenciatura, fecha de aplicación, maestro que aplica el examen, selección de áreas por guión.

6.- *¿Bases para los exámenes?*

R = Se dieron documentos de la ANFECA y del CENEVAL.

7.- *¿Tamaño máximo de las preguntas?*

R = Relativas al manual del CENEVAL.

8.- *¿Tipo de exámenes?*

R = Impreso

Sugerencia: Impreso e Interactivo, Aceptada

9.- *En caso de examen interactivo, duración de éste?*

R = Total 4 horas, se dividirá en dos sesiones de 2 horas cada una.

10.- *En caso de examen interactivo, ¿se desean resultados inmediatos y directos al alumno?*

R = No.

11.- *¿Características mínimas del hardware donde se implementará el sistema?*

R = Procesador 386 SX, Monitor CGA, Unidad de disco flexible de 3.5 o 5.25 in. de alta densidad, 5 MB de espacio en Disco duro, 1 MB de memoria RAM.

12.- ¿Se desea que el profesor pueda elegir los reactivos para los exámenes o que sea en forma sistemática por el sistema?

R = En forma sistemática.

Con el cuestionario y entrevistas se definió más claramente el problema, el cuál fue planteado por completo anteriormente.

Estudio de factibilidad

Un resultado importante de la investigación preliminar es la determinación de que el sistema solicitado sea factible. Existen tres aspectos relacionados con el estudio de la factibilidad:

a) Factibilidad técnica

El trabajo para el proyecto, ¿puede realizarse con el equipo actual, la tecnología existente de software y el personal disponible?

R = Si se pretende cumplir con el sistema así se logra identificar los

Si se necesita nueva tecnología, ¿cuál es la posibilidad de desarrollarla?

R = En el caso de los exámenes interactivos, se requiere de una red computarizada, se instalará posteriormente.

b) Factibilidad económica

Al crear el sistema, ¿los beneficios que se obtienen serán suficientes para aceptar los costos?

R = No requiere de costos.

c) Factibilidad operacional

Si se desarrolla e implanta, ¿será utilizado el sistema? o ¿habrá resistencia algún cambio por parte de los usuarios?

R = Si, y no habrá resistencia a algún cambio ya que el sistema es de innovación.

Aprobación de la solicitud

La solicitud se aprobó, ya que el sistema es deseable y factible.

Determinación de los requerimientos del sistema

Un requerimiento es una característica que debe incluirse en un nuevo sistema. Esta puede ser la inclusión de determinada forma para capturar o procesar datos, producir información; controlar una actividad de la empresa o brindar soporte a la gerencia. Es así como la determinación de requerimientos vincula el estudio de un sistema con la recopilación de detalles relacionados con él; y para esto, se desglosa en las siguientes puntos:

Funciones del sistema

Las funciones de un sistema deben hacer enfoque y recalcar que es lo que se pretende cumplir con el sistema, así se logra identificar las principales funciones del sistema. Esto se puede lograr con preguntas como:

¿consultas esporádicas?, ¿presentación de datos en diferentes formas?, ¿reportes simples?, ¿Qué desempeñará esencialmente el sistema?, ¿se introducirán datos o se trabajara con bases de datos existentes?, etc.

Para poder identificar las funciones que realiza el sistema de evaluación para exámenes de calidad, me apoye en los siguientes puntos:

1.- ¿Que debe realizar el sistema?

- Permitir la captura, edición o modificación y eliminación de datos.
- Generar reportes de profesores, reactivos y exámenes.
- Debe ser de fácil manejo y accesible.
- Permitir la selección de licenciaturas, semestres y áreas correspondientes al examen.
- Consultar la información de los catálogos.
- Generar exámenes impresos e interactivos.
- Evaluar los exámenes, según ponderaciones que le otorguen los maestros a los reactivos.

2.- *¿Cómo se capturará la información?*

- Con presentación de pantallas, de forma selectiva (popups) y captura directa.

3.- *¿Qué información se capturará?*

- Los reactivos, parámetros de tiempo y ponderación, opciones de solución.
- Licenciaturas, áreas y semestres correspondientes a los reactivos.
- Usuarios del sistema.
- Guión para selección del tipo de examen.
- Datos de los alumnos que se evaluarán.

4.- *¿Procesos por realizar?*

- Almacenamiento de la información capturada en bases de datos específicas
- Mostrar la información en pantalla y en forma de reportes
- Generación de la clave de acceso
- Procesos de evaluación.
- Presentación de catálogos
- Permitir cancelar algún proceso de captura, edición o eliminación
- Consultas generales y específicas
- Procesos para el manejo de errores
- Procesos de seguridad
- Procesos durante la ejecución

La clave de acceso

Para la generación de la clave de acceso, había optado por dejar el proceso al administrador del sistema, que fuera él quien designará la clave a los usuarios; pero existía un inconveniente que debía tomarse en cuenta. ¿Qué tipo de mecanismo utilizaría, una clave de acceso, de identificación o al azar, etc.?, de tal manera que existían probabilidades de asignación de una clave sencilla, por ejemplo la del RFC, y de esta forma identificable por algún otro usuario, lo cual no era deseable. Por lo tanto era más seguro que se creará un proceso para generar dicha clave con un criterio especial de identificación para cada categoría de usuario y de creación.

Determinación de los datos de entrada y salida

La determinación de los datos de entrada y salida es uno de los puntos esenciales en el sistema, ya que los datos de entrada y salida son los que determinan la forma utilizable para su procesamiento. Para esto debe conocerse las características de los medios o documentos, que servirán para obtener los datos de entrada.

Modelos del EGCP-C

El siguiente es un modelo de los reactivos que integran el EGCP-C realizado por el CENEVAL.

Contabilidad

La siguiente es la balanza de comprobación de la inmobiliaria Águila, S.A. al 30 de junio de 1994:

	<u>DEBE</u>	<u>HABER</u>
Sobregiro de efectivo		N\$ 10
Cuentas netas por cobrar	N\$ 35	
Inventarios	N\$ 58	
Pagos anticipados	N\$ 12	
Terrenos (para reventa)	N\$100	
Propiedades, planta y equipo	N\$ 95	
Cuentas y gastos por pagar		N\$ 32
Capital social		N\$ 25
Superávit pagado		N\$150
Utilidades retenidas		N\$ 83
	N\$300	N\$300

Se tienen en existencia cheques girados por N\$ 30, los cuales fueron registrados en la contabilidad el 29 de junio de 1994 y, por falta de fondos, fueron enviados por correo hasta el 9 de julio de 1994.

Los terrenos para reventa fueron vendidos en efectivo el 15 de julio de 1994.

En el balance al 30 de junio de 1994, ¿qué cantidad debe reportar la Inmobiliaria Águila, S.A. como activo circulante ?

- A) N\$ 125
- B) N\$ 195
- C) N\$ 205
- D) N\$ 225

Datos de Entrada:

Maestros :

Nombre Identificación completa del profesor.

Reactivos :

Reactivo Es el texto del reactivo.
Opción a Primera opción.
Opción b Segunda opción.
Opción c Tercera opción.
Opción d Cuarta opción.
Carrera Identifica a que licenciatura pertenece el reactivo.
Área Identifica el área del reactivo.
Semestre Correspondiente al semestre del reactivo.
Tiempo Es una medida para controlar la duración de respuesta de los exámenes.
Peso Sirve para la ponderación de los reactivos.

Exámenes:

Carrera Identifica para que licenciatura se creó el examen.
Semestre A que semestre va dirigido el examen.
Número de copias Sirve para generar el número de exámenes que vayan a aplicarse.
Guión Sirve para la elección del número de preguntas por área.

Interactivo:

Clave del prof. Identifica que profesor que aplico el examen.
Carrera Identifica para que licenciatura se creo el examen.
Semestre A que semestre va dirigido el examen.
Guión Sirve para la elección del número de preguntas por área.

Datos de Salida:

Profesores :

Clave de acceso Clave de acceso al sistema, para los usuarios.

Exámenes:

Nombre	Nombre del alumno que presento el examen.
Carrera	Licenciatura a la que pertenece el alumno.
Semestre	Semestre del alumno.
Calificación	Resultado de su evaluación.

Reportes:

- Nombre de los usuarios.
- Datos de los reactivos.
- Resultados de los exámenes impresos e interactivos.

Recopilación y análisis de hechos

La determinación de requerimientos comprende la recopilación y análisis de hechos; para dicho efecto se aplicó un método estadístico, el cual comprende la identificación, organización, presentación y análisis. Esto proporcionó un marco de referencia para identificar las necesidades del sistema. El método se encuentra apoyado por las siguientes técnicas y métodos:

Entrevistas

Las entrevistas se emplean para reunir información proveniente de personas o de grupos. Por lo común los entrevistados son los usuarios del sistema existente o en potencia del sistema propuesto. En algunos casos, los entrevistados son gerentes o empleados que proporcionan datos para el sistema propuesto o que serán afectados por él.

Es importante recordar que los entrevistados y los analistas conversan durante una entrevista, es decir no se interroga a los primeros. Las entrevistas dan oportunidades para reunir información de las personas que han seleccionado debido a sus conocimientos del sistema que está bajo estudio. A menudo este método es la mejor fuente de información

cualitativa (opiniones, políticas, descripciones subjetivas de actividades y problemas). Este método puede ser de especial utilidad para reunir información de personas que no se comunican por escrito en forma adecuada o que no disponen de tiempo para llenar los cuestionarios. Las entrevistas permiten descubrir áreas mal comprendidas, expectativas poco realistas e incluso indicadores de resistencia hacia el sistema propuesto.

En las entrevistas que se tuvieron con el administrador y con maestros relacionados con el sistema se obtuvieron los resultados siguientes:

- Se trata de un sistema para realizar exámenes de calidad, éstos ya son realizados por el Centro Nacional de Evaluación para la Educación Superior, A.C. (CENEVAL), éste organismo realiza exámenes denominados: Examen General de Calidad Profesional en Contabilidad (EGCP-C). Los exámenes que realizan tienen las siguientes características:

Beneficios que aporta el EGCP-C:

- Que el recién egresado de la facultad cuente con comprobante de su formación académica, expedido por una instancia externa a la institución donde haya realizado sus estudios y que sirva de apoyo a su curriculum vitae.
- Conocer los resultados de las acciones nacionales tendientes a la armonización del nivel académico alcanzado por los egresados de las diferentes y variadas instituciones educativas del país.
- Contar con una modalidad más de titulación, ya vigente en algunas instituciones públicas y particulares.

Características del EGCP-C:

- Es un examen voluntario y esta dirigido a los egresados de las escuelas y facultades de Contabilidad que hayan concluido totalmente sus estudios hasta tres años antes de presentar el examen, estén o no titulados.

- Es una prueba válida estadísticamente que evalúa la posición relativa del egresado respecto a los demás sustentantes del examen, así como el estado que guardan sus conocimientos y habilidades en relación al perfil general y específico aprobados por el Consejo Técnico.
- No condicionan la expedición de título ni de la cédula profesional.
- Está integrado con preguntas de opción múltiple y evalúa resultados de aprendizaje (no se refiere a los elementos ni a los procesos para lograr ese aprendizaje).
- Los resultados son confidenciales y precisan en una constancia, sin expresiones aprobatorias o reprobatorias. En los casos para efecto de titulación, cada centro educativo establece el nivel requerido para ello.
- Puede presentarse tantas veces como se desee, siempre y cuando se cumpla con los requisitos de inscripción que el CENEVAL requiere.

Información obtenida:

- La dirección de las licenciaturas en Administración y Contabilidad desea realizar un examen singular al EGCP-C, pero en forma interna a los alumnos no egresados de dichas licenciaturas.
- Ya que a cada colegio le corresponde vigilar el cumplimiento del programa de educación profesional continua, se supervisará el grado de conocimiento de los alumnos de hasta el 9º semestre.
- La obtención de los resultados servirá para reafirmar conocimientos, métodos de enseñanza-aprendizaje y preparar a los alumnos para que se encuentren dentro de la calidad de profesionistas a nivel nacional e incluso a nivel internacional.

- Los exámenes servirán como estadísticas internas sobre los grados de conocimientos respectivos a las áreas de las licenciaturas.
- Que los alumnos logren alcanzar un grado máximo de calidad y desempeñarse como buenos profesionistas.
- Los reactivos serán realizados por un consejo técnico integrado por profesores de las áreas correspondientes a las licenciaturas.

Cuestionarios

El uso de cuestionarios permite reunir información relacionada con varios aspectos del sistema. El empleo de formatos estandarizados para las preguntas puede proporcionar datos más confiables que otras técnicas; para dicho efecto se tomaron en cuenta las respuestas del cuestionario que sirvió también para la aclaración de la solicitud, obteniéndose los siguientes resultados:

Se evaluarán 8 áreas cada una con sus materias y semestres correspondientes, ya especificadas en las características de los exámenes.

1. Fundamentos de administración
2. Finanzas
3. Mercadotecnia
4. Producción e Investigación de Operaciones
5. Administración de personal
6. Contabilidad y Costos
7. Fiscal
8. Auditoría.

- a) Serán exámenes de opción múltiple con 4 opciones de solución para cada pregunta.
- b) Se aplicarán como máximo 110 reactivos para cada examen.
- c) Duración total del examen 4 horas, se dividirá en dos sesiones de 2 horas cada una.
- d) Los exámenes contarán con los siguientes datos: Nombre del alumno, semestre, licenciatura, fecha de aplicación, maestro que aplica el examen, selección de áreas por guión.
- e) El tamaño máximo de las preguntas serán relativas a las proporcionadas por el CENEVAL.
- f) Los exámenes podrán ser de forma Impresa o Interactiva.
- g) En el caso del examen interactivo, se terminara el examen por decisión del alumno o por cierre automático con aviso de 15 minutos antes del tiempo señalado.
- h) Los resultados en el examen interactivo no serán entregados de forma directa ni inmediata por el sistema.
- i) Los reactivos para los exámenes serán proporcionados por el sistema.
- j) El sistema proporcionará 3 Tipos de reportes: de usuarios, reactivos y exámenes; en varias modalidades.

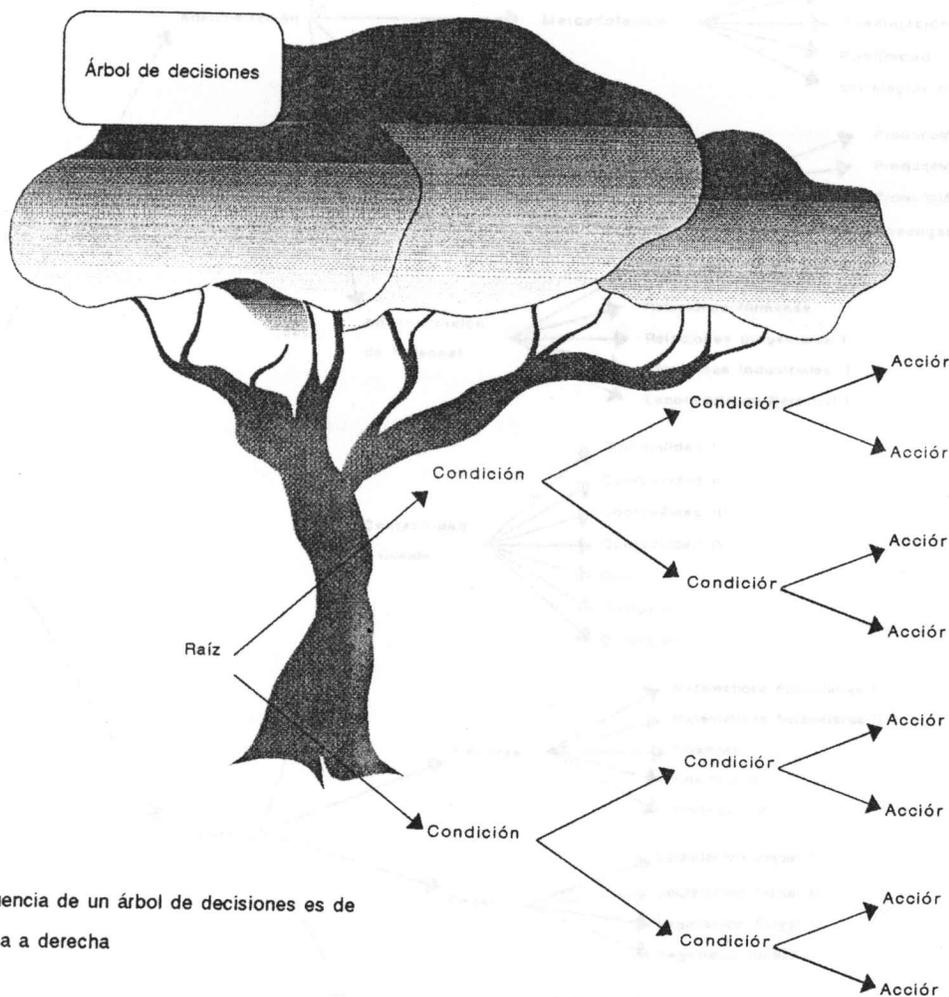
Árboles de decisión

Las personas tienen diferentes formas de ver y decir lo mismo, por ejemplo: "Un vaso medio lleno o un vaso medio vacío". Tener diferentes formas de ver y decir lo mismo puede crear dificultades de comunicación durante los estudios de sistemas (pueden existir malentendidos sobre los comentarios entre el analista y el gerente u olvidar discutir todos los detalles). Por consiguiente se busca evitarlas malas interpretaciones, asimismo, se necesita organizar la información recopilada con respecto a la toma de decisiones.

Los árboles de decisión son uno de los tres métodos que se emplean para describir decisiones y evitar dificultades en la comunicación.

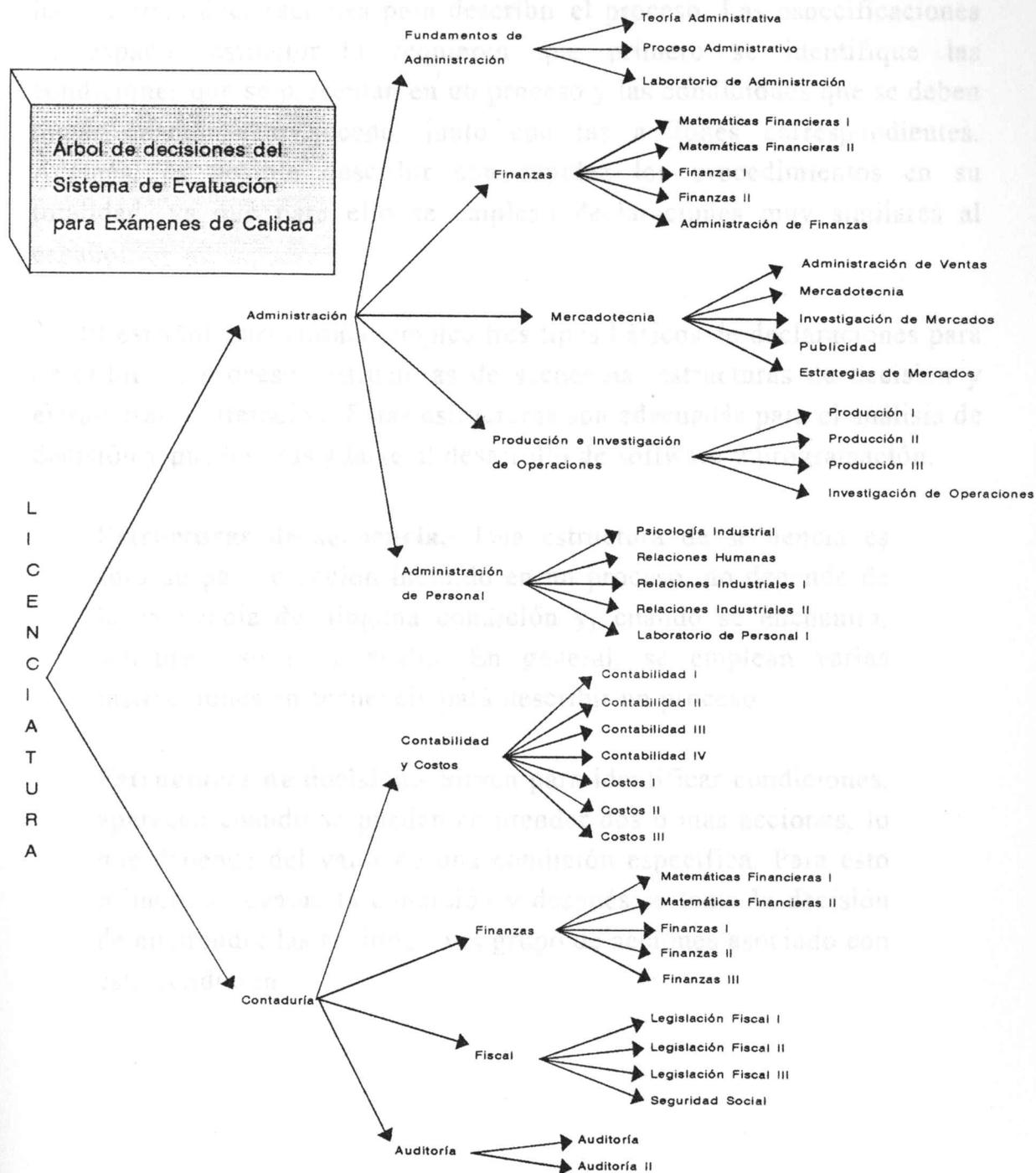
El árbol de decisión es un diagrama que representa en forma secuencial condiciones y acciones; muestra que condiciones se consideran en primer lugar, cuáles en segundo y así sucesivamente. Este método también permite mostrar la relación que existe entre cada condición y el grupo de acciones permisibles asociado con ella; los diagramas de este tipo se parecen a las ramas de un árbol, de aquí su nombre.

La siguiente figura muestra la estructura de un árbol de decisiones.



La secuencia de un árbol de decisiones es de izquierda a derecha

Apoyado en los diagramas de árboles de decisión se identificó de la estructura que tendrán los reactivos y los exámenes del Sistema de Evaluación para Exámenes de Calidad. El siguiente diagrama representa las condiciones en la que se encuentran sujetos los procesos de captura de reactivos, elección de reactivos para los exámenes y captura del guión.



Español estructurado o pseudocódigo

El español estructurado es un método para evitar ambigüedades del lenguaje al establecer condiciones y acciones, tanto en procedimientos como en decisiones. Este método no hace uso de árboles de decisión, en su lugar utiliza declaraciones para describir el proceso. Las especificaciones en español estructurado requieren que primero se identifique las condiciones que se presentan en un proceso y las condiciones que se deben tomar cuando esto sucede, junto con las acciones correspondientes. Además, es posible describir con rapidez los procedimientos en su totalidad, ya que para ello se emplean declaraciones muy similares al español.

El español estructurado emplea tres tipos básicos de declaraciones para describir un proceso: estructuras de secuencia, estructuras de decisión y estructuras de iteración. Estas estructuras son adecuadas para el análisis de decisión y pueden trasladarse al desarrollo de software y programación.

Estructuras de secuencia.- Una estructura de secuencia es solo un paso o acción incluido en un proceso, no depende de la existencia de ninguna condición y, cuando se encuentra, siempre se lleva a cabo. En general, se emplean varias instrucciones en secuencia para describir un proceso.

Estructuras de decisión.- Sirven para identificar condiciones, aparecen cuando se pueden emprender dos o más acciones, lo que depende del valor de una condición específica. Para esto primero se evalúa la condición y después se toma la decisión de emprender las acciones o el grupo de acciones asociado con esta condición.

Estructuras de iteración.- En las actividades rutinarias de operación, es común encontrar que algunas de ellas se repiten mientras existen ciertas condiciones o hasta que éstas se presentan; las estructuras de iteración permiten al analista describir estos casos.

El siguiente ejemplo además de mostrar la estructura del español estructurado, funciona en el desarrollo del sistema de evaluación para exámenes de calidad, adecuando el ciclo de vida para el desarrollo de sistemas y el desarrollo por análisis estructurado.

Si cumple con la investigación preliminar

Ejecuta hasta que los requerimientos hayan sido determinados

Utilizar métodos para encontrar hechos

Ajustar todos los requerimientos

Si cumple con los requerimientos

Avanzar al diseño

Fin de Si

Fin de Ejecuta

Ejecuta hasta que el diseño este concluido

Ajustarse a los estándares del diseño

Utilizar elementos del diseño

Si el diseño se encuentra definido

Implantar diseño

Fin de Si

Fin de Ejecuta

Ejecuta hasta desarrollar el software

Desarrollar el software

Si el desarrollo es correcto

Implantar software

Fin de Si

Fin de Ejecuta

Ejecuta prueba del sistema hasta quedar completamente depurado
Realizar pruebas
Si el sistema esta correcto
Termina sistema
Fin de Si
Fin de Ejecuta
Fin de Si

Debido a las características de este método, se utiliza como apoyo en la etapa de diseño.

Categorías de usuarios

Se emplea el termino usuario para referirse a las personas que no son especialistas en sistemas de información pero que utilizan las computadoras para desempeñar su trabajo. Los usuarios pueden agruparse en cuatro categorías, como lo describe James A. Senn en su libro Análisis y Diseño de Sistema de Información, de la siguiente manera:

Los usuarios primarios.- Son los que interactuan con el sistema. Ellos lo alimentan con los datos (entradas) o reciben salidas, quizá por medio de una terminal.

Los usuarios indirectos.- Son aquellos que se benefician de los resultados o reportes generados por estos sistemas pero que no interactuan de manera directa con el hardware o software.

Los usuarios gerentes.- Existe un tercer tipo de usuarios, éstos son los usuarios gerentes, que tienen responsabilidades administrativas en los sistemas de aplicación. Mientras estas personas no utilicen los sistemas ya se directa o indirectamente, no tendrán la responsabilidad ante la organización de manera efectiva de los sistemas.

Los usuarios directivos.- De particular importancia reviste el hecho de que los usuarios directivos, el cuarto grupo de usuarios, toman cada vez mayor responsabilidad en el desarrollo de sistemas de información. Las organizaciones bien dirigidas consideran el posible impacto y los beneficios de los sistemas de información cuando elaboran su estrategia competitiva. El uso creciente de los sistemas de información, sin embargo, es un arma de dos filos que tiene beneficios y tiene riesgos. Dado que los sistemas de información desarrollados en forma inadecuada pueden entorpecer , e incluso dañar, las actividades de una organización, los directivos deben evaluar de manera constante los riesgos a los que se expone la empresa en caso de falla de los sistemas.

Los cuatro tipos de usuarios son importantes. Cada uno posee información esencial sobre las funciones de la organización y hacia donde se dirige ésta, sin embargo, los analistas son los que proporcionan las ideas (la imaginación) con respecto a las mejores formas para usar eficientemente las computadoras.

Identificación de usuarios del S.E.E.C.

Uno de los puntos que no se deben omitir por ningún motivo, es el puntualizar hacia quien va dirigido el sistema. Ya que un sistema no sólo se diseña con iniciativa propia, sino que también se visualiza hacia quien va dirigido o usuarios de éste, con el fin de no omitir ningún área o función de interés; es por eso que se debe conocer a los usuarios, quienes al final serán los que manejen o controlen el sistema.

Administrador del Sistema.- Dicho usuario se encargará en administrar el sistema, dando acceso a éste por medio de claves, las cuales permitirán entrar e interactuar con él.

Maestros que aplicarán los exámenes.- Los maestros tendrán acceso al sistema por medio de claves asignadas por el administrador. Su función consistirá en aplicar los exámenes impresos e interactivos a los alumnos, así como en entregar los resultados de los exámenes.

Capturistas.- Estos introducen la información de los reactivos y los resultados de los exámenes. Con estos datos el sistema opera internamente obteniendo los resultados necesarios para la obtención de los exámenes y los resultados de las evaluaciones.

Alumnos.- Los alumnos entrarán al sistema al aplicárseles los exámenes interactivos.

Ubicación de categorías de los usuarios del S.E.E.C.

Conforme a las cuatro categorías y a la identificación de los usuarios del S.E.E.C. se ubica a estos últimos de la siguiente forma:

En el sistema de evaluación se ubica en la categoría de usuarios primarios a los capturistas, ya que ellos serán quienes introduzcan toda la información de los reactivos, los datos de entrada.

En el tipo de examen interactivo son los alumnos quienes también toman el papel de usuarios primarios ya que ellos introducen información de los exámenes.

Dentro de la categoría de usuarios indirectos entran los alumnos, ya que ellos no interactúan con el sistema, pero son los que se están evaluando de una manera oportuna y eficaz (en el proceso de exámenes impresos).

En la categoría de usuarios gerentes se encuentran los maestros, ya que de ellos tienen ciertas funciones administrativas del sistema, al ser ellos quienes preparen y apliquen los exámenes utilizando el sistema, además de obtener los resultados y administrar las aplicaciones que el sistema les ofrece.

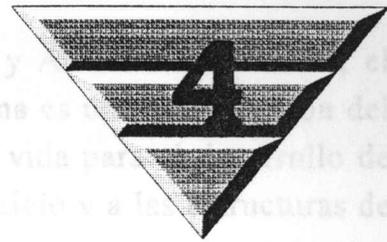
Se estableció la ubicación del administrador del sistema en la categoría de usuario directivo, ya que es él quien dirige todos los aspectos y estrategias del sistema.

Un esfuerzo conjunto

Una de las características de los sistemas de información con mayor éxito (éxito en términos de beneficio para la empresa) se originan con los usuarios. Una razón para ello es que las solicitudes de estos sistemas se originan de una necesidad de la organización que los usuarios perciben: por ejemplo, la necesidad de resolver un problema en particular, de manejar funciones rutinarias o de monitorear la información para evitar ciertos problemas.

El hecho de que en estas empresas los usuarios contribuyan con ideas que conduzcan hacia sistemas con éxito, tal como debe ser, demuestra el propósito fundamental de un sistema de información y el más importante, es mejorar la organización y no el de probar el valor de una tecnología sofisticada. El desarrollo de sistemas con éxito sin embargo, es un esfuerzo conjunto. Las contribuciones de los usuarios son importantes y los analistas tienen un papel esencial: extraer las mejores ideas de los usuarios para su análisis y discusión.

DISEÑO DEL SISTEMA



El diseño de un sistema de información produce los detalles que establecen la forma en que el sistema cumplirá con los requerimientos identificados durante la fase del análisis (diseño lógico) y la fase que entra con el desarrollo del software (diseño físico). Para entrar a la etapa del diseño físico se concluyó con la etapa del diseño lógico, ya que esta sirvió de transición entre el estudio y análisis del sistema, los requerimientos y la especificación de los elementos lógicos; como se observará posteriormente en cada uno de los puntos que lo integran .

No hay una idea única por donde debe iniciarse un diseño, por lo que el diseño lo adecua el analista según su criterio; no obstante la gran mayoría inicia con el proceso de diseño de identificación de reportes y demás salidas que debe producir el sistema. Además de especificar con precisión los datos para cada reporte y salida. El diseño indica también los datos de entrada, aquellos que serán calculados y los que deben ser almacenados. Así mismo, se escriben con todo detalle los procedimientos de cálculo y los datos individuales. Los diseñadores seleccionan las estructuras de archivo y los dispositivos de almacenamiento, tales como discos y cintas magnéticas o incluso archivos en papel. Los procedimientos indican cómo procesar los datos y producir las salidas. Los documentos que contienen las especificaciones de diseño representan a éste de muchas maneras (diagramas, tablas y símbolos especiales). La información detallada del diseño se proporciona para comenzar la fase de desarrollo de software.

CARACTERÍSTICAS DEL DISEÑO

Como se comentó en el capítulo de Estudio y Análisis del sistema, el método que se eligió para el desarrollo del sistema es una combinación del método estructurado y el método por el ciclo de vida para el desarrollo de sistemas, por lo cual se recurre a cada etapa del ciclo y a las estructuras de desarrollo. Ya entrada la etapa de diseño se tomaron ciertos elementos esenciales como a continuación se muestran:

Preparando el diseño lógico

Para preparar el diseño se deben de tomar ciertos aspectos y estrategias acordes al criterio y la creatividad del analista, estos criterios surgen de diferentes enfoques, los cuáles podemos identificar más fácilmente si nos cuestionamos sobre la calidad que debemos darle al diseño.

- ¿Es posible que se efectúen pasos adicionales o no autorizados?
- ¿Pueden presentarse actividades duplicadas?
- ¿Cuál es la forma apropiada de los procesos?
- ¿De que forma se obtendrá mayor funcionalidad del diseño del sistema?
- ¿Está disponible la información necesaria para llevar a cabo una tarea o alcanzar cierto objetivo?
- ¿Los detalles se pueden recuperar y presentar en forma adecuada cuando son necesarios?

Conforme a lo anterior se tomaron los siguientes aspectos:

Accesibilidad de la información

Cuando es difícil el acceso a la información, se debe considerar si hay un problema de información (¿Existe la información correcta y disponible?) o es sólo cuestión de procedimientos (¿que procedimientos

deben seguirse para tener acceso y procesar la información?). Se debe tener en cuenta que, aunque exista la información necesaria, las personas tienden a confiar más en su memoria o corazonadas porque el procedimiento de recuperación es demasiado laborioso. El método de recuperación puede ser difícil o bien la información se presenta en forma equivocada.

Entre las estrategias más representativas para evitar los problemas anteriores, se encuentran las siguientes:

o Predeterminar información

Cierta información puede ser recuperada y utilizada por un sistema automatizado sin la necesidad de la intervención del usuario. En otros casos, el desarrollo de opciones preestablecidas, ejecutadas a menos que se presenten consideraciones, eliminará un requerimiento de información.

Opciones preestablecidas del S.E.E.C.

- Para los procesos de agregar y editar reactivos, se diseñaron los datos de las licenciaturas y áreas correspondientes. Cuando se capture la carrera únicamente aparecerán la información que se requiera agregar al reactivo, éstas se encuentran de la siguiente manera:

Administración
Contabilidad
Admón. y
Conta.

Para las áreas se utiliza el mismo criterio que para la carrera; en este se selecciona una de las ocho áreas que integran a las licenciaturas:

Admón. de Personal
Auditoria
Contabilidad y Costos
Finanzas
Fiscal
Fundamentos de Admón.
Mercadotecnia
Producción e I. de O.

- Otro caso en el que se realizan opciones preestablecidas es en la generación de exámenes, en el cuál se presentan las siguientes opciones:

Carrera

Administración
Contabilidad

Semestre

Séptimo
Octavo
Noveno

- Además en el proceso de agregar usuarios, cuenta con opciones para la elección del tipo de categoría del usuario:

Administrador
Maestro
Capturista

o Facilitar el acceso a la información y a los procesos

Los sistemas se pueden diseñar para que respondan a mandatos y dependan de opciones preestablecidas que se ajusten a las situaciones que se presentan con mayor frecuencia.

Dentro del sistema de evaluación se apoyan las situaciones que se presentan con mayor frecuencia utilizando mandatos que en forma directa, esto es sin la necesidad de seguir un procedimiento, realizan el proceso para el que fueron creadas.

Mandatos del S.E.E.C.

Ayuda con:	SHIFT+F1
Maestros con:	ALT+M
Reactivos con :	ALT+R
Exámenes con:	ALT+E
Reportes con :	ALT+T
Salir con :	CTRL+X
Agregar con :	ALT+F5
Editar con:	ALT+F6
Eliminar con:	ALT+F7
Terminar proceso con:	CTRL+X
Generar y Aplicar con:	SHIFT+F5
Calificar y Revisar con	SHIFT+F6

o Eliminar la necesidad de procesamiento

Los detalles utilizados con mayor frecuencia pueden almacenarse en una forma que no requiera procesamiento o manejo. En este caso mejoran tanto la disponibilidad como la accesibilidad.

En éste caso se trata de procesos que se pueden efectuar con frecuencia, lo cual podemos identificar de inmediato como es el **proceso de ayuda**, se almacena o carga en memoria y se encuentra disponible cuando sea necesario.

Complejidad

La elegancia de la sencillez es un atributo de los sistemas de información mejor diseñados. Requerir que una persona participe en tareas interrelacionadas, es algo que puede crear una complejidad innecesaria. Los procedimientos que contienen un número muy grande de tareas, pasos o actividades, a menudo dan como resultado un rendimiento inaceptable o incompleto. Para reducir la complejidad se deben considerar tres estrategias:

- ***Simplificación***

La simplificación puede obtenerse al eliminar los pasos que no son necesarios o el registro de información que no se utiliza. El uso de estándares preestablecidos simplificará los procesos. Los casos que no se ajustan a los establecido por los estándares pueden manejarse con excepciones. Proporcionar estándares más explícitos o lineamientos también pueden ayudar a simplificar procesos que de otra manera serían complejos.

- ***División***

Dividir un proceso complejo en tareas separadas disminuye con frecuencia la complejidad. (El mismo principio se emplea en el análisis de flujo de datos, dividir los sistemas de información en sus componentes.) Al dividir un proceso, se debe considerar qué tareas tiene que ser hechas por las personas y cuáles son más adecuadas por una computadora.

- **Cambios en la secuencia**

Cambiar el orden en el que ocurre el proceso puede disminuir la complejidad aparente. De esta manera es posible poner a disponibilidad información esencial y las respuestas a las preguntas difíciles se vuelven inmediatamente obvias.

UN SISTEMA FÁCIL DE UTILIZAR

Uno de los objetivos principales del diseño, consiste en facilitar las actividades del usuario u operador para manejar el sistema; en relación a éste aspecto, el diseño:

- **Incorpora características que hacen al sistema fácil de comprender y utilizar**

Las opciones de menús, opciones preestablecidas y la captura de información guían paso a paso al usuario y le indican las acciones que debe realizar según su elección.

- **No desalienta a los usuarios por errores cometidos o la falta de cuidado por parte de ellos**

Por ejemplo si el usuario al generar un examen comete un error en la selección del guión, el sistema le permite regresar a modificar la selección; o en el caso de cometer errores en la captura de los datos, se presenta una opción de editar, la cuál permite modificar elecciones preestablecidas e incluso información de captura, como es el texto del reactivo, opciones, etc.

- **Evita fallas o procedimientos inapropiados que generan perjuicios o complicaciones para los usuarios**

Para evitar fallas o procedimientos inapropiados, el sistema se encuentra diseñado con el apoyo de controles que permiten verificar y validar la información que se maneja, así cómo el acceso al sistema y a las bases de datos, entre otros.

- **Tiene suficiente flexibilidad para adaptarse a las necesidades de cada usuario**

La flexibilidad que ofrece el sistema permite al usuario seleccionar la información o los procedimientos por medio de un conjunto de teclas, con el uso del ratón o con las flechas del teclado.

- **Permite la creciente familiaridad del usuario con el sistema**

La información se encuentra ubicada secuencialmente, lo que permite al usuario incrementar su familiaridad en cada ocasión que utilice el sistema; además de que ofrece una consistencia en cada uno de sus procedimientos.

- **Funcionamiento ergonómico para la comodidad del usuario**

Como son los colores para la presentación de la información, la ubicación de teclas de mandato, flexibilidad o acceso por diferentes medios (conjunto de teclas, uso del ratón, por menús, etc.), los métodos de interacción con el sistema y opciones para confirmar o cancelar los procedimientos.

Proporcionar especificaciones detalladas para el desarrollo de software

Al igual que con las características de un sistema de información, el software también debe ser diseñado con cuidado. El diseño de sistemas incluye la formulación de especificaciones de software. Estas especificaciones establecen las funciones de entrada, salida y procesamiento así como los algoritmos necesarios para efectuarlas. El software junto con las rutinas, se enfocan sobre lo que cada función realiza; asimismo, se especifican los procedimientos para llevar a cabo dichas funciones. La selección de lenguajes de programación, paquetes de software y utilerías se efectúa durante el proceso de diseño lógico y las recomendaciones se incluyen como parte de las especificaciones del software.

Especificaciones para el desarrollo de software del S.E.E.C.

Se eligió como lenguaje de programación el xBase en su versión FoxPro ver. 2.6 para MS-DOS, por su aplicabilidad al manejo de las bases de datos y ofrecer herramientas para la creación de ventanas que facilitan la elaboración del sistema.

Los paquetes de apoyo seleccionados fueron: Harvard Graphics ver. 3.0 para MS-DOS, Microsoft Word ver. 2.0/6.0 para Windows, en la creación de gráficas y edición de texto, así como para diagramas de flujo de datos, de control de procesos, etc. Y como utilerías se ocupó: QPro ver. 2.0/3.0 para Windows, para la comprobación de los cálculos (promedios).

Estándares de diseño

Los objetivos del diseño de sistemas son muy amplios y afectan aspectos tanto de la aplicación como de la organización en la que será utilizado el sistema. Es por eso que también deben mantenerse estándares para el desarrollo; las especificaciones de diseño se establecen dentro del marco fijado por los estándares.

Las áreas incluidas en el diseño de los estándares del S.E.E.C. son:

Estándares para datos.-

Los lineamientos para asignar nombre a los datos y especificar su longitud y tipo se tomaron conforme al control de la información que se maneja en las bases de datos. Los campos de los nombres similares a los datos, el tipo de acuerdo al manejo, por ejemplo: carreras, áreas y semestres son representativos y se utilizan como preestablecidas las cuales

son de tipo numérico al igual que con las que se hacen operaciones de calculo, de tipo carácter las que llevan información que debe ser legible para el usuario y la longitud conforme a los formatos establecidas por el consejo técnico ya comentados con anterioridad; por último los campos memo son aquellos en los que se necesita editar datos.

Ejemplo de los estándares para los datos:

<i>Dato</i>	<i>Campo</i>	<i>Tipo</i>	<i>No.</i>	<i>Dec.</i>
Carreras	Carreras	Numérico	1	0
Áreas	Áreas	Numérico	1	0
Semestres	Semestres	Numérico	1	0
Nombre del reactivo	Nombre	Carácter	20	
Tiempo del reactivo	Tiempo	Numérico	2	0
Peso del reactivo	Peso	Numérico	2	0
Opción No. 1	Op1	Carácter	70	
Opción No. 2	Op2	Carácter	70	
Opción No. 3	Op3	Carácter	70	
Opción No. 4	Op4	Carácter	70	
Texto del reactivo	Texto	Memo	10	

Estándares de codificación.-

Las abreviaturas y designaciones formales que describen las actividades y entidades dentro de la organización se diseñaron conforme a la categoría que tienen los usuarios y los tipos de movimientos que realizan.

Dichas designaciones son las siguientes:

Administrador del Sistema.- Dicho usuario se encargará en administrar el sistema, dando acceso a los usuarios que bajo su criterio designe la categoría como administrador, maestro o capturista.

Maestros.- Los maestros tendrán acceso al sistema por medio de claves asignadas por el administrador. Su función consistirá en aplicar los exámenes impresos e interactivos a los alumnos, así como en los movimientos para entregar los resultados de los exámenes.

Capturistas.- Estos introducen la información de los reactivos y los resultados de los exámenes. Con estos datos el sistema opera internamente obteniendo los resultados necesarios para la obtención de los exámenes y los resultados de las evaluaciones.

Alumnos.- Los alumnos entrarán al sistema al aplicárseles los exámenes interactivos y sólo lo utilizarán para presentar dichos exámenes.

Estándares estructurales.-

Los lineamientos sobre cómo se estructuró el software y el sistema se encuentran bajo las siguientes políticas para dividir el software en módulos, para la codificación estructurada y la relación existente entre los componentes del sistema.

Una computadora no tiene capacidad para solucionar problemas más que cuando se le proporcionan los pasos sucesivos que realizará, éstos indican las instrucciones a ejecutar por la máquina y que constituyen el algoritmo.

Los problemas se pueden resolver más eficazmente con la computadora cuando se rompen o descomponen en subproblemas que sean más fáciles de solucionar que el original. La descomposición del problema original en subproblemas más simples y a continuación éstos en otros más simples que puedan ser implementados para su solución en la computadora, se le denomina diseño descendente o diseño

modular. Las ventajas más importantes del diseño modular son las siguientes:

- * **El diseño se comprende más fácilmente al dividirse en partes más simples denominadas módulos.**
- * **El control de las acciones de los módulos es más fácil y eficiente.**
- * **La comprobación del problema se verifica más fácilmente.**
- * **Se obtiene una mayor oportunidad de solución.**

Estándares de documentación.-

Las descripciones de las características del diseño de sistemas, de la relación entre componentes y de las características de operación se encuentran apoyadas por diagramas funcionales, español estructurado, diagramas Warnier, diagramas de flujo de datos, de control de procesos, árboles de decisión, gráficas, etc. Esto ayudó a una descripción correcta del sistema y a mantener los estándares de documentación.

Elementos del diseño

¿Con que características se deben diseñar?. Las especificaciones de diseño describen las características del sistema, los componentes o elementos y la forma en que éstos aparecerán a los usuarios. Los componentes ya descritos durante el análisis de requerimientos, necesitan de ciertos elementos de apoyo para el proceso de transición, estos son los siguientes:

Flujo de datos

El flujo de datos se muestra mejor de forma gráfica; las herramientas utilizadas al seguir esta estrategia muestran todas las características esenciales del sistema y la forma en que se ajustan entre sí. Puede ser difícil de comprender en su

totalidad un proceso si se emplea para ello sólo una descripción verbal; es por eso que los diagramas facilitan el entendimiento. El siguiente diagrama muestra el flujo de datos del S.E.E.C.

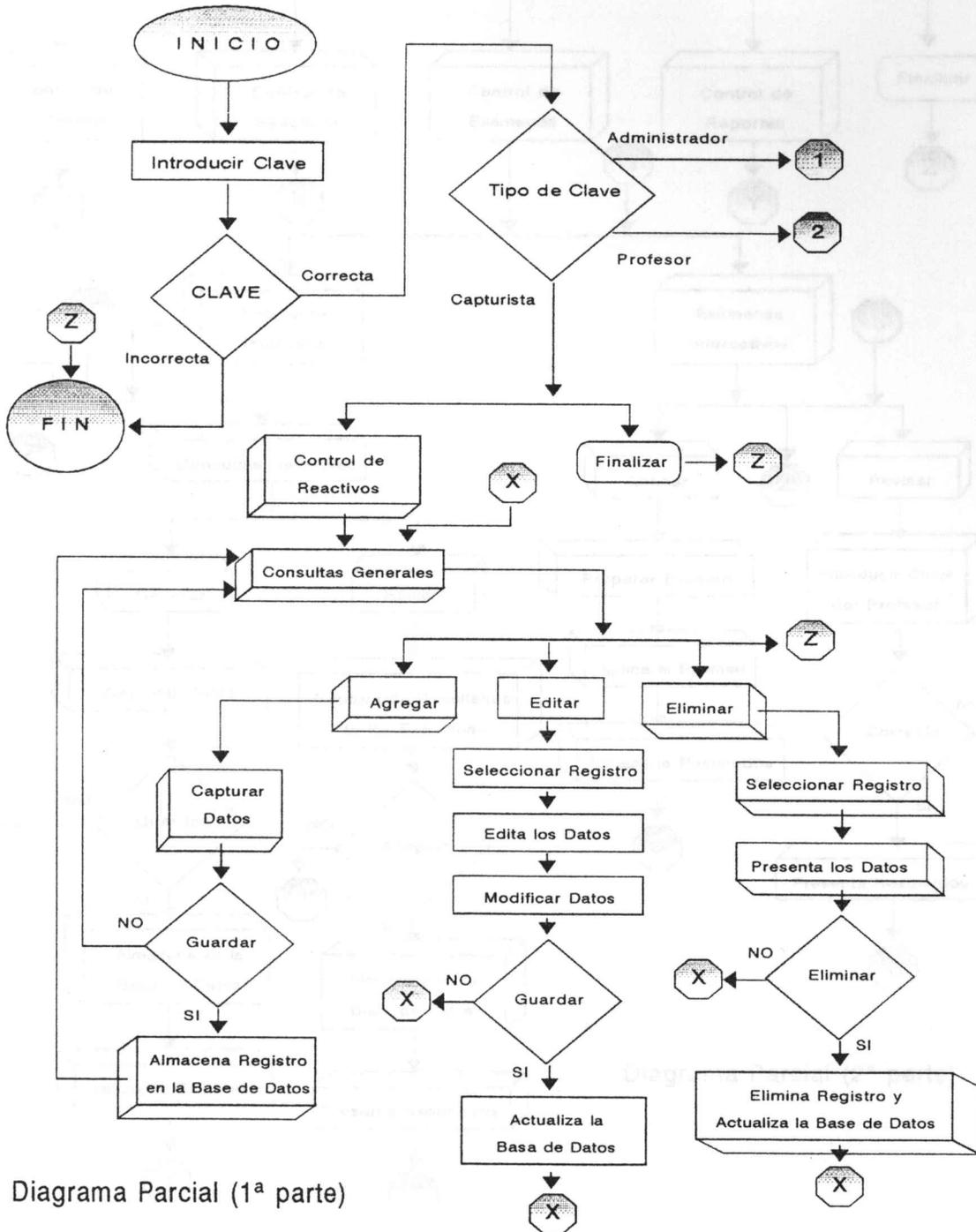


Diagrama Parcial (1ª parte)

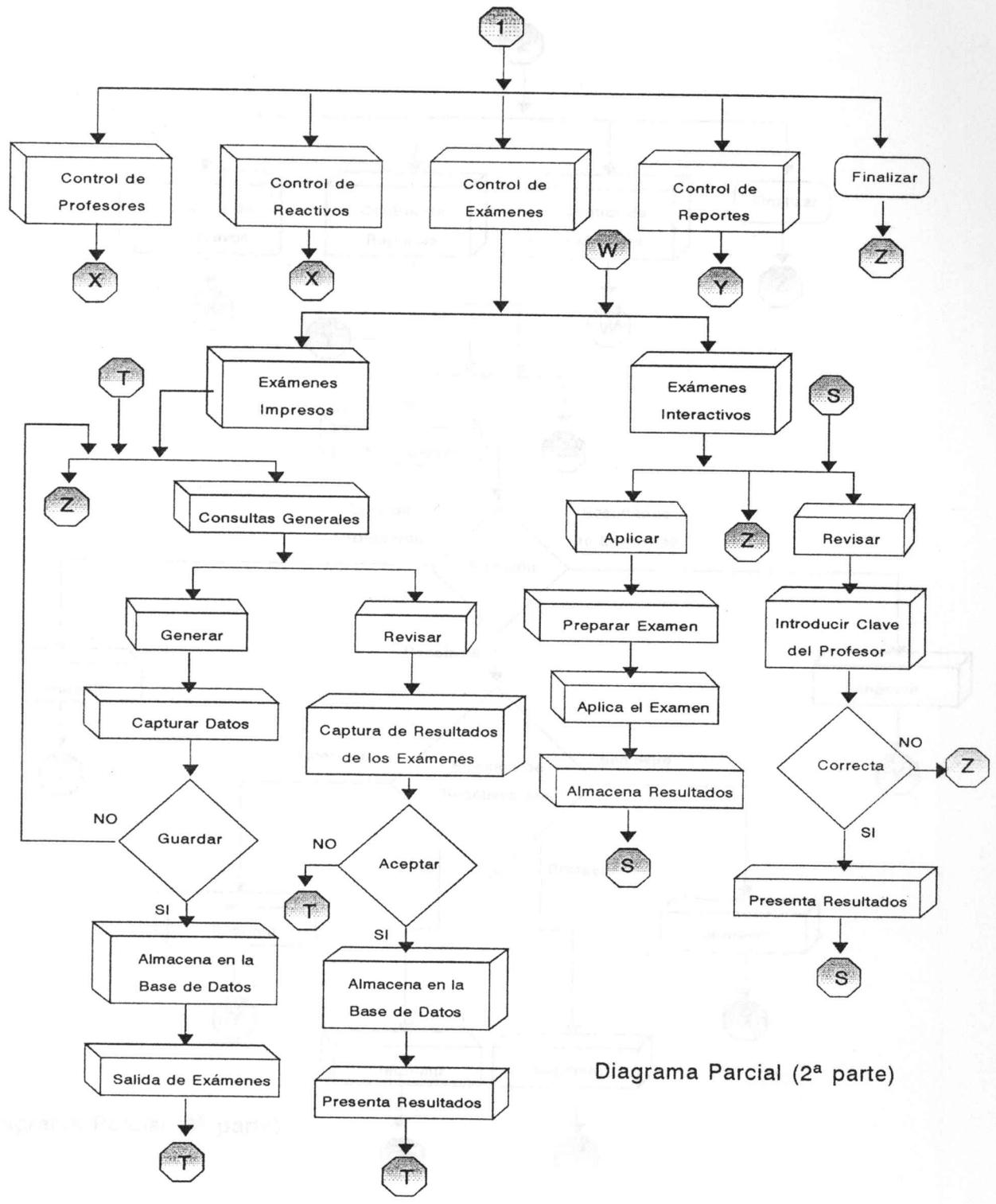


Diagrama Parcial (2ª parte)

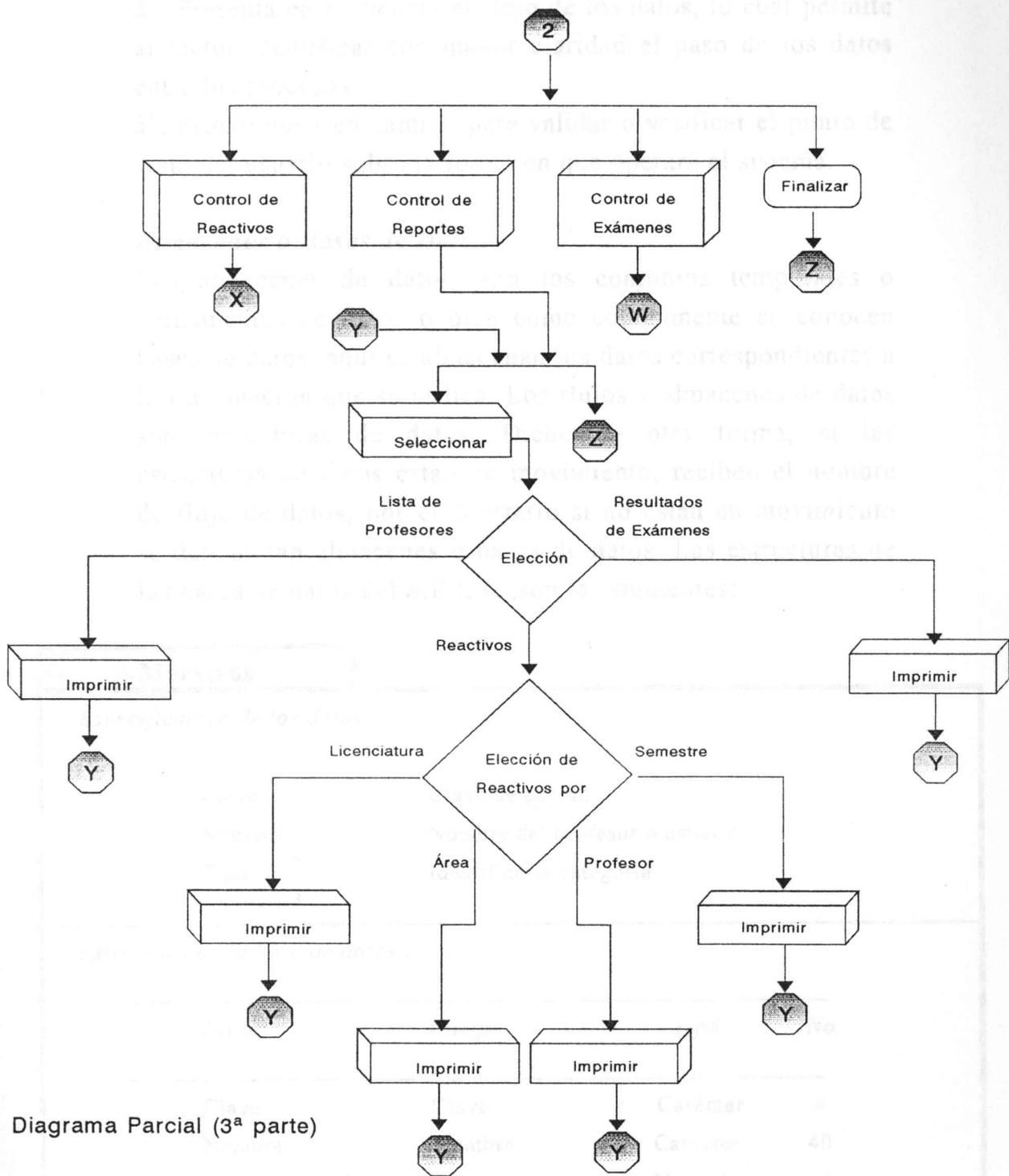


Diagrama Parcial (3ª parte)

Se empleo este diagrama por las siguientes razones.

- 1°. La interacción es más fácil de comprender que las políticas empleadas para administrar la aplicación.
- 2°. Presenta en secuencia el flujo de los datos, lo cuál permite al lector identificar con mayor claridad el paso de los datos entre los procesos.
- 3°. Proporciona un camino para validar o verificar el punto de vista del usuario sobre la forma en que operará el sistema.

Almacenes o Bases de Datos

Los almacenes de datos, son los conjuntos temporales o permanentes de datos o bien como comúnmente se conocen bases de datos, aquí se almacenan los datos correspondientes a la información que se utiliza. Los flujos y almacenes de datos son estructuras de datos. Dicho de otra forma, si las estructuras de datos están en movimiento, reciben el nombre de flujo de datos; por el contrario si no están en movimiento se denominan almacenes o bases de datos. Las estructuras de las bases de datos del S.E.E.C. son las siguientes:

Maestros			
<i>Especificación de los datos:</i>			
Clave	Clave de acceso.		
Nombre	Nombre del profesor o usuario.		
Tipo	Identifica la categoría.		
<i>Estructura de la base de datos :</i>			
<i>Dato</i>	<i>Campo</i>	<i>Tipo</i>	<i>No.</i>
Clave	Clave	Carácter	4
Nombre	Nombre	Carácter	40
Tipo de usuario	Tipo	Numérico	1

Reactivos

Especificación de los datos:

Texto	Es el texto del reactivo.
Opción a	Primera opción.
Opción b	Segunda opción.
Opción c	Tercera opción.
Opción d	Cuarta opción.
Nombre	Es un nombre corto que identifica al reactivo.
Carreras	Identifica a que licenciatura pertenece el reactivo.
Áreas	Identifica el área del reactivo.
Semestres	Correspondiente al semestre del reactivo.
Tiempo	Es una medida para controlar la duración de respuesta de los exámenes.
Peso	Sirve para la ponderación de los reactivos.

Estructura de la base de datos :

<i>Dato</i>	<i>Campo</i>	<i>Tipo</i>	<i>No.</i>	<i>Dec.</i>
Texto	Texto	Memo	10	
Opción a	Op1	Carácter	70	
Opción b	Op2	Carácter	70	
Opción c	Op3	Carácter	70	
Opción d	Op4	Carácter	70	
Nombre	Nombre	Carácter	20	
Carreras	Carreras	Numérico	1	0
Áreas	Áreas	Numérico	1	0
Semestres	Semestres	Numérico	1	0
Tiempo	Tiempo	Numérico	2	0
Peso	Peso	Numérico	2	0

Exa0001

Especificación de los datos generados:

Clave	Clave del registro con respecto a la base de datos.
Exanum	Número del examen.
Solución	Ubicación de las 4 opciones.
Tiempo	Tiempo del reactivo
Peso	Peso del reactivo

Estructura de la base de datos :

<i>Dato</i>	<i>Campo</i>	<i>Tipo</i>	<i>No.</i>
Clave	Clave	Carácter	5
Número del examen	Exanum	Numérico	4
Solución del examen	Solución	Carácter	5
Tiempo	Tiempo	Numérico	2
Peso	Peso	Numérico	2

Exámenes

Especificación de los datos:

Carrera	Identifica para que licenciatura se creó el examen.
Semestre	A que semestre va dirigido el examen.
Clave	Clave del examen.
Número de exámenes	Sirve para generar el número de exámenes que vayan a aplicarse.
Pregunta 1	Representa el número de preguntas del área de Fundamentos de Administración.
Pregunta 2	Representa el número de preguntas del área de Finanzas.
Pregunta 3	Representa el número de preguntas del área de Administración de Personal.
Pregunta 4	Representa el número de preguntas del área de Mercadotecnia.
Pregunta 5	Representa el número de preguntas del área de Producción e Investigación de Operaciones.
Pregunta 6	Representa el número de preguntas del área de Contabilidad y Costos.
Pregunta 7	Representa el número de preguntas del área Fiscal.
Pregunta 8	Representa el número de preguntas del área de Auditoría.
Clave del Profesor	Relaciona al profesor que generó el examen.
Fecha	Fecha de creación del examen.
Maestro	Nombre del maestro que diseñó el examen.

Estructura de la base de datos :

<i>Dato</i>	<i>Campo</i>	<i>Tipo</i>	<i>No.</i>	<i>Dec.</i>
Carrera	Carrera	Numérico	1	0
Semestre	Semestre	Numérico	1	0
Clave	Clave	Carácter	14	
Número de exámenes	Numexam	Numérico	2	0
Pregunta 1	Pregunta1	Numérico	2	0
Pregunta 2	Pregunta2	Numérico	2	0
Pregunta 3	Pregunta3	Numérico	2	0
Pregunta 4	Pregunta4	Numérico	2	0
Pregunta 5	Pregunta5	Numérico	2	0
Pregunta 6	Pregunta6	Numérico	2	0
Pregunta 7	Pregunta7	Numérico	2	0
Pregunta 8	Pregunta8	Numérico	2	0
Clave del profesor	Claveprof	Carácter	4	
Fecha	Fecha	Date	8	
Maestro	Maestro	Carácter	40	

Ayuda

Especificación de los datos:

Referencia Sirve de apoyo referencial con el dato que solicita la ayuda.

Texto Es la información de ayuda sobre al dato.

Estructura de la base de datos :

<i>Dato</i>	<i>Campo</i>	<i>Tipo</i>	<i>No.</i>
Referencia	Referencia	Carácter	30
Texto	Texto	Memo	10

SEEC

Especificación de los datos:

Texto

Es la información sobre datos del S.E.E.C.

Estructura de la base de datos :

<i>Dato</i>	<i>Campo</i>	<i>Tipo</i>	<i>No.</i>
Texto	Texto	Memo	10

Procesos o Interacciones

Los procesos que se utilizaron en las actividades para aceptar, manejar y suministrar datos e información dentro del sistema, fueron los siguientes:

Básicos:

Agregar
 Editar
 Eliminar
 Consultar

Específicos

Generar
 Calificar
 Aplicar
 Revisar

Procedimientos

Los métodos y rutinas para utilizar el sistema de información y lograr con ello los resultados específicos se diseñaron para la facilitar el uso del sistema de una manera simple y efectiva bajo los puntos que se estudian en el *Diseño de Procedimientos*.

- Entrada al sistema
- Selección de opciones
- Captura de datos
- Generación de exámenes
- Generación de reportes
- Salida del sistema

Controles

Los controles son estándares y procedimientos para determinar si las actividades están ocurriendo en la forma anticipada o aceptada, es decir si se encuentran bajo control. Asimismo especifican las acciones que tienen que emprenderse cuando ocurren problemas o se presentan circunstancias inesperadas.

Funciones del personal

Las responsabilidades de todas las personas que tienen que ver con el nuevo sistema, incluyendo a los usuarios, operadores de computadora y personal de apoyo. Abarca todo el espectro de componentes del sistema, incluso desde la entrada de datos hasta la distribución de salidas o resultados.

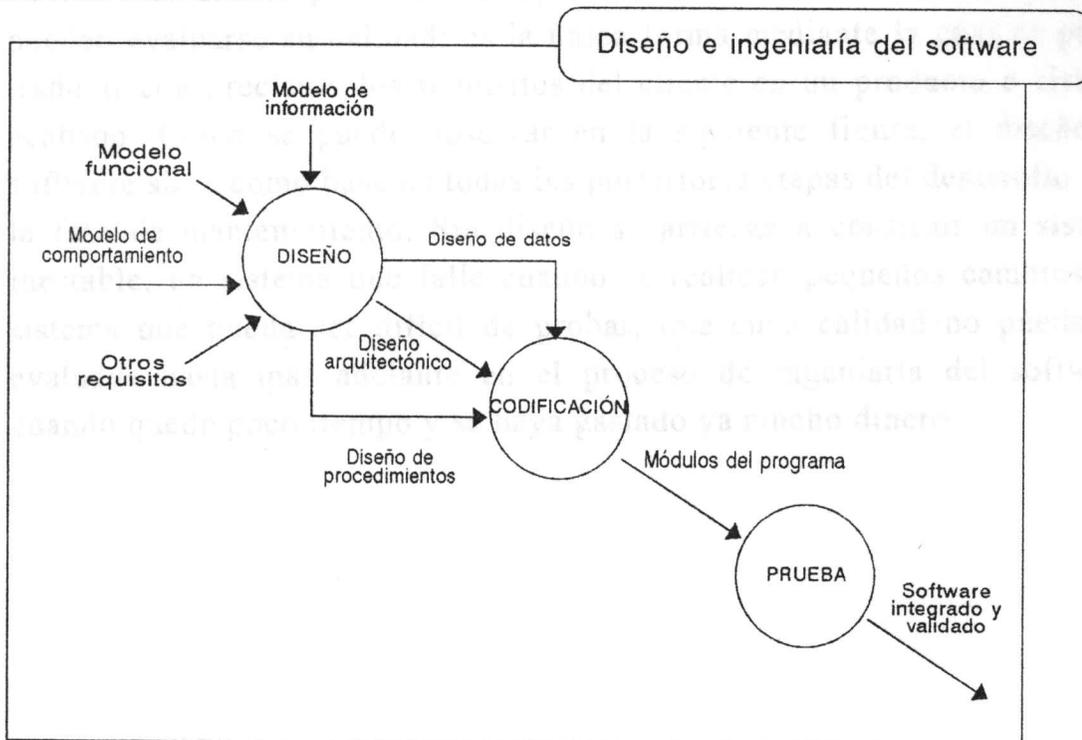
Posteriormente estos elementos se describirán totalmente al irse integrando en cada una de las etapas del diseño.

Iniciando el proceso del diseño físico o software

El diseño es el primer paso de la fase de desarrollo de cualquier producto o sistema de ingeniería. Taylor, E.S., en su libro *Un reporte en Diseño de Ingeniería* define el diseño como: "... el proceso de aplicar distintas técnicas y principios con el propósito de definir un dispositivo, proceso o sistema con los suficientes detalles como para permitir su realización física". El diseño de software se asienta en el núcleo técnico del proceso de ingeniería del software. Una vez que se han establecido los requisitos del software, el diseño del software es la primera de tres actividades técnicas *diseño, codificación y prueba*.

Cada actividad transforma la información de forma que finalmente se obtiene un software para computadora válido.

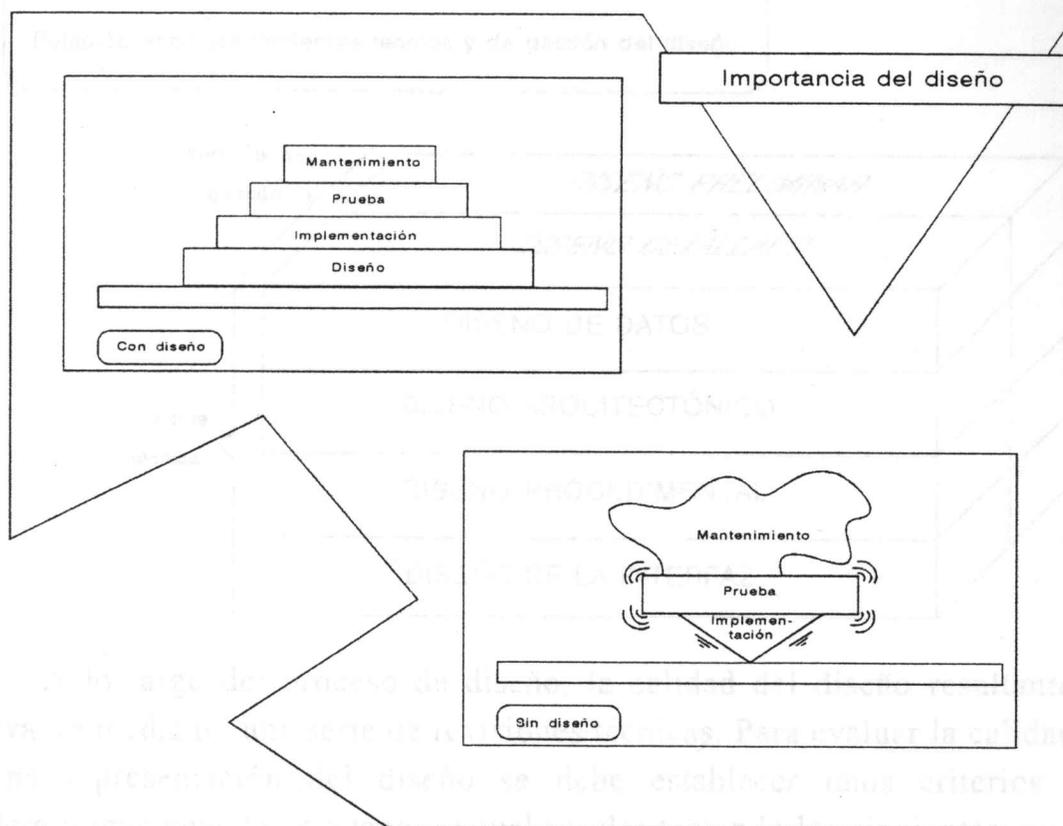
En la siguiente figura se muestra el flujo de la información durante la fase de desarrollo.



Los requisitos del programa, establecidos mediante el estudio y el análisis del sistema alimentan el paso del diseño. Para esto se integra en tres partes: diseño de datos, diseño arquitectónico y diseño procedimental.

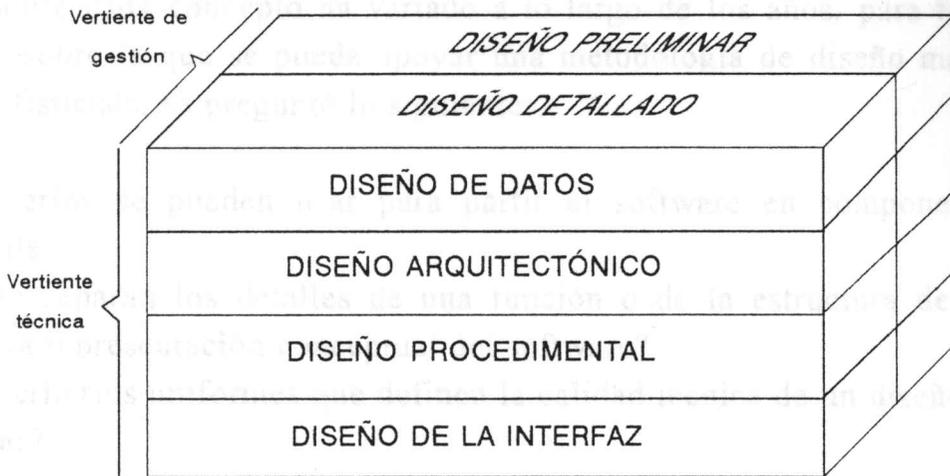
- El diseño de datos transforma el modelo del campo o datos creados en el estudio y análisis, en las estructuras de datos que se van a requerir para implementar el software.
- El diseño arquitectónico define las relaciones entre los principales elementos estructurales del programa.
- El diseño de procedimientos transforma los elementos estructurales en una descripción procedimental del software. Se genera el código fuente y para integrar y validar el software, se llevan a cabo las pruebas.

La importancia del diseño físico se puede sentar con una única palabra *calidad*. El diseño produce las representaciones del software de las que pueden evaluarse su calidad; es la única forma mediante la cual se puede traducir con precisión los requisitos del cliente en un producto o sistema acabado. Como se puede observar en la siguiente figura, el diseño de software sirve como base de todas las posteriores etapas del desarrollo y de la fase de mantenimiento. Sin diseño se arriesga a construir un sistema inestable, un sistema que falle cuando se realicen pequeños cambios; un sistema que pueda ser difícil de probar, que cuya calidad no pueda ser evaluada hasta más adelante en el proceso de ingeniería del software, cuando quede poco tiempo y se haya gastado ya mucho dinero.



El diseño del software es un proceso mediante el que se traducen los requisitos en una representación del software. Desde el punto de vista de la gestión del proyecto, el diseño del software se realiza en dos pasos. El *diseño preliminar* se centra en la transformación de los requisitos en los datos y la arquitectura del software. El *diseño detallado* se ocupa del refinamiento de la representación arquitectónica que lleva a una estructura de datos detallada y a las representaciones algorítmicas del software. Dentro del contexto de los diseños preliminar y detallado, se llevan a cabo varias actividades de diseño diferentes. Además del diseño de datos, del diseño arquitectónico y del diseño procedimental, muchas aplicaciones modernas requieren una actividad distinta de *diseño de la interfaz*. El diseño de la interfaz establece la disposición y los mecanismos para la interacción hombre - máquina. Las relaciones entre las vertientes técnica y de gestión se muestran en la siguiente figura.

Relación entre las vertientes técnica y de gestión del diseño



A lo largo del proceso de diseño, la calidad del diseño resultante se evaluó mediante una serie de revisiones técnicas. Para evaluar la calidad de una representación del diseño se debe establecer unos criterios que determinan cuándo es bueno, lo cual resultó teniendo los siguientes:

- Un diseño debe exhibir una organización jerárquica que haga un uso inteligente del control de los componentes del software.
- Un diseño debe ser modular; esto es, el software debe estar dividido de forma lógica en elementos que realicen funciones y subfunciones específicas.
- Debe contener representaciones distintas y separadas de los datos y de los procedimientos.
- Debe llevar a módulos que exhiban características funcionales independientes.
- Debe llevar a interfaces que reduzcan la complejidad de las conexiones entre los módulos y el entorno exterior.
- Un diseño debe obtenerse mediante un método que sea reproducible y que esté conducido por la información obtenida durante el análisis y requerimientos.

Fundamentos del diseño físico

En las últimas tres décadas se han establecido un conjunto de conceptos fundamentales para el diseño del software. Aunque el grado de interés sobre cada concepto ha variado a lo largo de los años, para tener una base sobre la que se pueda apoyar una metodología de diseño más o menos sofisticada, se preguntó lo siguiente:

¿Qué criterios se pueden usar para partir el software en componentes individuales?

¿Cómo se separan los detalles de una función o de la estructura de los datos de la representación conceptual del software?

¿Existen criterios uniformes que definen la calidad técnica de un diseño de programas?

M.A. Jackson en su libro Principales Diseños de Programación, dice: "El comienzo de la sabiduría de un programador de computadoras está en reconocer la diferencia entre obtener un programa que funcione y obtener uno que funcione correctamente".

Los conceptos fundamentales del diseño de software proporcionan la base necesaria para que funcione correctamente, teniendo entre ellos a los siguientes:

Abstracción

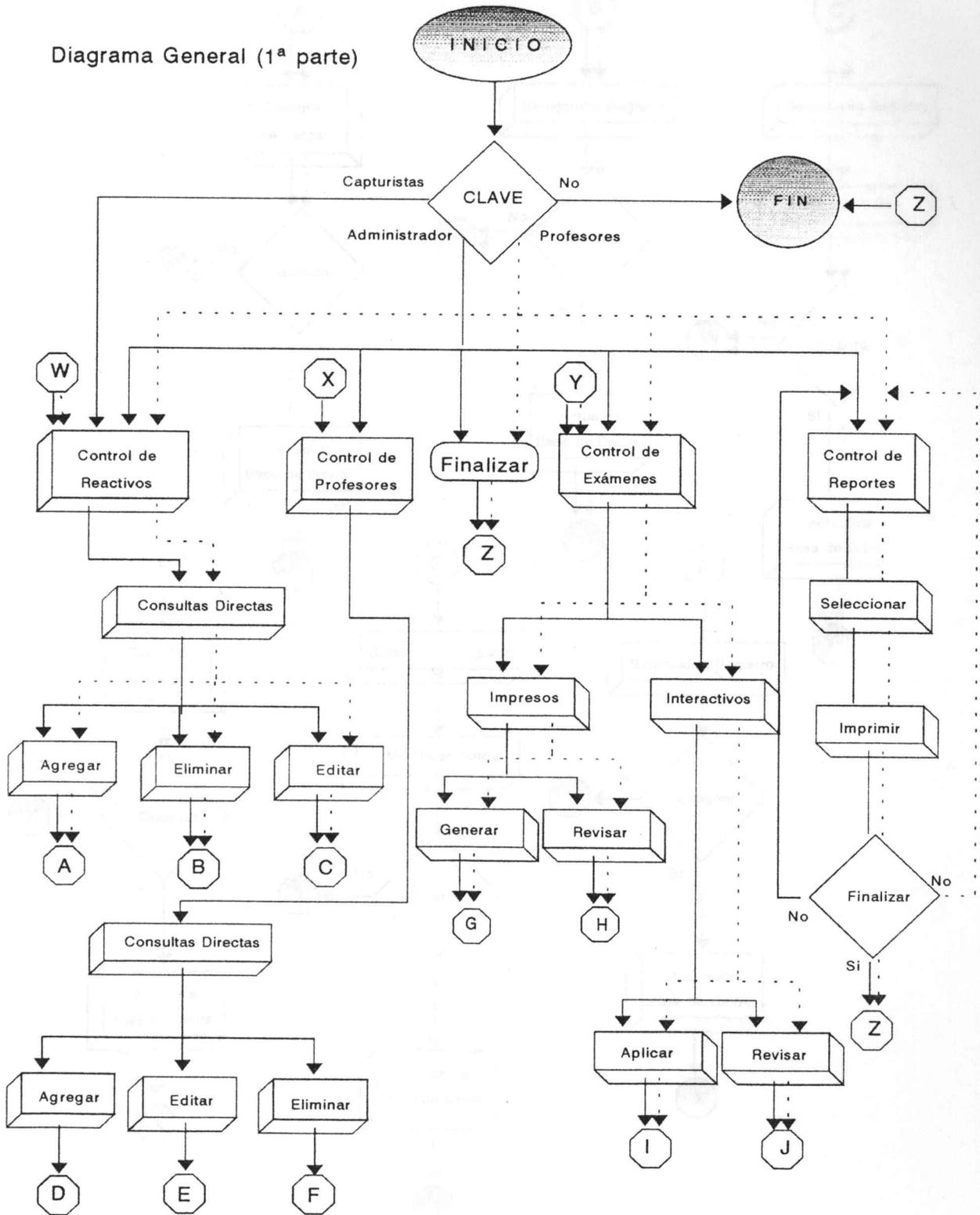
Cuando se considera una solución modular para cualquier problema, pueden formularse diferentes formas y niveles de abstracción, la abstracción permite concentrarse en un problema al mismo nivel de generalización, independientemente de los detalles irrelevantes de bajo nivel; el uso de la abstracción también permite trabajar con conceptos y términos que son familiares al entorno del problema, sin tener que transformarlos a una estructura.

En la fase de abstracción se utilizó como herramienta auxiliar de trabajo un diagrama de flujo de datos general del sistema, el cual sirvió para poder ir orientando y diseñado los procesos que serían incluidos en el sistema; dicho diagrama permitió tener una visión de los procesos que debían contemplarse en el sistema.

A continuación se muestra el diagrama de flujo de datos general del S.E.E.C., el cual sirvió como una forma de abstracción.



Diagrama General (1ª parte)



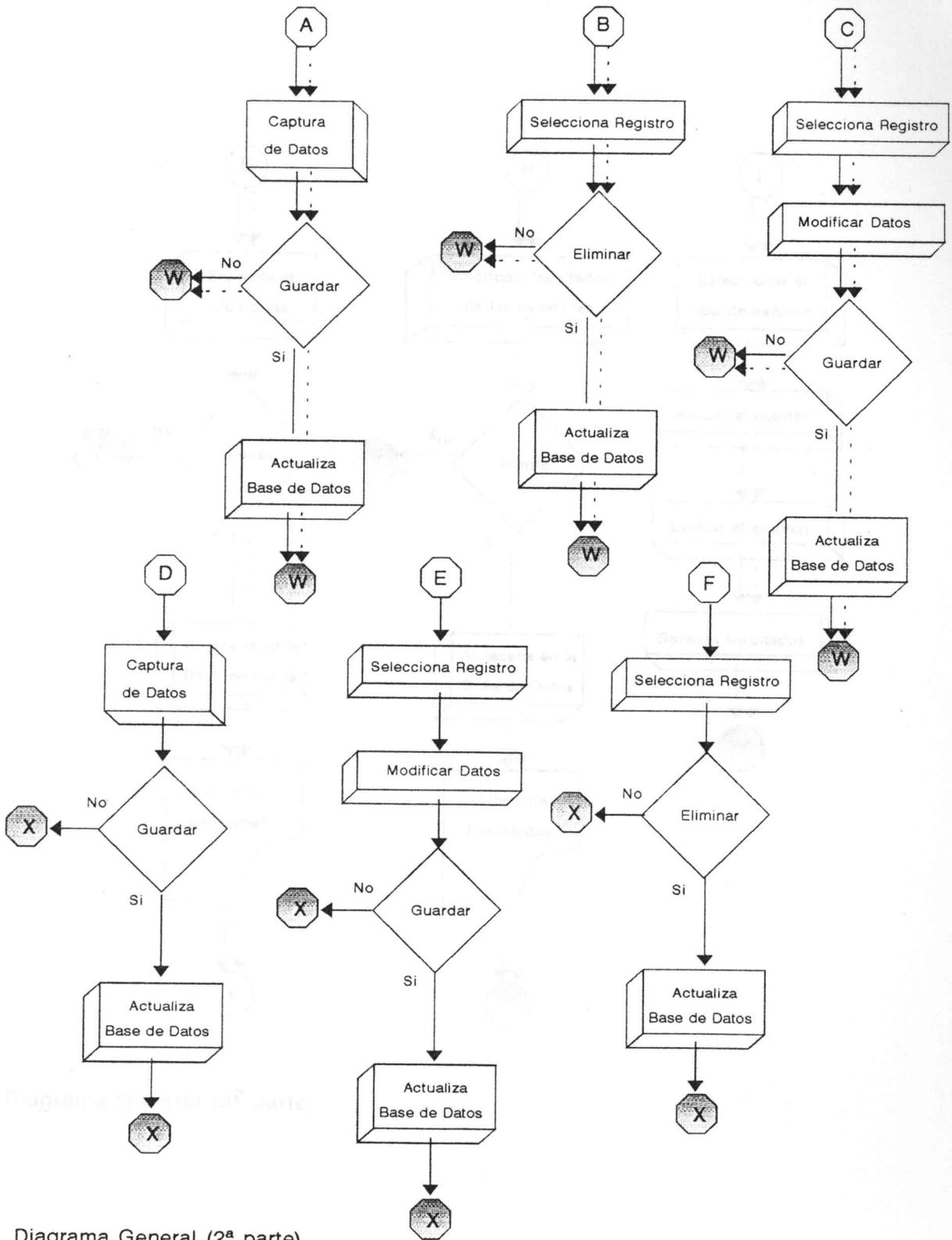


Diagrama General (2ª parte)

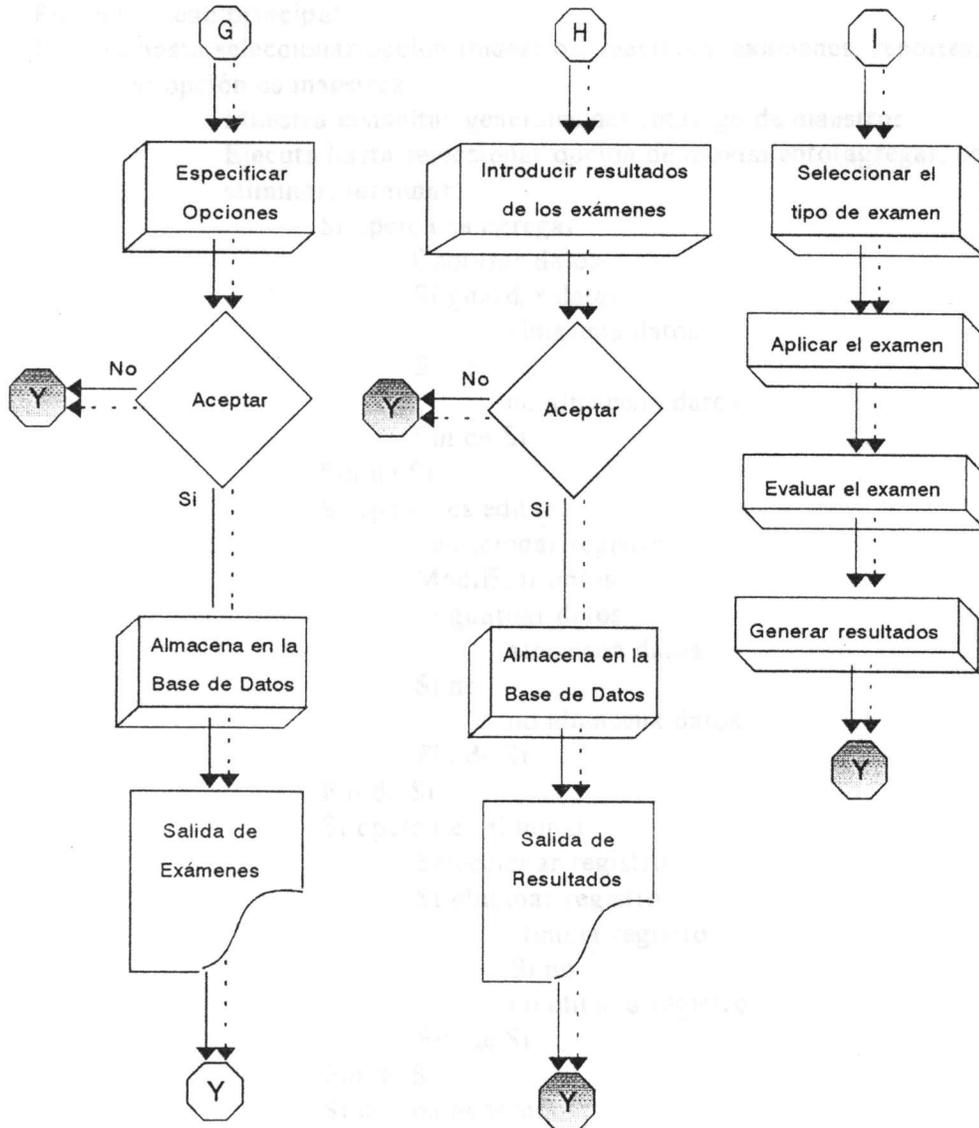


Diagrama General (3ª parte)

DISEÑO DEL SISTEMA

Además de tener una visión gráfica se obtuvo información por medio del español estructurado, que en esta etapa fungió como otro medio de abstracción.

Introducir clave de acceso

Si clave de acceso correcta

Presenta menú principal

Ejecuta hasta seleccionar opción (maestros, reactivos, exámenes, reportes, salir)

Si opción es maestros

Muestra consultas generales del catálogo de maestros

Ejecuta hasta seleccionar opción de movimiento (agregar, editar eliminar, terminar)

Si opción es agregar

Capturar datos

Si guardar datos

almacena datos

Si no

no almacena datos

Fin de Si

Fin de Si

Si opción es editar

Seleccionar registro

Modificar datos

Si guardar datos

almacena datos

Si no

no almacena datos

Fin de Si

Fin de Si

Si opción es eliminar

Seleccionar registro

Si eliminar registro

elimina registro

Si no

no elimina registro

Fin de Si

Fin de Si

Si opción es terminar

termina opción de movimiento maestros

Si no

Continúa opción de movimiento maestros

Fin de Si

Fin de Ejecuta

Fin de Si

Si opción es reactivos

Muestra consultas generales del catálogo de reactivos

Ejecuta hasta seleccionar opción de movimiento (agregar, editar, eliminar, terminar)

Si opción es agregar

Capturar datos

Si guardar datos

almacena datos

Si no

no almacena datos

Fin de Si

Fin de Si

Si opción es editar

Seleccionar registro

Modificar datos

Si guardar datos

almacena datos

Si no

no almacena datos

Fin de Si

Fin de Si

Si opción es eliminar

Seleccionar registro

Si eliminar registro

elimina registro

Si no

no elimina registro

Fin de Si

Fin de Si

Si opción es terminar

termina opción de movimiento reactivos

Si no

Continúa opción de movimiento reactivos

Fin de Si

Fin de Ejecuta

Fin de Si

Si opción es exámenes

Muestra consultas generales del catálogo de exámenes

Ejecuta hasta seleccionar opción de movimiento (generar, calificar, terminar)

Si opción es generar

Capturar datos

Si guardar datos

almacena datos

genera examen

Si no

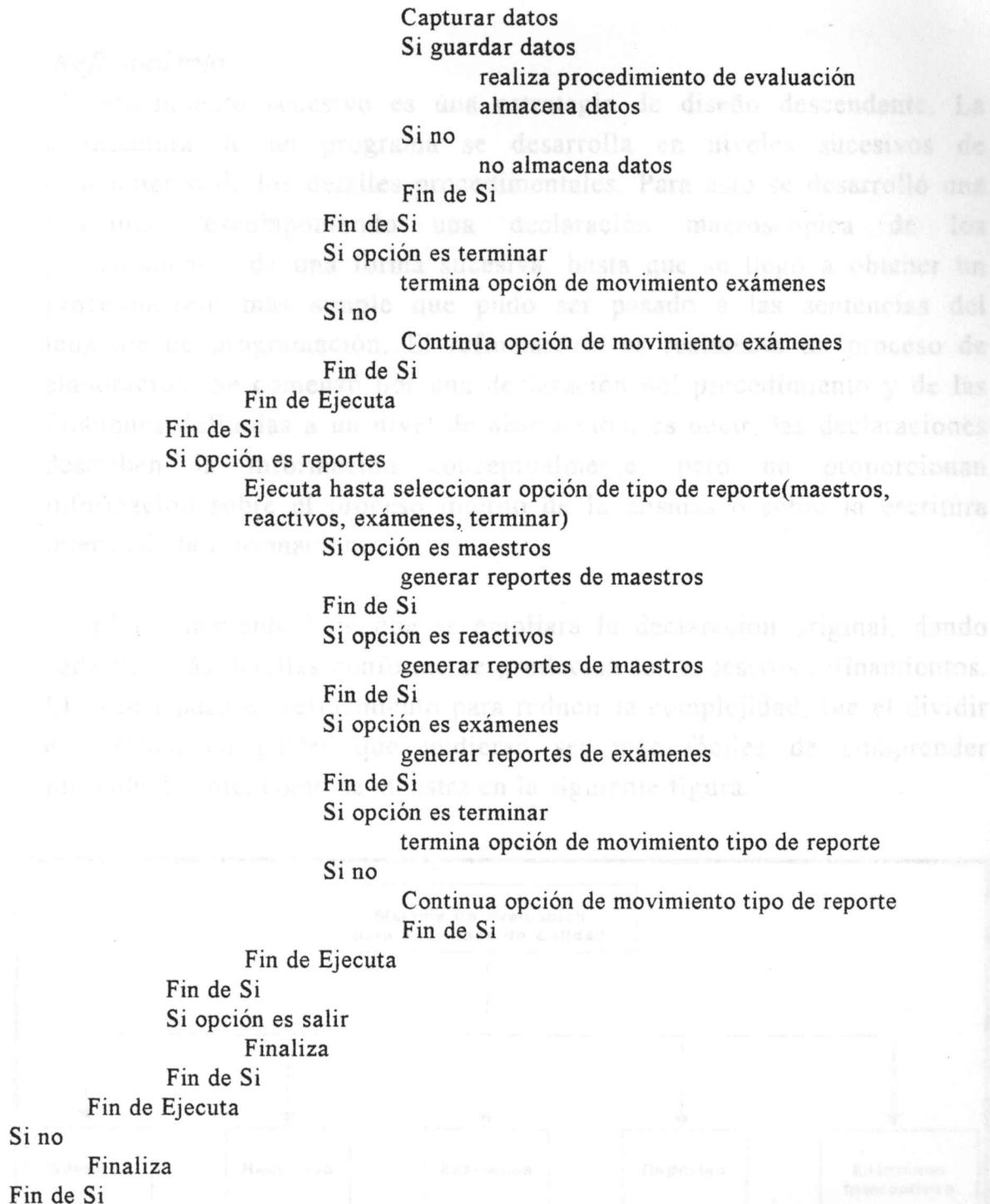
no genera examen

Fin de Si

Fin de Si

Si opción es calificar

Seleccionar registro



Lo anterior sirvió para tener una visión general de los procedimientos que debían incluirse en el sistema; aunque por el cierto grado de complejidad que se presentó, la solución del problema se descompuso en otras partes. Como ya se mencionó anteriormente, para obtener una solución menos compleja, más eficiente y en forma que pudiera ser revisada por partes, se utilizó la siguiente estrategia.

Refinamiento

El refinamiento sucesivo es una estrategia de diseño descendente. La arquitectura de un programa se desarrolla en niveles sucesivos de refinamiento de los detalles procedimentales. Para esto se desarrolló una jerarquía descomponiendo una declaración macroscópica de los procedimientos de una forma sucesiva, hasta que se llegó a obtener un procedimiento mas simple que pudo ser pasado a las sentencias del lenguaje de programación. El refinamiento es realmente un proceso de elaboración. Se comenzó por una declaración del procedimiento y de las funciones definidas a un nivel de abstracción, es decir, las declaraciones describen la información conceptualmente, pero no proporcionan información sobre el proceso interno de la mismas o sobre la escritura interna de la información.

El refinamiento hizo que se ampliará la declaración original, dando cada vez más detalles conforme se producían los sucesivos refinamientos. El primer paso de refinamiento para reducir la complejidad, fue el dividir el sistema en partes que pudieran ser más fáciles de comprender individualmente, como se muestra en la siguiente figura.

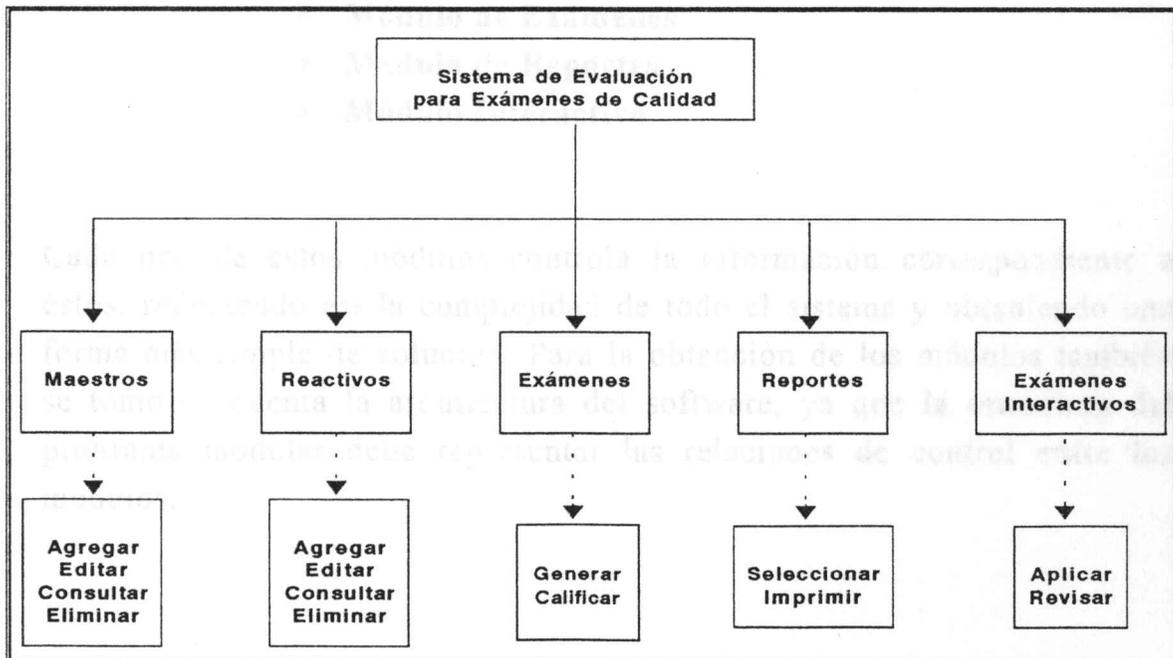


Fig. a.6 Diseño estructurado descendente.

Al reducir la complejidad dividiendo la solución en varias partes más simples entramos en lo que es modularidad.

Modularidad

La arquitectura implica modularidad; esto es el software se divide en componentes con nombres y ubicaciones determinados, que se denominan *módulos* y que se integran para satisfacer los requisitos de solución del problema. La modularidad es el atributo individual del software que permite a un programa ser intelectualmente manejable. El software monolítico (es decir, un programa compuesto de un solo módulo) no puede ser fácilmente abarcado por un lector (como se mostró anteriormente). El número de caminos de control, la expansión de las referencias, el número de variables y la complejidad global podrían hacer imposible su correcta comprensión.

Los módulos del S.E.E.C. acorde al funcionamiento y a la información que estos controlarían se diseñaron de la siguiente forma:

- **Módulo Principal**
- **Módulo de Maestros**
- **Módulo de Reactivos**
- **Módulo de Exámenes**
- **Módulo de Reportes**
- **Módulo Interactivo**

Cada uno de estos módulos controla la información correspondiente a éstos, reduciendo así la complejidad de todo el sistema y obteniendo una forma más simple de solución. Para la obtención de los módulos también se tomo en cuenta la arquitectura del software, ya que la estructura del programa modular debe representar las relaciones de control entre los módulos.

Arquitectura del software

La arquitectura del software se refiere a dos características importantes del software de computadora:

- 1) La arquitectura jerárquica de los componentes procedimentales (módulos) y
- 2) La estructura de los datos

La arquitectura del software se obtiene mediante un proceso de partición, que relaciona los elementos de una solución de software con partes del problema real definido implícitamente durante el análisis de los requerimientos. La siguiente gráfica muestra el diseño de la arquitectura; integrando los módulos con las estructuras de los datos.

Diagrama de Módulos

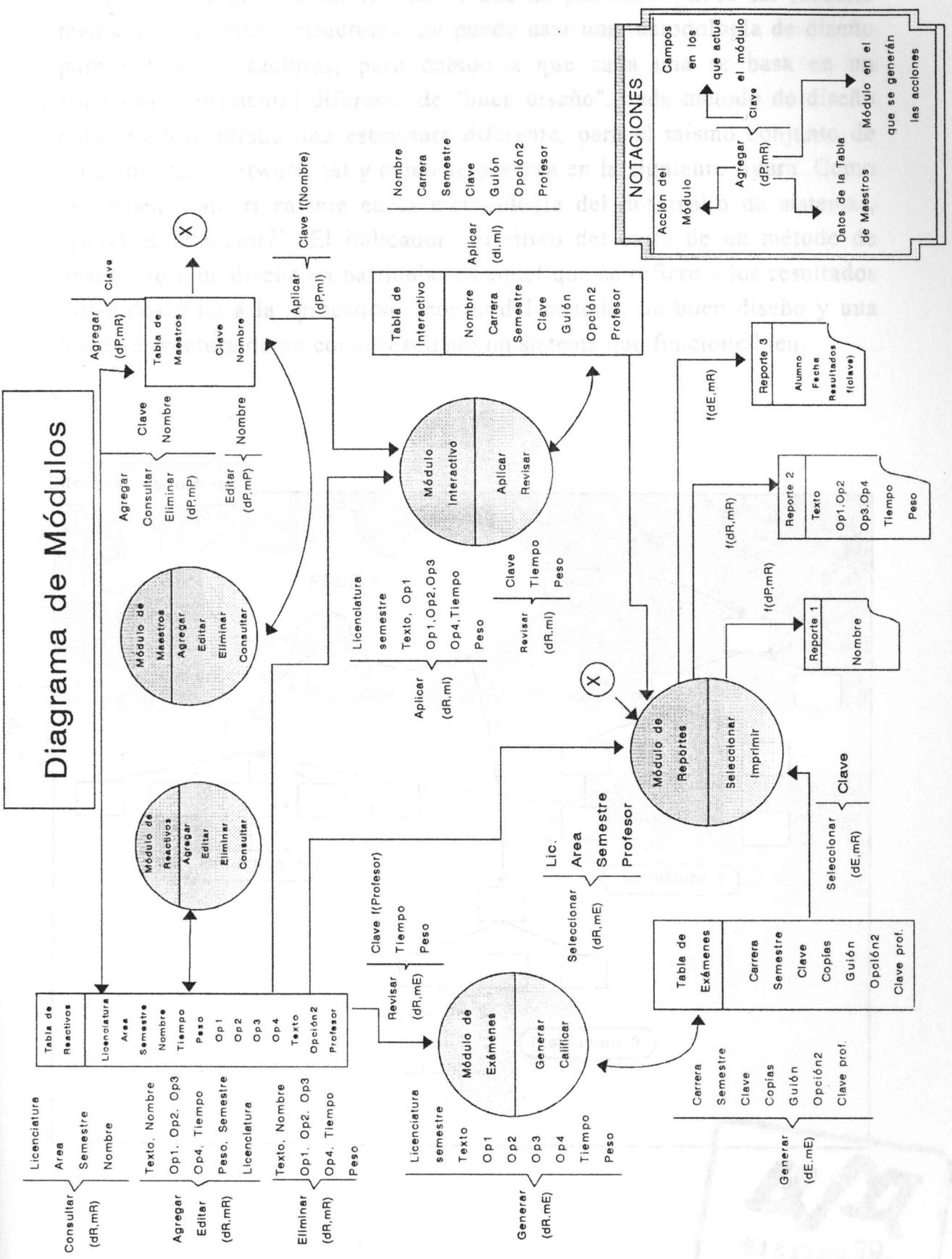
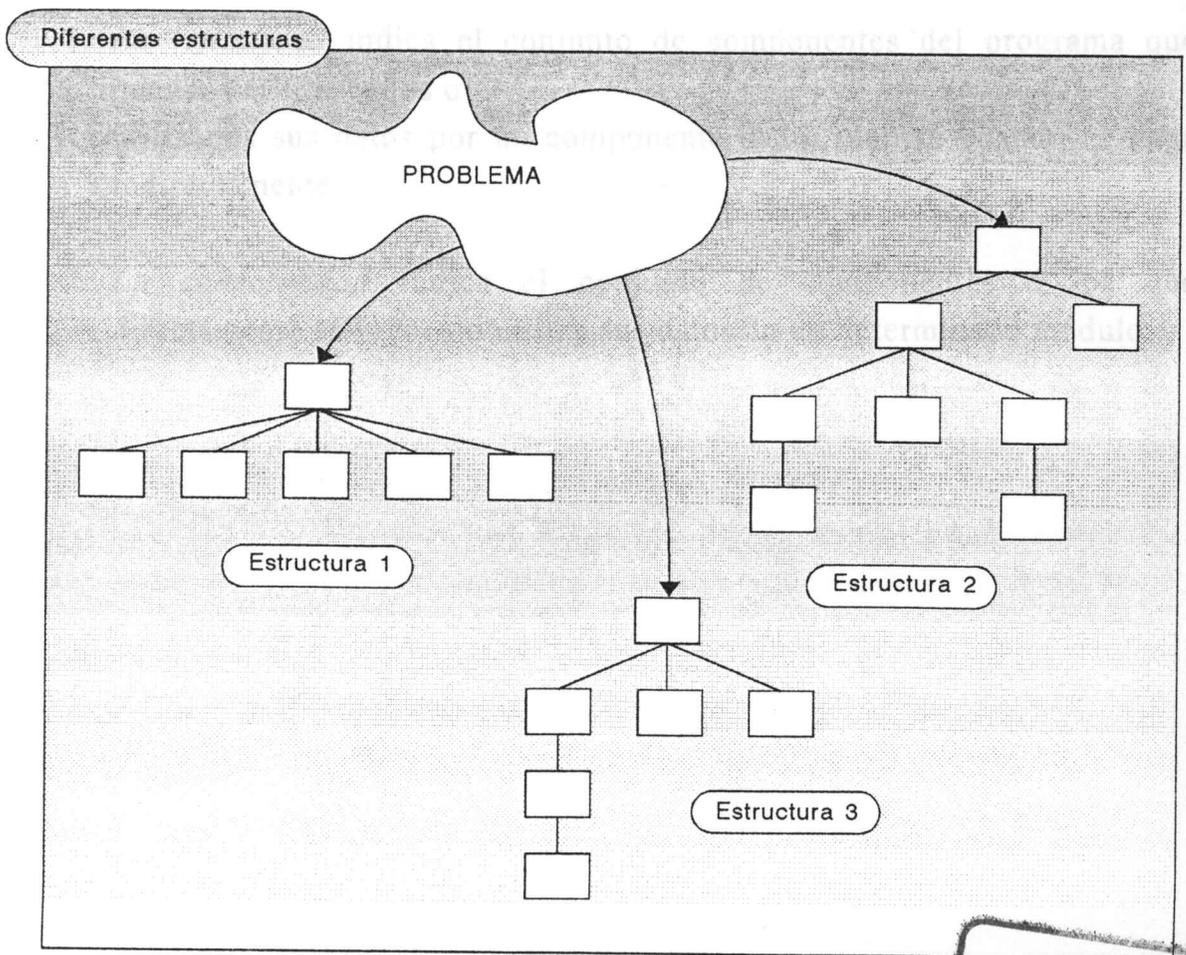


Diagrama de Módulos

Observando la gráfica anterior se ve que un problema puede ser resuelto mediante diferentes estructuras. Se puede usar una metodología de diseño para obtener estructuras, pero debido a que cada una se basa en un concepto fundamental diferente de "buen diseño", cada método de diseño dará como resultado una estructura diferente, para el mismo conjunto de requisitos del software, tal y como se observa en la siguiente figura. Como se comentó anteriormente en la metodología del desarrollo de sistemas, "¿cuál es el mejor?". El indicador definitivo del éxito de un método de desarrollo o de diseño en particular es aquel que se refiere a los resultados obtenidos y no a la "precisión" teórica del método; un buen diseño y una buena estructura darán como resultado un sistema que funcione bien.



Jerarquía de control

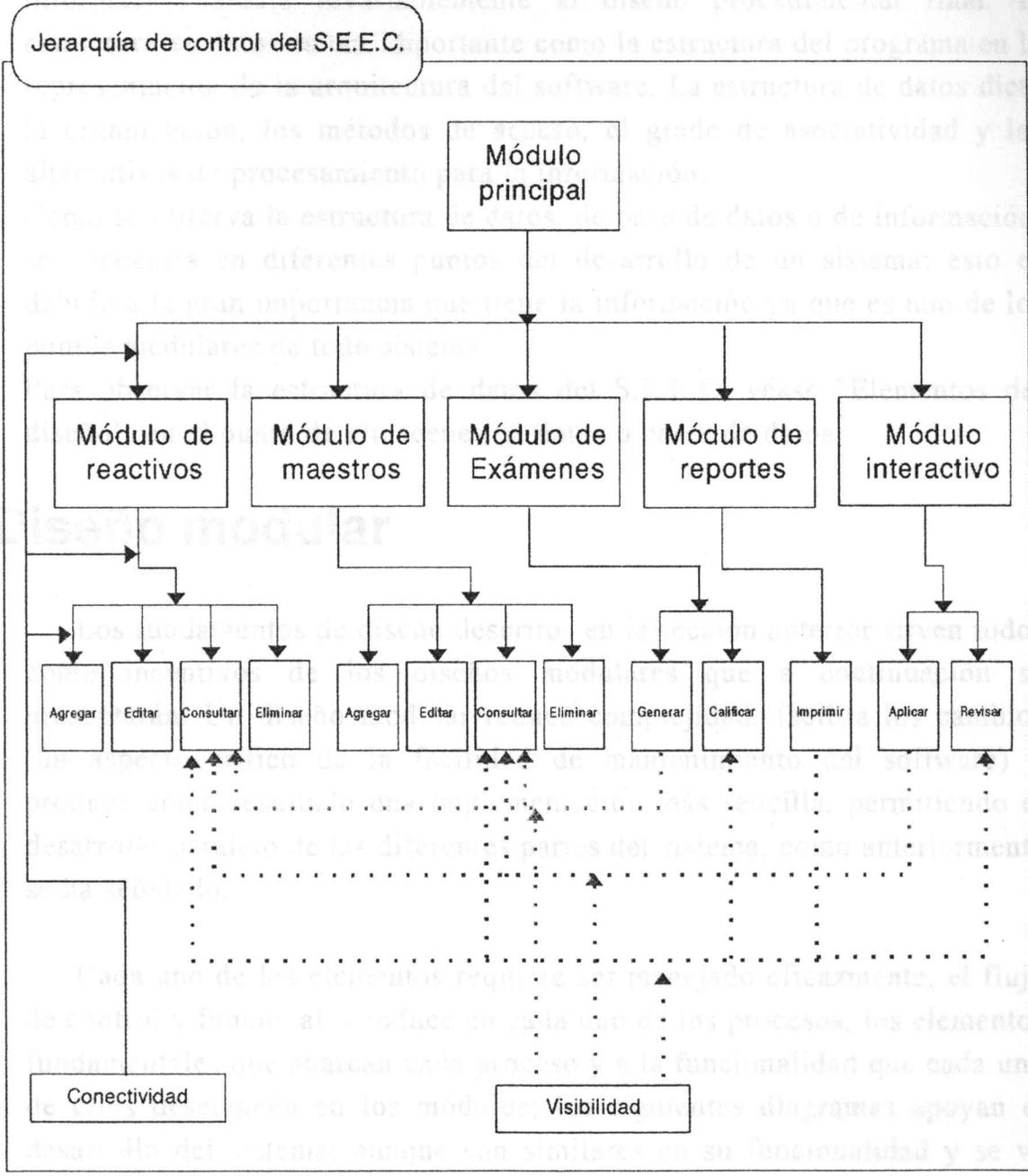
La jerarquía de control, también denominada estructura del programa, representa la organización de los componentes del programa (módulos) e implica una jerarquía de control. No representa aspectos procedimentales del software, tales como la secuencia de procesos la ocurrencia u orden de decisiones o la repetición de las operaciones. La siguiente figura muestra la jerarquía de control del S.E.E.C. Las relaciones de control dentro los módulos se expresan de la siguiente forma: un módulo que controla a otro módulo se dice que es superior a él, e inversamente un módulo controlado por otro se dice que es subordinado del controlador.

La jerarquía de control también representa dos características sutilmente diferentes, de la arquitectura del software: *la visibilidad y la conectividad*.

1. *La visibilidad* indica el conjunto de componentes del programa que pueden ser invocados o utilizados sus datos por un componente dado, incluso cuando se haga indirectamente.
2. *La conectividad* indica el conjunto de componentes a los que directamente se invoca o utiliza sus datos en un determinado módulo.

Estructura de datos

La estructura de datos es una representación de la relación lógica existente entre los elementos individuales de datos. Debido a que la estructura de la información influye inevitablemente al diseño procedimental final, la estructura de datos es una parte importante como la estructura del programa en la arquitectura del software. La estructura de datos dice



Estructura de datos

La estructura de datos es una representación de la relación lógica existente entre los elementos individuales de datos. Debido a que la estructura de la información afecta invariablemente al diseño procedimental final, la estructura de datos es tan importante como la estructura del programa en la representación de la arquitectura del software. La estructura de datos dicta la organización, los métodos de acceso, el grado de asociatividad y las alternativas de procesamiento para la información.

Como se observa la estructura de datos, de base de datos o de información, se encuentra en diferentes puntos del desarrollo de un sistema; esto es debido a la gran importancia que tiene la información ya que es uno de los puntos medulares de todo sistema.

Para observar la estructura de datos del S.E.E.C. véase "Elementos del diseño", en el punto de almacenes de datos o bases de datos.

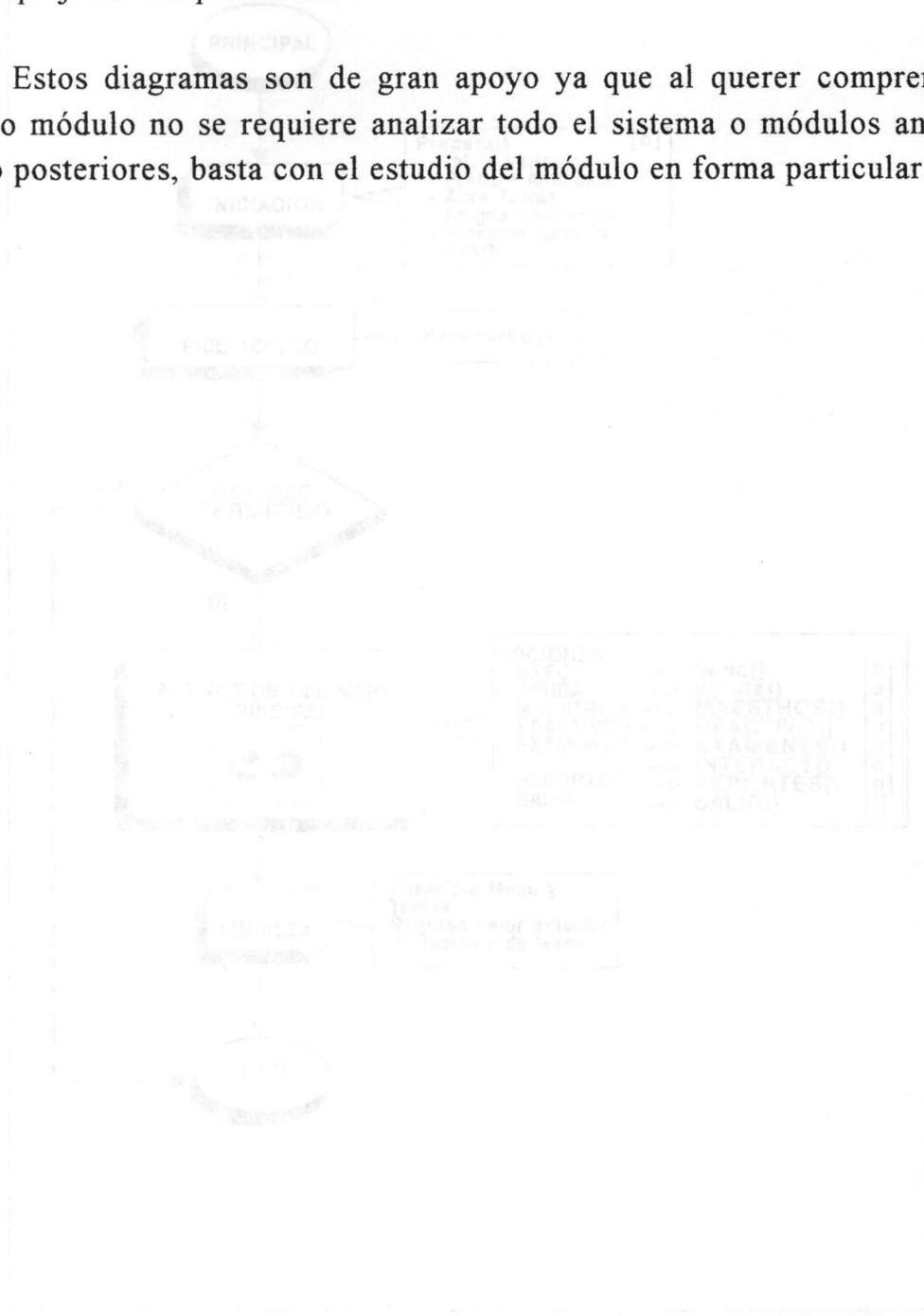
Diseño modular

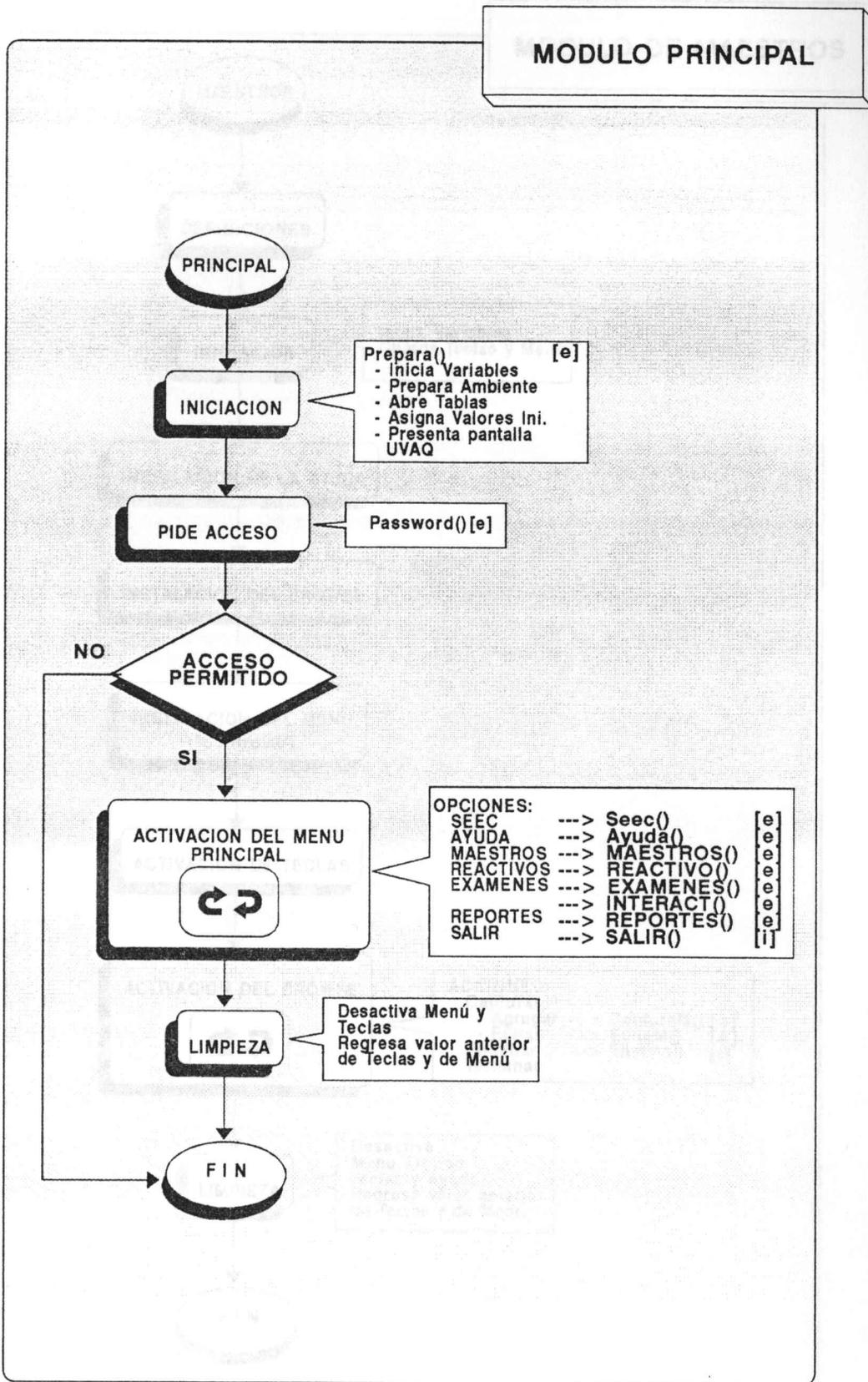
Los fundamentos de diseño descritos en la sección anterior sirven todos como incentivos de los diseños modulares que a continuación se presentarán. Un diseño modular reduce complejidad, facilita los cambios (un aspecto crítico de la facilidad de mantenimiento del software) y produce como resultado una implementación más sencilla, permitiendo el desarrollo paralelo de las diferentes partes del sistema, como anteriormente se ha señalado.

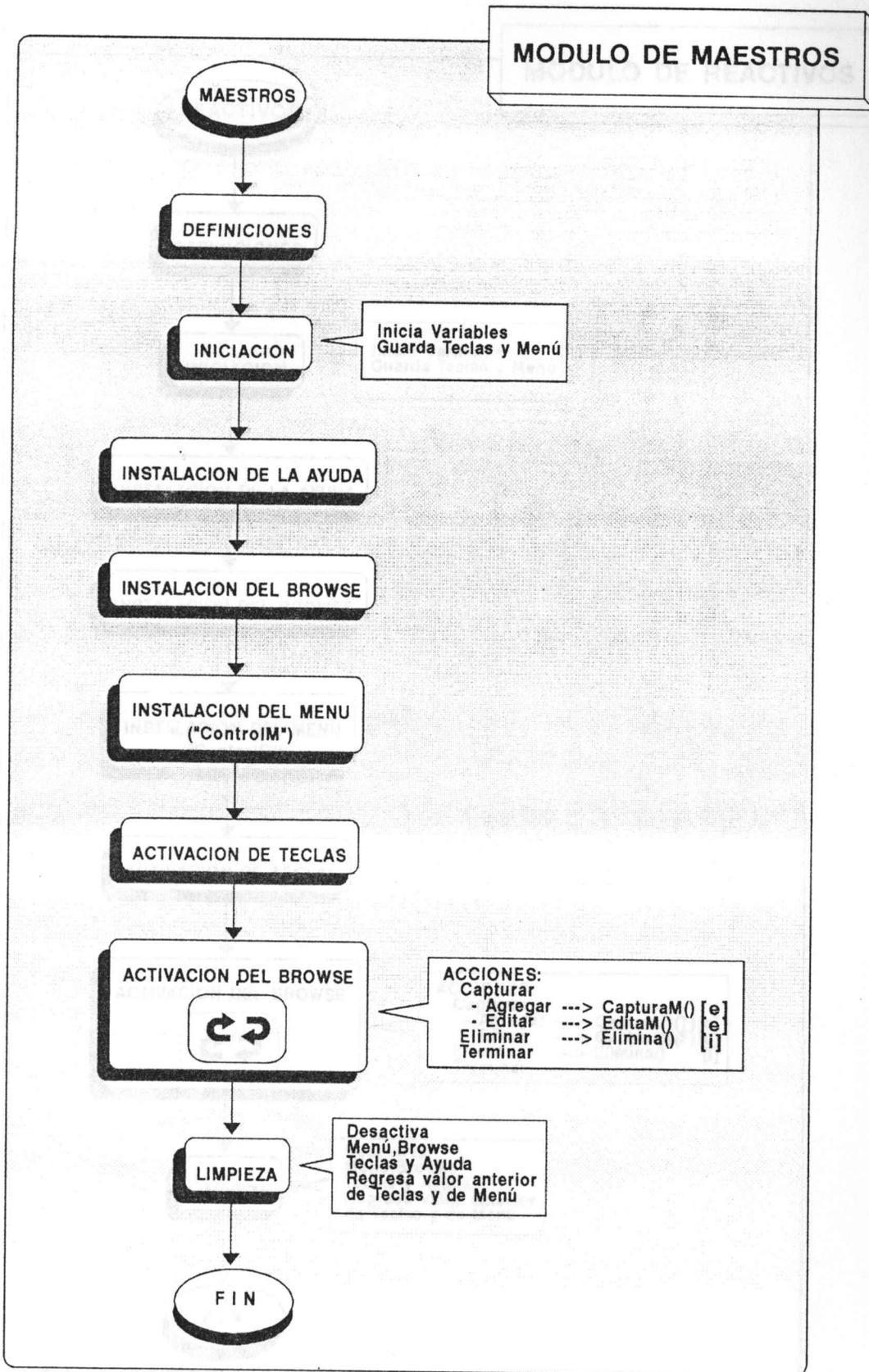
Cada uno de los elementos requiere ser manejado eficazmente, el flujo de control y funcional conduce en cada uno de los procesos, los elementos fundamentales que abarcan cada proceso y a la funcionalidad que cada uno de ellos desempeña en los módulos; los siguientes diagramas apoyan el desarrollo del sistema; aunque son similares en su funcionalidad y se ve reflejado al presentarse, tienen un control individual que facilita su análisis.

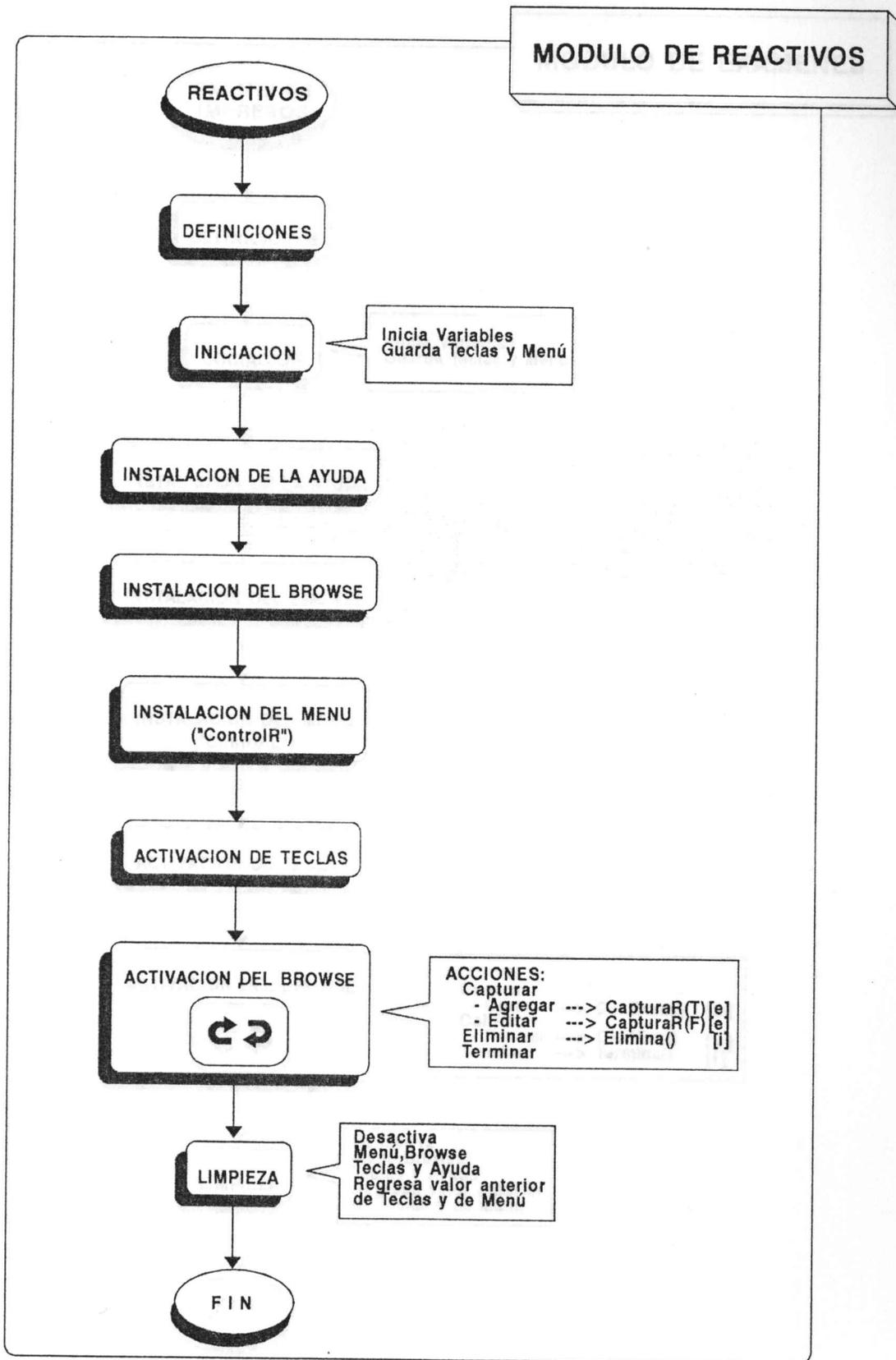
Su disposición es jerárquica, cada actividad se representa con una caja, las flechas indican el flujo del control además de indicar que para realizar un proceso se requiere haber realizado el anterior, las cajas con referencia invocan opciones, funciones o acciones que tienen cierta actividad dentro del proceso, lo cual se encuentra integrado en un nivel inferior de módulos, subtareas, funciones o procedimientos que reducen al mínimo la complejidad del problema.

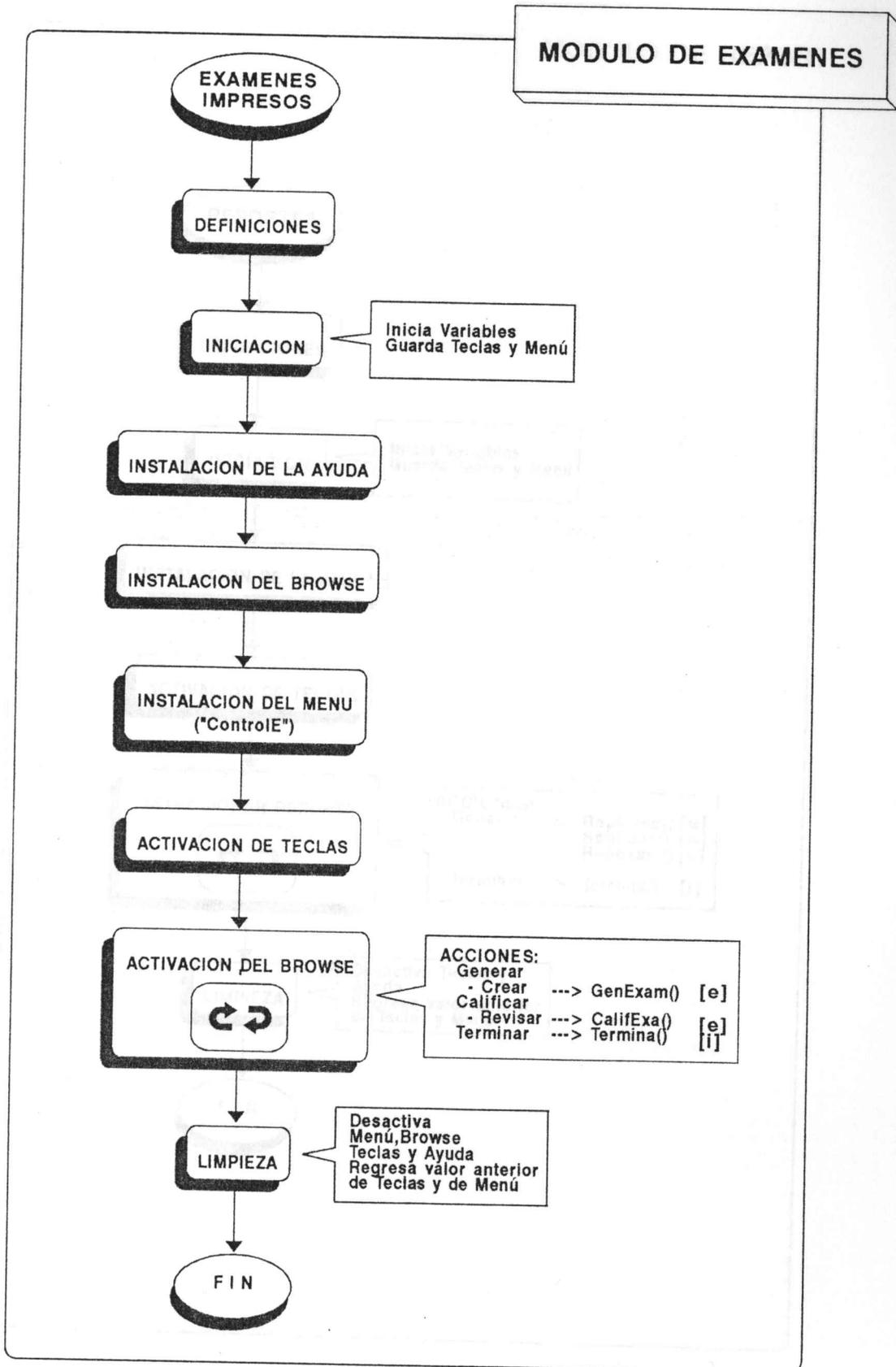
Estos diagramas son de gran apoyo ya que al querer comprender un solo módulo no se requiere analizar todo el sistema o módulos anteriores y/o posteriores, basta con el estudio del módulo en forma particular.

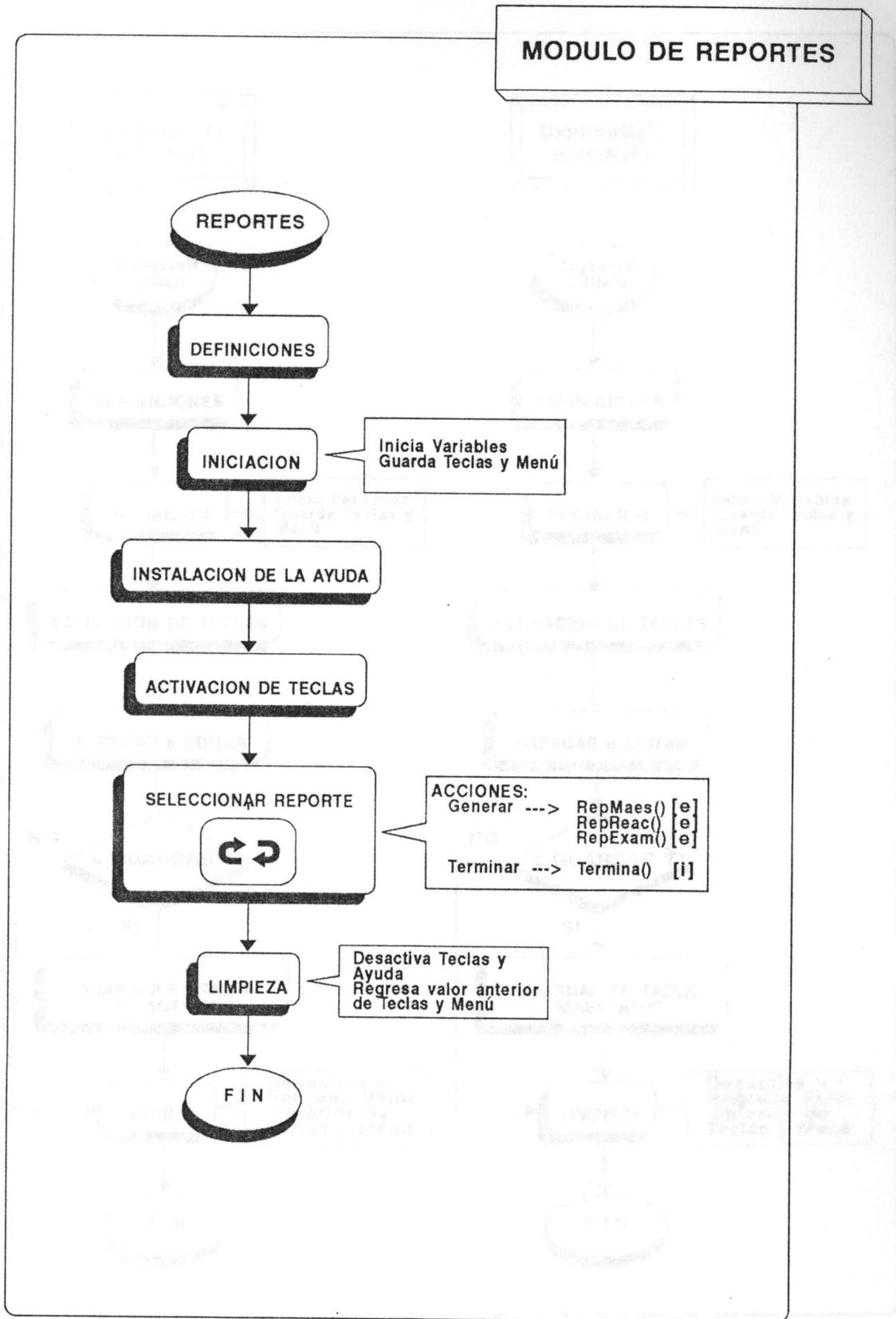


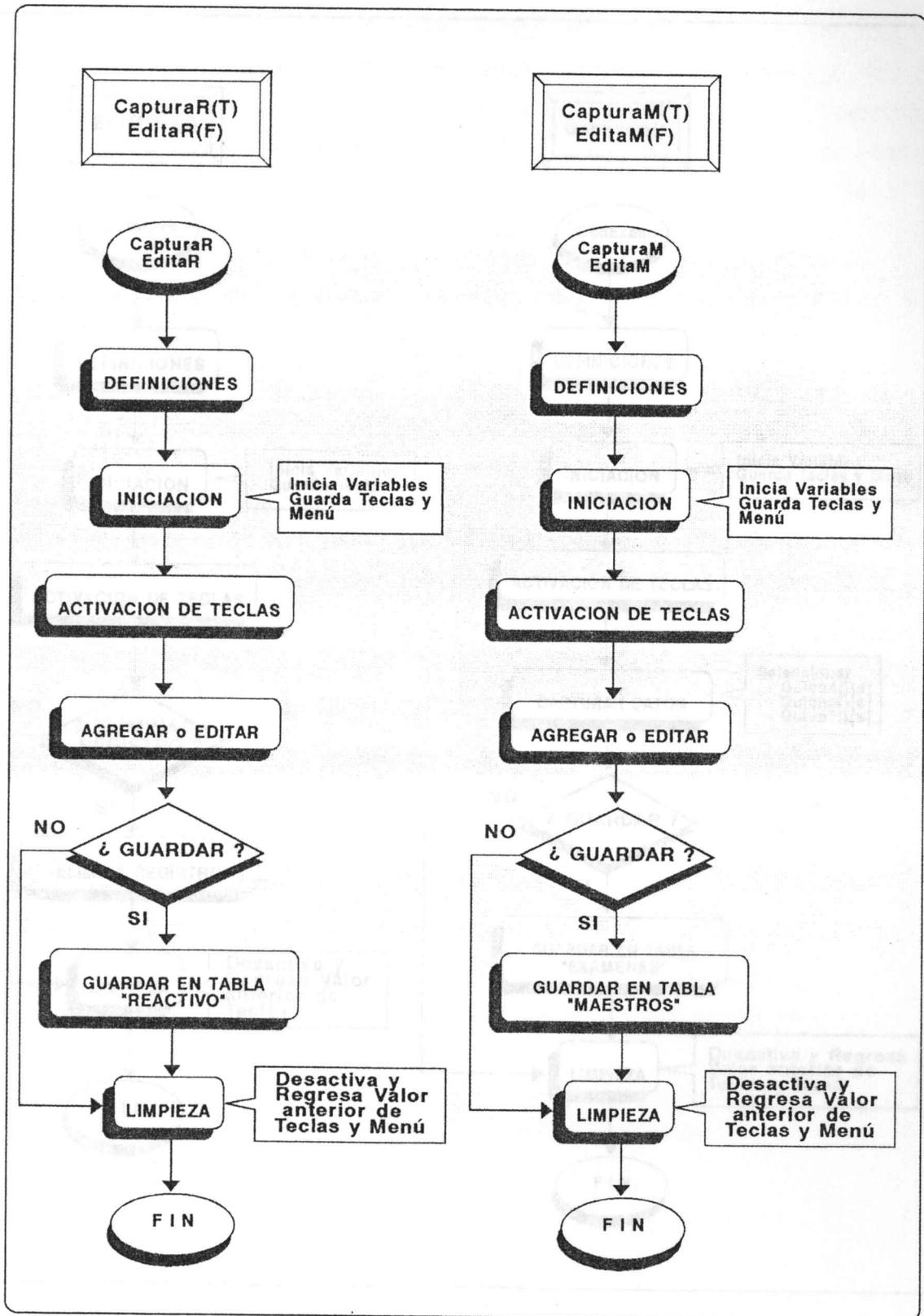


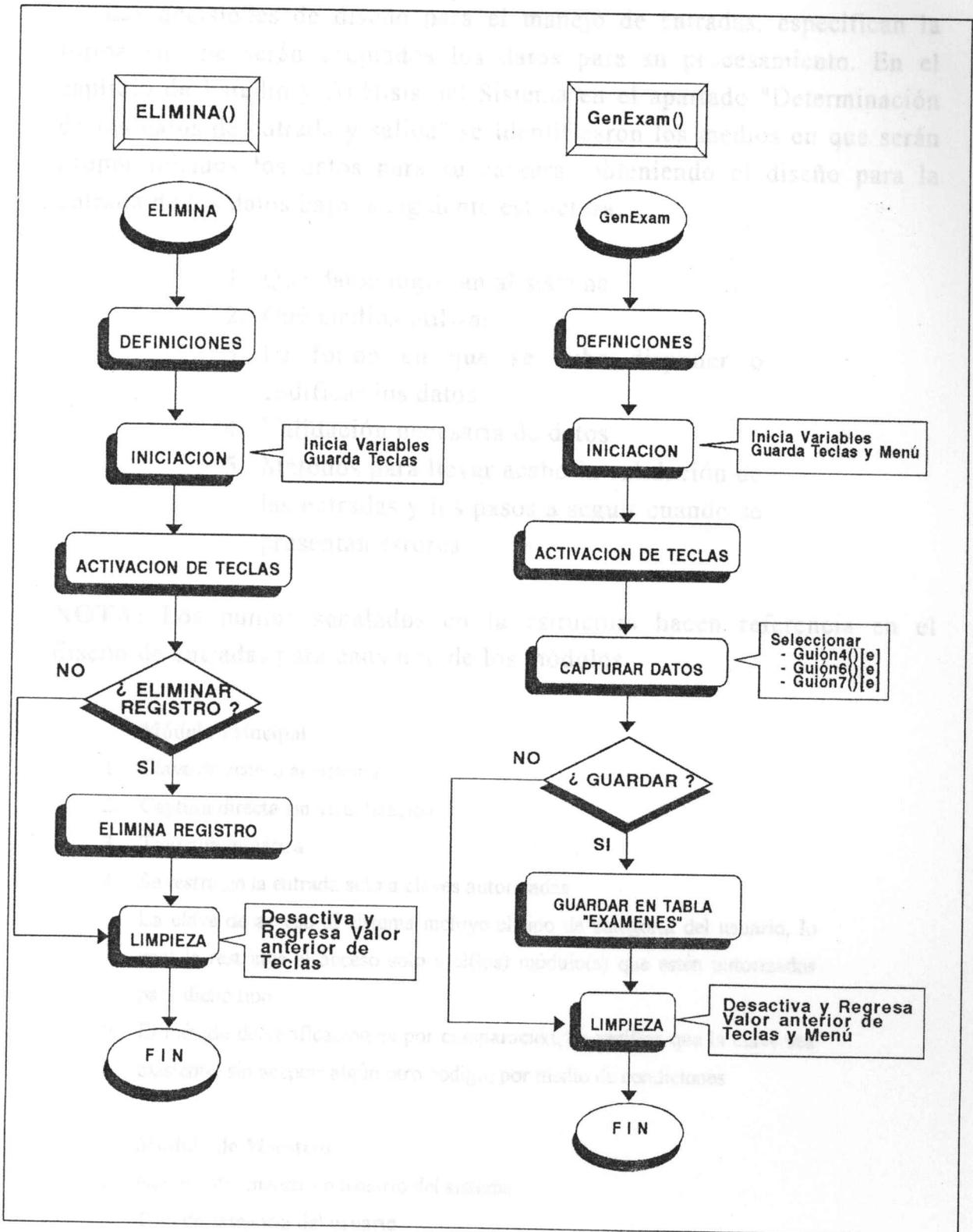












Diseño de la entrada

Las decisiones de diseño para el manejo de entradas, especifican la forma en que serán aceptados los datos para su procesamiento. En el capítulo de Estudio y Análisis del Sistema en el apartado "Determinación de los datos de entrada y salida" se identificaron los medios en que serán proporcionados los datos para su captura, obteniendo el diseño para la entrada de los datos bajo la siguiente estructura:

1. Qué datos ingresan al sistema
2. Qué medios utilizar
3. La forma en que se debe disponer o codificar los datos
4. Validación necesaria de datos
5. Métodos para llevar a cabo la validación de las entradas y los pasos a seguir cuando se presentan errores.

NOTA: Los puntos señalados en la estructura hacen referencia en el diseño de entradas para cada uno de los módulos.

Módulo Principal

1. Clave de acceso al sistema
2. Captura directa sin visualización
3. Tipo alfanumérica
4. Se restringe la entrada solo a claves autorizadas

La clave de acceso al sistema incluye el tipo de categoría del usuario, lo cual le restringe el acceso solo a el(los) módulo(s) que estén autorizados para dicho tipo

5. El método de verificación es por comparación, se verifica que la clave sea existente, sin aceptar algún otro código, por medio de condiciones

Módulo de Maestros

1. Nombre del maestro o usuario del sistema
Tipo de categoría del usuario
Clave de acceso

2. Para el dato del nombre del usuario captura directa
Para el Tipo de usuario por medio de selección predefinida con las tres categorías.
La clave de acceso es generada aleatoriamente por una función con la categoría como parámetro.
3. El *Nombre* es tipo carácter
La *Clave* es de tipo alfanumérica
El *Tipo* de categoría es numérico
4. No se puede generar el registro sin el nombre
El *Tipo* se encuentra inicializado
La clave se genera automáticamente con el dato *Tipo*
5. Se verifica por comparación el campo *Nombre* antes de almacenar el registro, si se encuentra vacío, ocurre un mensaje de advertencia, no permite almacenar el registro si el dato es omitido
El campo *Tipo* es inicializado
En el campo *Clave* se verifica por comparación, que no exista otra igual, si lo está genera otra

Módulo de Reactivos

1. Reactivo o pregunta
 - Opciones a, b, c y d, de solución al reactivo
 - La Carrera a la que pertenece el reactivo
 - El Área a la que pertenece el reactivo
 - El Semestre correspondiente al reactivo
 - Nombre corto de referencia del reactivo
 - Tiempo de duración de respuesta del reactivo
 - El Peso que sirve para la ponderación del reactivo
2. Para el campo del reactivo, opciones a, b, c, d, semestre, nombre corto de referencia, peso y tiempo son de captura directa
Para la Carrera y el Área medio de selección predefinidas.
3. Reactivo es de tipo Memo
 - Opciones a, b, c y d, tipo carácter
 - La Carrera tipo numérico
 - El Área tipo numérico
 - El Semestre tipo numérico
 - Nombre corto de referencia tipo carácter

Tiempo tipo numérico

El Peso tipo numérico

4. No se puede generar el registro sin: el reactivo, opciones y nombre del reactivo

Los campos: Carrera, Área y Semestre se encuentran inicializados

Los campos numéricos no pueden ser negativos

5. Se verifican por comparación los campos de captura directa antes de almacenar el registro, si se encuentra alguno de ellos vacío, ocurre un mensaje de advertencia, no permite almacenar el registro si el dato es omitido

Con la inicialización de los campos: Carrera, Área y Semestre se valida la información, además de ser verificada si es omitida.

Módulo de Exámenes

1. La Carrera a la que se aplicara el examen

El Semestre al que va dirigido el examen

El número de copias o exámenes que vayan a aplicarse

El Guión que sirve para la elección del número de preguntas por área

Las Respuestas de los exámenes a calificar

Nombre de los alumnos a los que se le s aplico el examen

2. La Carrera y el Semestre son de selección predefinidas

El número de copias y el guión de captura directa

Los Resultados y el Nombre de captura directa

3. El campo Carrera tipo numérico

El Semestre tipo numérico

El número de copias tipo numérico

Las preguntas del guión tipo numérico

Los resultados y el Nombre de tipo carácter

4. No se puede generar el examen sin el número de copias

No permite valores negativos en las preguntas del guión

Los campos: Carrera y Semestre se encuentran inicializados

No permite generar exámenes si el total de preguntas es igual a cero

No se puede calificar exámenes sin el Nombre

5. Se verifica por comparación el campo de número de copias antes de pasar a la captura del guión

Con la inicialización de los campos: Carrera y Semestre se valida la información y se verifica si es omitida

Se realizan operaciones con condiciones para evitar que los valores sean menores o igual a cero

Para calificar los exámenes se verifica por comparación que el campo Nombre no se encuentre vacío

Módulo Interactivo

1. Nombre del alumno al que se aplicará el examen

La Carrera y Semestre a los que pertenece

El Guión que sirve para la elección del número de preguntas por área

2. El nombre es de captura directa

La Carrera y el Semestre son de selección predefinidas

El guión es de captura directa

3. El campo Nombre es de tipo carácter

El campo Carrera tipo numérico

El Semestre tipo numérico

Las preguntas del guión tipo numérico

4. No permite valores negativos en las preguntas del guión

Los campos: Carrera y Semestre se encuentran inicializados

No permite generar exámenes si el total de preguntas es igual a cero

5. Se verifica por comparación si el campo Nombre se encuentra vacío y

si lo esta no se puede aplicar el examen

Con la inicialización de los campos: Carrera y Semestre se valida la información además de verificar si es omitida

Se realizan operaciones con condiciones para evitar que los valores sean menores o igual a cero

En todas las entradas de datos se utilizan los métodos de verificación y validación de la información, además de contar con dos opciones de validación final, que son las de **Guardar** o **Cancelar**, las cuáles permiten a los usuarios **confirmar** los datos o los procesos.

Diseño de la salida

El término salida se refiere a los resultados e información que se genera con el sistema. Para muchos usuarios, la salida es la única razón para el desarrollo del sistema y la base sobre la que ellos evaluarán la utilidad de la aplicación. Esto es debido a que pueden existir usuarios que no operen el sistema, pero que utilicen la salida.

Tomando en cuenta lo anterior se tomaron los siguientes puntos:

- Determinar que información presentar.
- Decidir si la información será presentada en forma visual en pantalla, impresa o algún otro medio.
- Disponer la presentación de la información en un formato aceptable.
- Decidir cómo distribuir la salida entre los posibles destinatarios.

El diseño de la salida se encuentra distribuido de la siguiente forma:

Diseño de salidas en pantalla

El diseño de las salidas en pantalla se estructuró de la siguiente forma: Se desarrollaron ventanas de consulta general que se presentan al entrar a cada módulo, dicha ventana puede ser ampliada o minimizada según lo requiera el usuario, la filosofía que se tomó es evitar pasos innecesarios a los usuarios para obtener esta información, se utilizan ventanas para el procedimiento de cada proceso, lo cual simplifica la operación del sistema, las ventanas pueden ser colocadas en diferentes partes de la pantalla, con esto el usuario visualiza completamente cada uno de los procesos y la forma en que estos se llevan a cabo.

Diseño de las salidas impresas

En el diseño de las salidas impresas, únicamente se diseñaron los datos que son de utilidad para los usuarios, evitando extensos reportes e información innecesaria.

S.E.E.C.

istema dición entana ovimientos 8:49:57 am

Elabora	Carrera	Semestre	Fecha	Examen
JAMES A. SENN	Administración	Noveno	21/11/1996	EXA00001
JAMES A. SENN	Administración	Noveno	23/11/1996	EXA00002
JAMES A. SENN	Administración	Noveno	26/11/1996	EXA00003

02/12/1996

Ventana de Browse interactivo del módulo de exámenes.

S.E.E.C.

istema dición entana ovimientos 8:59:29 am

Control de Reactivos

La naval, S.A. presenta los siguientes datos:

	Importe
Costo total de producción	N\$ 895
Costo de los artículos disponibles para su venta	N\$ 725
Inventario final de productos terminados	N\$ 590
Inventario final de productos en proceso	N\$ 170
Inventario inicial de preectos terminados	N\$ 0

Con base en lo anterior, el costo de la producción vendida es:

<p>Opciones</p> <p>a. \$ 125</p> <p>b. \$ 560</p> <p>c. \$ 340</p> <p>d. \$ 410</p>	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;">Nombre</td> <td colspan="3">COSTO DE PRODUCCION</td> </tr> <tr> <td>Tiempo</td> <td>8</td> <td>Peso</td> <td>8</td> </tr> <tr> <td>Carrera</td> <td colspan="3">Administración</td> </tr> <tr> <td>Area</td> <td colspan="3">Adnón. de Personal</td> </tr> </table>	Nombre	COSTO DE PRODUCCION			Tiempo	8	Peso	8	Carrera	Administración			Area	Adnón. de Personal		
Nombre	COSTO DE PRODUCCION																
Tiempo	8	Peso	8														
Carrera	Administración																
Area	Adnón. de Personal																

« Guardar »
< Cancelar >

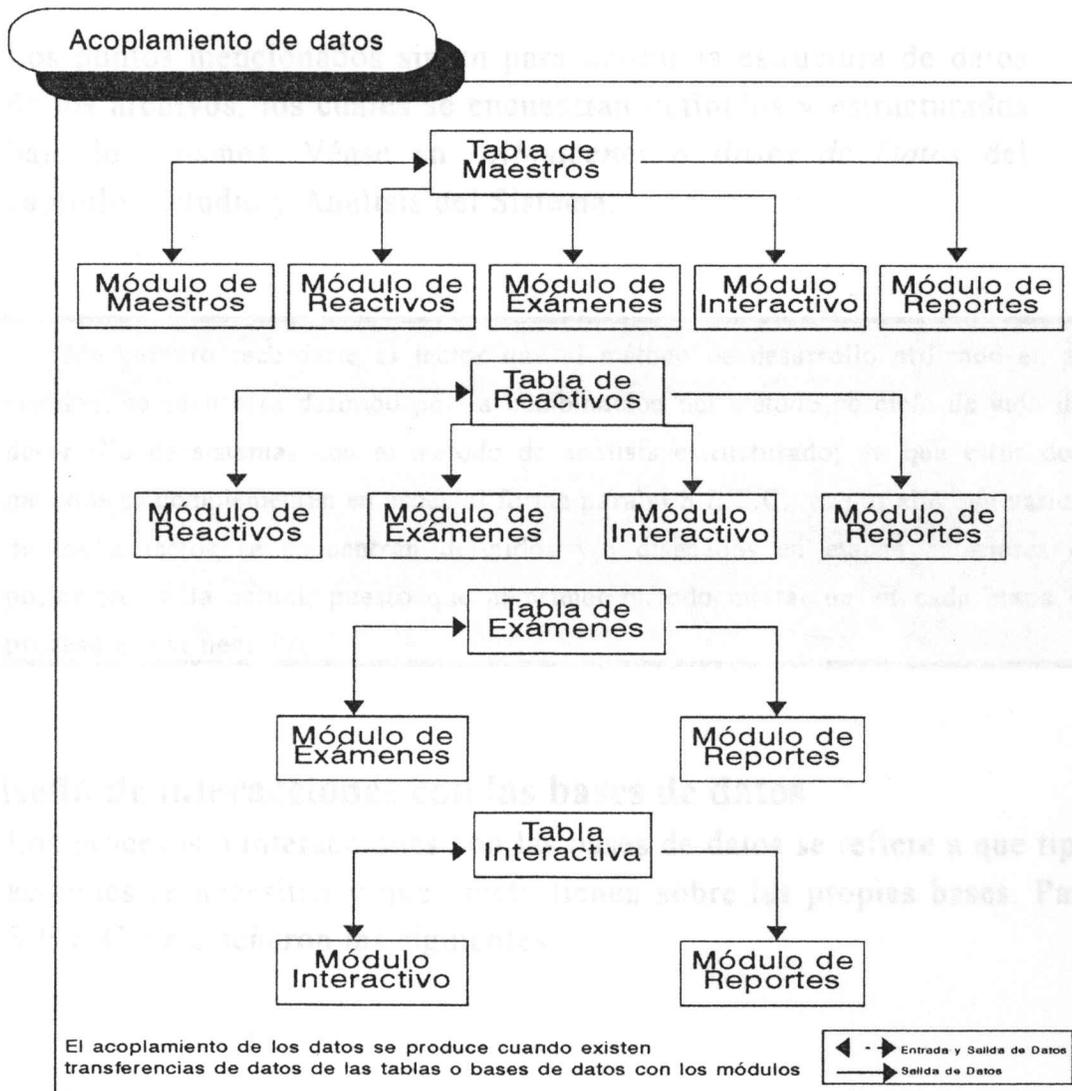
02/12/1996

Ventana de captura y edición de reactivos.

Diseño de archivos

Como se observó en "Los elementos del diseño", el diseño de los archivos se encuentra bajo las estructuras que se requieren para cada aplicación y las operaciones que se realizan con cada uno de ellos. Uno de los aspectos importantes es el de identificar los módulos que operan directamente con los archivos, a lo cual se le denomina acoplamiento de datos. El acoplamiento de datos es la transferencia de datos o parámetros de un módulo a otro, esto se realiza identificando la transferencia de información entre los módulos.

Por lo cual tenemos el siguiente diagrama.



Apoyados por el diagrama de acoplamiento de datos y teniendo en cuenta que datos se requerían para cada aplicación, fue más sencillo definir los archivos, y esto se realizó con los siguientes puntos:

- Los datos que deben incluirse en el formato de los registros contenidos en el archivo.
- La longitud y el tipo de cada registro, con base en las características del dato que contiene o de la función que desempeñara dentro del sistema.
- La secuencia a disposición de los registros dentro del archivo.

Los puntos mencionados sirven para definir la estructura de datos de los archivos, los cuáles se encuentran definidos y estructurados bajo los mismos. Véase en *Almacenes o Bases de Datos* del capítulo Estudio y Análisis del Sistema.

Me permito recordarle al lector que el método de desarrollo utilizado en el sistema, se encuentra definido por la combinación del método de ciclo de vida de desarrollo de sistemas con el método de análisis estructurado; ya que estos dos métodos se complementan en especial forma para el S.E.E.C.; es por ello que varios de los aspectos se encuentran definidos y/o diseñados en etapas anteriores o posteriores a la actual, puesto que el primer método interactúa en cada etapa o proceso que se necesite.

Diseño de interacciones con las bases de datos

Los procesos o interacciones con las bases de datos se refiere a que tipo de acciones se necesitan y que efecto tienen sobre las propias bases. Para el S.E.E.C. se diseñaron las siguientes:

Básicas:

Agregar.- Es una de las actividades que consisten en dar de altas registros, esto es agregar un registro a la base de datos correspondiente. Esta interacción integra los datos que el usuario captura, de tal forma que este conjunto que conforma un registro, sea añadido a la base de datos a la que pertenecen y poder ser ocupados posteriormente.

Editar.- Esta interacción permite el acceso a la información que se encuentra en las bases de datos, de tal forma que se pueda consultar y modificar si así se desea. Lo que permite al editar los datos, es proporcionar una alternativa de modificación de éstos junto con la posibilidad de cancelar dicho acto, según crea conveniente el usuario. Si los datos son modificados y aceptados la base de datos se actualizará con la información reciente.

Eliminar.- Permite al usuario excluir la información de la base de datos disponible. Toda aquella información que sea eliminada no aparecerá en las bases de datos, de tal forma que no se puede tener acceso nuevamente a esa información. En los módulos se presentan las funciones de eliminar dando una posibilidad de cancelar la eliminación, con esto la información queda intacta y disponible para alguna otra operación.

Consultar.- A diferencia de la edición, la consulta permite sólo el acceso a la información en la pantalla, sin poder realizar ningún cambio en ella, es decir sólo le permite revisar pero no modificar.

Específicas:

Generar.- Permite crear de forma particular los exámenes. Lo que realiza es conjuntar cierta información agrupando los elementos necesarios y así adecuarlos para formar un examen, es decir reúne los datos en los cuales se describe las características que permiten crear o generar un examen.

Calificar.- Ofrece un método para poder revisar los exámenes ya antes aplicados, su principal objetivo es la revisión de los exámenes, ya sean impresos o interactivos.

Aplicar.- Permite lo mismo que la de generar pero para los exámenes interactivos.

Revisar.- Permite la revisión de los exámenes interactivos, lo que realiza es visualizar los datos de los resultados.

Diseño de Controles

Uno de los aspectos que no se deben omitir es la anticipación a los errores, los cuales pueden cometerse al ingresar los datos en el sistema o al solicitar la ejecución de ciertas interacciones. Algunos errores no tienen importancia ni consecuencias, pero otros pueden ser tan serios que ocasionarían el borrado de los datos o el uso inapropiado del sistema. Aunque exista sólo una probabilidad de cometer un error serio, se debe ofrecer los medios para detectar y manejar el error; por lo cual se diseñaron los siguientes:

- 1. Se asegura que sólo tengan acceso al sistema los usuarios autorizados**
- 2. Se garantiza que las transacciones son aceptables**
- 3. Se valida los datos y se comprueba su exactitud**
- 4. No permite conclusión de interacciones si se han omitido datos**

5. Se genera automáticamente la clave de acceso
6. El acceso a los módulos del sistema se restringe a la categoría del usuario
7. Las bases de datos son protegidas por un código, evitando su uso inapropiado
8. Existen procedimientos para el manejo de errores externos
9. Hay un control de módulos subordinados

Cada uno de los controles tiene un método de verificación, validación o acción para el control de errores o protección de la información, según sea el caso; dichos métodos se encuentran estructurados bajo el criterio de asegurar la efectividad y calidad del sistema.

Diseño de Procedimientos

Los procedimientos especifican qué tareas deben efectuarse al utilizar el sistema y quiénes son los responsables de llevarlas a cabo. Dichos procedimientos se encuentran diseñados bajo los siguientes puntos:

- ◆ Procedimientos para entrada de datos
- ◆ Procedimientos durante la ejecución
- ◆ Procedimientos para el manejo de errores
- ◆ Procedimientos de seguridad y respaldo

Procedimientos para entrada de datos

Los métodos y secuencia para la captura de datos, las interacciones y el ingreso al sistema son los siguientes:

Para entrar al sistema se necesita teclear la clave de acceso, la cual permite al usuario según su tipo de categoría utilizar los módulos del mismo.

El acceso al módulo de Maestros se encuentra disponible únicamente para los usuarios de categoría: Administrador.

Los usuarios con ésta categoría tienen el acceso para el control de los datos de los usuarios, lo cuál se realiza eligiendo del menú Sistema la opción Maestros, esta opción agrega un menú de Movimientos que permite las interacciones: Agrega, Edita, Elimina y Termina junto con el catálogo de Maestros, en donde aparecen los datos de los usuarios.

En la interacción Agrega se tienen tres opciones predeterminadas para la selección del tipo de usuario que se dará de alta (Administrador, Maestro o Capturista), se puede utilizar el ratón o el teclado para elegir cualquiera de las opciones y automáticamente se le asigna una clave, el nombre se captura por medio del teclado; se puede guardar o cancelar la operación.

Para la interacción Edita, se debe seleccionar al usuario en el catálogo y elegir Edita del menú Movimientos, de esta forma presentará una ventana con el nombre del usuario permitiendo modificarlo y así guardarlo o cancelarlo, de la misma forma que en Agrega.

En la interacción Elimina se elige de igual manera al usuario que se pretende eliminar del catálogo y elegir Elimina del menú Movimientos, apareciendo una ventana que presenta el mensaje de ¿Eliminar Usuario?, permitiendo confirmar o cancelar la operación.

La operación Termina cierra el catálogo de Maestros y regresa al menú Sistema.

Para los datos de los reactivos, accesibles a las tres categorías, se elige Reactivos del menú Sistema, apareciendo el catálogo de éstos, en donde se pueden

consultar sus datos generales (Carrera, Área, Semestre y Nombre del reactivo), además del menú Movimientos que

al igual que el de Maestros permite las interacciones: Agrega, Edita, Elimina y Termina.

En la interacción Agrega se capturan los datos: reactivo, opciones (a, b, c y d), nombre, tiempo, peso y semestre por medio del teclado y carrera y área ya sea por medio del ratón o del teclado.

Las interacciones Edita y Elimina realizan el mismo proceso que las de Maestros.

Termina finaliza el módulo de Reactivos cerrando su catálogo y regresando al menú Sistema.

Para el control de los exámenes, restringido para los Capturistas, se elige Exámenes del menú Sistema, el cuál presenta un submenú con las opciones: Impresos e Interactivos.

Al seleccionar impresos presenta el catálogo de los exámenes junto con el menú Movimientos, que muestra las interacciones Generar y Calificar.

Al seleccionar Generar aparece una ventana que permite especificar el número de copias (número de exámenes a generar), carrera y semestre al que se aplicará el examen, a diferencia del número de copias, carrera, semestre y el guión se pueden seleccionar también por medio del ratón; el guión presenta otra ventana en donde aparecen las áreas a quienes va dirigido el examen y espera se capturen el número de preguntas para cada área, ya realizado este procedimiento se puede generar o regresar a la ventana anterior donde se pueden actualizar los datos antes seleccionados, hasta generar o cancelar el proceso.

Para los Reportes, también restringido a los Capturistas, existen cuatro opciones (Impresora, Maestros, Reactivos y Exámenes).

Seleccionando Impresora aparece una ventana donde se puede configurar la impresora con la que se generaran los reportes.

Al elegir la interacción de Maestros, accesible solo para el Administrador, se generan los reportes de los usuarios con los datos: clave, nombre y categoría.

Al seleccionar Reactivos, no accesible a los Capturistas, se presenta una ventana donde se eligen en que forma serán generados los reportes por: carrera, área y/o semestre.

Al escoger la opción Exámenes, también restringida para los Capturistas, aparece una ventana en donde se debe especificar el tipo de reporte (exámenes o resultados) y la carrera, semestre y fecha del examen.

Procedimientos durante la ejecución

Los pasos y acciones que deberán ser emprendidos por los operadores o usuarios del sistema para alcanzar los resultados deseados son definir y/o preparar:

Para dar de alta usuarios.- El usuario o usuarios a dar de alta y las categorías correspondientes.

Para agregar reactivos.- El texto o pregunta, sus 4 opciones de solución (teniendo la solución en el inciso "a"), un nombre corto que lo identifique, la carrera, área y semestre al que pertenece, el tiempo de solución y el peso de ponderación del reactivo.

En la generación de exámenes impresos.- La carrera, semestre, número de exámenes, preguntas por áreas y

tiempo de duración del examen y que la impresora se encuentre encendida y lista para imprimir.

En la aplicación de exámenes interactivos.- Que la red se encuentre activa y cada una de las terminales en las que se realizarán los exámenes y la carrera, semestre, número de exámenes, preguntas por áreas y tiempo de duración de los exámenes

En la generación de reportes.- Que la impresora se encuentre encendida y lista para imprimir.

Para crear respaldos.- Contar con disquetes preformateados y/o con espacio disponible para el almacenamiento de las bases de datos.

Procedimientos para el manejo de errores

Las acciones a seguir si se presentan errores inesperados durante alguna ejecución son:

Si la impresora se bloquea.- Apagar la impresora e intentar de nuevo la aplicación, si recurre a lo mismo, salirse del sistema en forma normal es decir con la opción **salir**, sin apagar el equipo o dar reset ya que esto podría traer errores al sistema con el control de las bases de datos.

Si el espacio en disco es insuficiente.- Cancelar la aplicación, salir del sistema y proporcionar espacio para que el almacenamiento de los datos sea completado, reiniciar el sistema y la aplicación.

Si hay un error al intentar leer datos de un archivo.- Cancelar la aplicación, salir del sistema y reinicie, si

reincide, anote el problema y consulte al administrador del sistema ya que esto pudiera ser por diferentes razones, como es el tener funcionando el sistema al mismo tiempo que otras aplicaciones, como windows por ejemplo.

Si el sistema se bloquea.- Es decir se encuentre en un estado en el que no se pueda avanzar o cancelar alguna aplicación, es muy poco probable, aunque depende del manejo de los usuarios al utilizarlo, y por ello no debe ser descartado; si llega a ocurrir éste problema, anote el procedimiento que se realizó hasta encontrarse en tal situación y coméntese al administrador del sistema.

Algunos errores son inesperados por diferentes razones, como es el perder datos por encontrarse el equipo contaminado por virus, por eliminación de los mismos por error, que se vaya la luz al encontrarse en funcionamiento, etc, problemas externos que pueden afectar el buen funcionamiento del sistema; es por eso que dichos errores suelen ser inesperados, si algún error llegará a ocurrir coméntese al administrador del sistema.

Procedimientos de seguridad y respaldo

Como se comento anteriormente existen diversas razones por las cuales el sistema podría tener problemas y para ello existen acciones que protegen el sistema y sus recursos contra posibles daños, esto es tratando de prevenir perdidas de información así como un mal funcionamiento del sistema.

Para que el sistema se pueda mantener bajo los parámetros de buen funcionamiento se deben realizar respaldos o copias de los archivos maestros así como de las bases de datos.

Con respecto a los archivos maestros del sistema, es preferible realizar dos copias: La primera que tenga como

funciones: la instalación del sistema en el equipo de trabajo y de apoyo para recobrar archivos perdidos o dañados. La segunda, previniendo el daño de algún archivo de la primera copia.

¿Cuándo se deben realizar las copias?

R= Es preferible hacerlas antes de iniciar la instalación, ya que los originales podrían ser dañados al estar realizando ésta.

¿Como hacer las copias?

R= Existen diferentes comandos del sistema operativo o aplicaciones para realizar copias de archivos, lo recomendable es que se realicen copias completas de disquetes, es decir de un disquete a otro y no por archivos, ya que además de garantizar todo el contenido de los originales es más rápido y sencillo.

En cuanto a las bases de datos también es preferible tener dos copias de éstas, una para dar apoyo y soporte al sistema instalado y la otra como respaldo de seguridad de la primera.

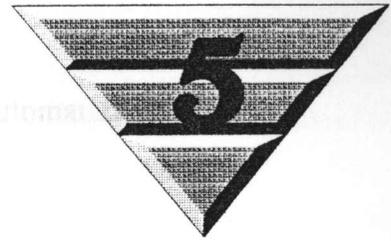
¿Cuándo deben realizarse?

R= Es recomendable realizarlas después de una etapa de actualización, esto es después de que se lleven a cabo ingresos, eliminaciones o modificaciones de las bases de datos como son: maestros, reactivos y/o exámenes.

¿Como realizarlas?

R= Se deben copiar las bases de datos que hayan sido actualizadas, si se actualiza maestros se copia la base de datos de maestros, si es reactivos la de reactivos, etc., deben de realizarse en los dos discos; no importa el comando o la aplicación que se utilice, siempre y cuando se cerciore que se realiza la copia.

DESARROLLO DEL SOFTWARE



El desarrollo del software en términos generales es la creación de las aplicaciones; dichas aplicaciones son los programas, subprogramas, funciones o procedimientos con los que se conforma el sistema, es decir el conjunto de partes que funcionalmente conforman el sistema.

En la etapa de desarrollo se pasó por un proceso de transición del estudio y el análisis con el diseño y del diseño con el desarrollo. Esto se debe a que cada una de éstas etapas producen los detalles con los que el sistema cumplirá con los requerimientos identificados, además de guiar los procedimientos en una forma coordinada y así administrar los recursos de manera que éstos puedan alcanzar las metas y objetivos eficientemente.

Como se comentó en el capítulo de *Estudio y Análisis del Sistema*, el sistema no se encuentra de ninguna manera sujeto a la improvisación, para iniciar la etapa de desarrollo se adecuaron y utilizaron técnicas de análisis y diseño que aumentarían la productividad de la creación del sistema; éstas técnicas se eligieron conforme a los puntos que se estudiaron para realizar dicho sistema.

¿Qué puntos indicaban como realizar el sistema?, para realizar el sistema se necesitaron conocer los siguientes puntos:

- Que realmente se solicitaba.
- La necesidad de que fuera un sistema automatizado.
- El tipo de sistema.
- Determinación de los requerimientos.
- Identificación de los usuarios del sistema.
- Hacia que se encontraba dirigido el sistema.
- La factibilidad del sistema.
- Técnicas de apoyo para el estudio y análisis del sistema.
- Herramientas de apoyo para el diseño.
- Como desarrollar el sistema.
- Estrategias para el desarrollo del sistema.
- Elección del lenguaje de programación.
- Administración del proceso de desarrollo.

De acuerdo a éstos puntos la información se fue obteniendo en cada una de las etapas, como se muestra en los capítulos anteriores, pero ¿como se eligió el método para el desarrollo del sistema?, la elección del método consistió precisamente en los puntos anteriores, para obtener la estrategia de desarrollo requería poder identificar dicha información, la cual fue resultado por la integración de dos métodos *Método del ciclo de vida para el desarrollo de sistemas* y el *Método por análisis estructurado*, ya que éstos dos métodos en forma conjunta cumplían con los puntos para desarrollar el sistema. (véase capítulo 2, Tema: *Estrategias para el desarrollo de sistemas*)

Ya que el ciclo de vida de desarrollo de sistemas incluye actividades de investigación preliminar, determinación de requerimientos, diseño, desarrollo y el análisis estructurado se concentra en especificar lo que se requiere que haga el sistema o la aplicación y sus elementos esenciales del análisis estructurado son símbolos gráficos, diagramas de flujo, etc. cubrían con los puntos detectados para desarrollar el sistema.

Como se mencionó anteriormente no existe ningún método correcto para desarrollar un sistema de información, pero sí existen diferentes formas para producir el sistema correcto para una aplicación. Algunos métodos tienen más éxito que otros y esto depende de cuándo se emplean, cómo se aplican y de los participantes en el proceso de desarrollo. En función al Sistema de Evaluación para Exámenes de Calidad el indicador definitivo son los resultados obtenidos.

Dentro del método de análisis estructurado permitió observar todos los aspecto lógicos y el método del ciclo de vida apoyo a obtener en forma concreta la información necesaria para el sistema, dando la facilidad de interactuar en cada una de las etapas según se necesitaba.

El desarrollo es un proceso formado por las etapas de estudio y análisis y diseño, inicia desde el momento en que surge la necesidad de un sistema, por lo tanto todas las actividades descritas en los capítulos anteriores conforman el desarrollo, dichas actividades están completamente relacionadas y en general son inseparables; quizá sea difícil determinar el orden de los pasos que siguen para efectuarlas al iniciar el proyecto ya que las diversas partes del proyecto pueden encontrarse al mismo tiempo en distintas fases del desarrollo; en algunos componentes en la fase de estudio y análisis mientras que otros en etapas avanzadas del diseño, es por eso que se deben tomar en cuenta puntos o aspectos que determinen la factibilidad de la estrategia para desarrollar en forma completa el sistema.

Para el desarrollo del software se tomaron ciertos aspectos específicos para obtener un diseño cómodo para los usuarios, en el sentido que fuera fácil de utilizar, en términos generales amigable, sin complicaciones para utilizarlo y que de manera que fuera siendo utilizado conocido para los usuarios, los aspectos que se tomaron en cuenta fueron los siguientes:

- Entrada restringida a la áreas del sistema.
- Menús de selección.
- Estructura modular.
- Accesibilidad a cada una de las interacciones.
- Ventanas de captura de datos fáciles para los usuarios.
- Determinación de información.
- Consultas generales al entrar a cada módulo.
- Consultas directas al editar un registro.
- Operaciones lógicas y aritméticas precisas.
- Tiempos de procesamiento factibles.
- Control de errores.
- Generación de reportes confiables.
- Pruebas de seguridad del sistema.

Uno de los aspectos importantes fue el diseñar el sistema en forma modular, ya que esto permitió mayor facilidad para comprender cada parte del sistema y desarrollar mejor el sistema.

Por ejemplo dividir el sistema en módulos permitió desarrollar cada una de las aplicaciones de una forma más eficiente, esto es concentrarse en una aplicación individual pero sin perder de vista el panorama global del sistema, como es en el módulo de maestros:

- Permite controlar la información de los usuarios del sistema solo al administrador.
- Permite Agregar o dar de alta usuarios.
- Tiene la opción de modificar datos de los mismos (editar).
- Ofrece la opción de eliminar usuarios.

- Categoriza a los usuarios del sistema.
- Asigna claves de acceso según la categoría.
- Restringe la entrada conforme a la categoría.
- Permite generar reportes de los usuarios solo al administrador.

Si se desarrollara el software en una forma global cada una de éstas aplicaciones haría un programa mas grande y complejo de entender y así mismo de desarrollar, el control sobre él mismo seria muy incomodo para depurar y agregarlo al sistema, pero de la forma modular se trabajo como diferentes aplicaciones mas fáciles de programar y factibles a la hora de integrarlas para conformar el módulo y éstos a su vez en el sistema.

La elección del lenguaje se debió a la diversidad de herramientas de trabajo para desarrollar el software; para este sistema se utilizó Fox Pro versión 2.6 para MS-DOS, el cuál aporta herramientas para el diseño de pantallas, menús de diferentes tipos, ventanas, selección de opciones diversas, etc. además de ser un lenguaje diseñado para el control de las bases de datos (xBASE), lo que permitió que el desarrollo del software fuera mas sencillo de crear.

Las habilidades para el desarrollo de software son diversas y se adecuan conforme a la experiencia del programador; conforme a los resultados obtenidos al realizar el sistema y a las características de aprendizaje que obtuve al realizarlo me permito sugerir tres puntos que considero importantes con respecto a éste tipo de sistemas:

1. Experiencia técnica.
2. Conocimiento específico de la aplicación.
3. Experiencia con los sistemas de información para la dirección.

La experiencia técnica se refiere a la capacidad de trabajar con hardware y software, aparte de la herramienta de desarrollo (por ejemplo, Fox Pro), se deben conocer otros aspectos como el sistema operativo y el entorno de red.

El conocimiento específico de la aplicación se refiere a la naturaleza de las tareas que se han de automatizar, por ejemplo si se desarrolla un sistema contable se encontrará difícil realizarlo si no se cuenta con un mínimo de formación contable, lo cual puede causar diversos problemas como el que no realice las operaciones en el orden adecuado, tenga un enfoque o terminología adecuado o genere informes en un formato estándar, entre otros.

La experiencia de sistemas de información para la dirección comprende técnicas para el correcto desarrollo de sistemas, incluye la definición de necesidades del usuario, de un adecuado diseño interno, una interfaz de usuario apropiado y procedimientos normalizados de documentación y mantenimiento, entre otros.

El objetivo de estos puntos no perder de vista que el software ha de satisfacer las necesidades de los usuarios, que pueden ser tanto simples empleados como supervisores, gerentes o altos ejecutivos y por lo cual preferible se desarrollo como un proceso de varias etapas hasta conformar en su totalidad al sistema.

Otro de los puntos importantes es la estimación de tiempo; una razón por la cual los proyectos de software no cumplen sus plazos y exceden su presupuesto es la asignación de tiempo a cada etapa. Muchas veces se supone que la codificación consumirá la mayoría del tiempo, ignorando prácticamente las etapas de requisitos, estudio y análisis, diseño, prueba y mantenimiento; lo cual nos lleva a ver en la siguiente figura la estimación de la división de tiempo

Proceso de Desarrollo (percepción inicial)

Etapa	Porcentaje de tiempo
Estudio y análisis	5%
Diseño	10%
Desarrollo	70%
Prueba	10%
Mantenimiento	5%

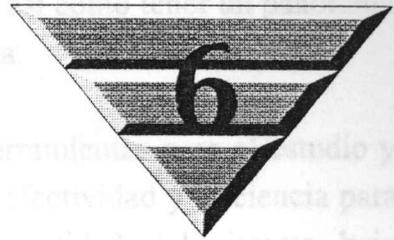
Lo cual surge por desconsiderar las etapas de estudio y análisis y diseño, pero al admitir la importancia del análisis y el diseño, se logra reducir el esfuerzo total y hacerlo más predecible, tomando en cuenta que la mitad del esfuerzo se consume antes de haber escrito una sola línea de código, en la siguiente figura se puede observar como la codificación ocupa casi un tercio del tiempo total, mientras que el estudio y análisis y el diseño juntos casi la mitad del total del proyecto.

Proceso de Desarrollo (aproximación corregida)

Etapa	Porcentaje de tiempo
Estudio y análisis	15%
Diseño	30%
Desarrollo	30%
Prueba	10%
Mantenimiento	15%

El método intuitivo es el enfoque más ampliamente usado, sin embargo es preferible utilizar estimaciones de tiempo basadas por cálculos más precisos como el propuesto por IBM, el método PERT, creando fórmulas estándar que tomen en cuenta las características del programa y el personal, los recorridos estructurados, entre otros más, esto sí se desea cumplir con la programación del desarrollo..

CONCLUSIONES Y RECOMENDACIONES



El obtener la habilidad para examinar la solicitud de un sistema y determinar si el empleo de una computadora es lo apropiado dentro del marco de la organización y para la situación planteada, requirió de saber cómo recolectar e interpretar hechos que ayudaran a diagnosticar acertadamente el problema, así como a entenderlo mas claramente.

Para conocer los procesos y decisiones, es necesario identificar condiciones y acciones y saber qué información está disponible, de esta manera se pueden emprender las acciones conforme a las combinaciones específicas de condiciones. El uso de herramientas para determinar los requerimientos del sistema, como: tablas, árboles y español estructurado o pseudocódigo, etc., apoyan para adquirir información con respecto a las tomas de decisión incluyendo los procedimientos para el procesamiento de transacciones.

La determinación de los requerimientos de un sistema se va dando en forma gradual conforme se vaya teniendo un nivel de comprensión del sistema o de un proceso determinado, para esto es preferible crear una estructura modular, es decir dividir el sistema en varias partes de manera que se conviertan en varios problemas y tener un enfoque mas particular, como se dice comunmente "Divide y vencerás".

En cuanto a la capacidad para diseñar y desarrollar un sistema conforme a las especificaciones solicitadas, debe estar en base a los conocimientos para seleccionar el método para el control de datos, como son: entrada, almacenamiento, acceso, procesamiento, salidas, etc., así como tener un panorama global del desarrollo del software y métodos de prueba.

En conclusión el uso apropiado de métodos y herramientas para el estudio y análisis, diseño y desarrollo de un sistema, mejora la efectividad y eficiencia para generarlo, al mismo tiempo que se beneficia la calidad del sistema bajo desarrollo, logrando una automatización con beneficios en tiempo, evitando tedio asociado con varias tareas, al seguir procedimientos consistentes y al realizar automáticamente la captura de datos sobre el sistema; esto se debe de llevar siempre con la idea en desarrollar un buen sistema, que trabaja bien y que sea eficiente.

La calidad de un sistema depende de su diseño, desarrollo, prueba e implantación. Un aspecto de la calidad del sistema es su confiabilidad, es confiable al usarse de manera razonable y conforme al seguimiento de los procesos e interacciones que se van realizando en cada uno de los módulos que integran el sistema, evitando fallas peligrosas con la información.

El sistema se encuentra bajo detección, corrección y tolerancia a errores utilizando estrategias para mantenerlo en buena operación y previniendo fallas, es por eso que es recomendable que se sigan los pasos que van surgiendo entre operación y operación de cada módulo o aplicación. Además el sistema cuenta con el apoyo de Ayuda en cada uno de los módulos e interacciones, que guían al usuario a efectuar cada procedimiento de manera efectiva y bajo un nivel de buen funcionamiento.

REGIONS EAST OF MEXICO 43 "Normal"
REGIONS WEST OF MEXICO 44 "Normal"
REGIONS SOUTH OF MEXICO 45 "Normal"
REGIONS NORTH OF MEXICO 46 "Normal"

MEXICO 47 "Normal" Mexico 48 "Normal"

REGIONS EAST OF MEXICO 49 "Normal"

REGIONS WEST OF MEXICO 50 "Normal"

APENDICE

REGIONS EAST OF MEXICO FROM 51 TO 52 "Normal"
REGIONS WEST OF MEXICO FROM 53 TO 54 "Normal"

REGIONS SOUTH OF MEXICO FROM 55 TO 56 "Normal"
REGIONS NORTH OF MEXICO FROM 57 TO 58 "Normal"

REGIONS EAST OF MEXICO 59 "Normal"

REGIONS WEST OF MEXICO 60 "Normal"

REGIONS SOUTH OF MEXICO 61 "Normal"

REGIONS NORTH OF MEXICO 62 "Normal"

REGIONS EAST OF MEXICO 63 "Normal"

REGIONS WEST OF MEXICO 64 "Normal"

Código Fuente

REGIONS EAST OF MEXICO 65 "Normal"
REGIONS WEST OF MEXICO 66 "Normal"
REGIONS SOUTH OF MEXICO 67 "Normal"
REGIONS NORTH OF MEXICO 68 "Normal"

REGIONS EAST OF MEXICO 69 "Normal"

REGIONS WEST OF MEXICO 70 "Normal"

REGIONS SOUTH OF MEXICO 71 "Normal"

REGIONS NORTH OF MEXICO 72 "Normal"

REGIONS EAST OF MEXICO 73 "Normal"

REGIONS WEST OF MEXICO 74 "Normal"

REGIONS SOUTH OF MEXICO 75 "Normal"

REGIONS NORTH OF MEXICO 76 "Normal"

REGIONS EAST OF MEXICO 77 "Normal"

REGIONS WEST OF MEXICO 78 "Normal"

```

** Función para prepara ambiente de entrada al sistema

Public nTipo, cKlave, cMaestro, cArchivo, a_Ayuda, c_Ayuda1, c_Ayuda2

```

```

CLEAR
POP KEY ALL
CLOSE DATABASES
SET EXCLUSIVE ON && OFF
SET DATE TO DMY
SET CENTURY ON
SET Exact OFF
SET FIXED OFF
SET DECIMALS TO 4
SET NOTIFY ON
SET CONFIRM OFF
SET SAFETY OFF
SET TALK OFF
SET CLOCK ON
SET DELETED ON
SET MEMOWIDTH TO 65
SET ESCAPE OFF
CLEAR MACROS
ON KEY LABEL Shift+F1 do ayuda with a_Ayuda

```

```
DO parnta
```

```

SELECT 1
USE maestros
SELECT 2
USE reactivo
SELECT 3
USE examenes
SELECT 4
USE seec
SELECT 5
USE Ayuda

```

```
nTipo = 0
```

```
*** Módulo principal SEEC
```

```
DO prepara
```

```

SELECT maestros
SET ORDER TO Tipo
GO TOP
cKlave = password()
nTipo = Tipo
cMaestro = Nombre

```

```

If LASTKEY() < 48 .OR. LASTKEY() > 122
WAIT WINDOW 'Clave no autorizada' NOWAIT
do salir
return

```

```
EndIf
```

```

If !empty(cKlave)
PUSH KEY CLEAR
PUSH MENU _Msysmenu
sigue = .T.
Do While sigue
DO MenuP1

```

```

DO CASE
CASE nTipo = 2
DO desactiva1
CASE nTipo = 3
DO desactiva2
ENDCASE
KEYBOARD "{ALT+S}"
If sigue
ACTIVATE MENU _MSYSMENU
Else
DEACTIVATE MENU _MSYSMENU
RELEASE MENU _MSYSMENU

```

```
EndIf
```

```

EndDo
POP MENU _Msysmenu
POP KEY

```

```

Else
RETURN
EndIf
DO salir

```

```
RETURN
```

```

PROCEDURE Desactiva1
RELEASE BAR 4 OF sistema && "Maestros"
RELEASE BAR 6 OF sistema && "-"
RETURN

```

```

PROCEDURE Desactiva2
* RELEASE BAR 3 OF sistema && "-"

```

```

RELEASE BAR 4 OF sistema && "Maestros"
RELEASE BAR 6 OF sistema && "-"
RELEASE BAR 7 OF sistema && "Ex menes"
RELEASE BAR 8 OF sistema && "Reportes"
RETURN

```

```
#define LOSTIPOS "Administrador Maestro Capturista "
```

```
* Módulo Maestros
```

```

PUSH KEY CLEAR
PUSH MENU _Msysmenu

```

```

c_Ayuda=a_Ayuda
a_Ayuda="Maestros"

```

```

SELECT maestros
SET ORDER TO Orden
GOTO TOP

```

```

DEFINE WINDOW maestros FROM 5,7 TO 18,72 TITLE "MAESTROS"
GROW FLOAT CLOSE MINIMIZE SYSTEM COLOR SCHEME 10
ZOOM SHADOW

```

```

BROWSE WINDOW maestros NOMENU NOAPPEND NODELETE
NOWAIT SAVE:

```

```

FIELDS Tipo = IIF( Between(Tipo, 1, 3), SubStr(LOSTIPOS, (Tipo - 1)
* 15 + 1, 15), Space(15) );
:15:H="Categoría", ;
NOMBRE :R :40 ;
CLAVE :R :5

```

```

RELEASE WINDOW maestros
DO ControlM.mpr
ON SELECTION BAR 3 OF movimiento Do Elimina
BROWSE LAST

```

```

a_Ayuda=c_Ayuda
POP MENU _Msysmenu
POP KEY

```

```
RETURN
```

```

FUNCTION Elimina
Private nOpcion

```

```
PUSH KEY CLEAR
```

```

DEFINE WINDOW elimina_m ;
FROM INT((SROW()-6)/2),INT((SCOL()-40)/2) ;
TO INT((SROW()-6)/2)+5,INT((SCOL()-40)/2)+39
FLOAT ;
NOCLOSE ;
SHADOW ;
NOMINIMIZE ;
DOUBLE ;
COLOR SCHEME 7

```

```

IF WWISIBLE("elimina_m")
ACTIVATE WINDOW elimina_m SAME
ELSE
ACTIVATE WINDOW elimina_m NOSHOW
ENDIF

```

```

c_Ayuda = a_Ayuda
a_Ayuda="Maestros Elimina"
ON KEY LABEL Shift+F1 do ayuda

```

```

@ 3,3 GET nOpcion ;
PICTURE "@*HN <Si;<No";
SIZE 1,10,12;
DEFAULT 1 ;
VALID CONFIRMA()
@ 1,9 SAY "% ELIMINAR USUARIO ?";
SIZE 1,20,0
@ 0,14 SAY "Precaución";
SIZE 1,10,0;
COLOR GR+/GR*

```

```

IF NOT WWISIBLE("elimina_m")
ACTIVATE WINDOW elimina_m
ENDIF

```

```

READ CYCLE MODAL ;
DEACTIVATE .t

```

```
a_Ayuda=c_Ayuda
```



```

if nTipo > 0
  POP MENU _MSYSMENU
  POP KEY
  RETURN
endif

a_Ayuda=c_Ayuda
POP MENU _MSYSMENU
POP KEY

RETURN

#define SEMESTRES2 "Septimo Octavo Noveno "
*
* +-----+
* | FUNCTION Exámenes |
* +-----+
*

PUSH KEY CLEAR

PUSH MENU _Msysmenu

SELECT examenes
SET ORDER TO examenes
GOTO TOP

DEFINE WINDOW Examen FROM 5,2 TO 18,77 TITLE "EXAMENES
INTERACTIVOS";
GROW FLOAT CLOSE MINIMIZE SYSTEM COLOR
SCHEME 10 ZOOM SHADOW

BROWSE WINDOW Examen NOMENU NOAPPEND NODELETE
NOWAIT SAVE;
  FIELDS Maestro :R:28 :H:"Aplicó" ; ;
  Carrera = IIF(Carrera = 1, "Administración", "Contaduría" );
  :R:14 ; ;
  Semestre = IIF(
Between(Semestre, 1, 3), SubStr(SEMESTRES2, (Semestre - 1) * 8 + 1,
8), Space(8) );
  :R:8 ; ;
  NumExam :R:8 :H:"Exámenes" ; ;
  Fecha :R:10 :H:"Fecha"

*
*
pregunta1 :R:6
:H="Preg.1" ; ;
pregunta2 :R:6
:H="Preg.2" ; ;
pregunta3 :R:6 :H:"Preg.3" ; ;
pregunta4 :R:6 :H:"Preg.4" ; ;
pregunta5 :R:6 :H:"Preg.5" ; ;
pregunta6 :R:6 :H:"Preg.6" ; ;
pregunta7 :R:6 :H:"Preg.7" ; ;
pregunta8 :R:6 :H:"Preg.8"

RELEASE WINDOW Examen
DO Controll.mpr
BROWSE LAST

if nTipo > 0
  POP MENU _MSYSMENU
  POP KEY
  RETURN
endif

POP MENU _MSYSMENU
POP KEY

RETURN

#define cTipo 'TIP'+transform(n,@L 9999)
n=1

*Parameters cAlumno, cMatric
*PRIVATE cAlumno

SELECT 6
GOTO TOP

For x=6 to 24
  select(x)
  if empty(alias())
    x=x-1
  select(x)

```

```

exit
Endif
EndFor

cont=0
nresp=exanum
numexa=exanum

For i=1 TO RECCOUNT()
  SKIP
  if numexa=exanum
    cont=cont+1
  else
    nresp=exanum
    numexa=nresp
  endif
Endif

EndFor

cont=RECCOUNT()-cont

WAIT WINDOW 'Asegurese que la impresora se encuentre lista'

for n=1 to cont
  select(x)
  COPY STRUCTURE TO cTipo
  SELECT *;
  FROM alias();
  WHERE EXANUM = n;
  INTO TABLE cTipo
  if SYS(13) = 'OFFLINE'
    Do while SYS(13) = 'OFFLINE'
      WAIT WINDOW 'La impresora no
esta lista'
    EndDo
    WAIT WINDOW '<ENTER> para continuar'
  Endif
  IF SYS(13) = 'READY'
    REPORT FORM reportee PREVIEW
  Else
    WAIT WINDOW 'Error de Impresora, impresion
cancelada'
  Endif
  USE IN cTipo
Endfor

*A=0
*DO WHILE A=0
* A=INT(RAND0)*(cont+1)
*ENDDO
*? A
*? cont

*SELECT 6
*COPY STRUCTURE TO cTipo

*SELECT *;
* FROM alias();
* WHERE EXANUM = A;
* INTO TABLE cTipo

*REPLACE ALL Alumno With cAlumno
*REPLACE ALL Matric With cMatric

*SELECT cTipo
*INDEX ON CLAVE TAG CLAVE ADDITIVE
*SET ORDER TO
*SET RELATION TO CLAVE INTO REACTIVO ADDITIVE
*SELECT cTipo
*GOTO TOP

*Do exainter.spr
*
*SET RELATION TO CLAVE INTO REACTIVO ADDITIVE

PARAMETERS nCarreras, nAreas6, nAreas4, nAreas1, nAreas7, nAreas8;;
nSemestres
PRIVATE nTemp, NumPreg6, NumPreg4, NumPreg1, NumPreg7,
NumPreg8

* Asignación de carrera, áreas, número de preguntas por área, semestres y
* número de exámenes

nCarreras=m.carrera

nAreas6=6
nAreas4=4

```

```

nAreas1=1
nAreas7=7
nAreas8=8

Numpreg8=m.pregunta8
Numpreg4=m.pregunta4
Numpreg1=m.pregunta1
Numpreg7=m.pregunta7
Numpreg8=m.pregunta8

DO CASE
CASE m.semestre=1
    nSemestres=7
CASE m.semestre=2
    nSemestres=8
OTHERWISE
    nSemestres=9
ENDCASE

examenes=m.numexam

IF Inew
CREATE CURSOR Auxiliar (ExaNum N(4), Clave C(5),
Solucion C(5), Tiempo N(2), Peso N(2), Alumno C(68), Matric C(9), Resp
C(1))
ENDIF

IF m.pregunta6 > 0
nAreas=nAreas6
Numpreg=Numpreg6
DO crea
ENDIF

IF m.pregunta4 > 0
nAreas=nAreas4
Numpreg=Numpreg4
DO crea
ENDIF

IF m.pregunta1 > 0
nAreas=nAreas1
Numpreg=Numpreg1
DO crea
ENDIF

IF m.pregunta7 > 0
nAreas=nAreas7
Numpreg=Numpreg7
DO crea
ENDIF

IF m.pregunta8 > 0
nAreas=nAreas8
Numpreg=Numpreg8
DO crea
ENDIF

RETURN

PROCEDURE crea
FOR k = 1 TO examenes
WAIT "Un momento, Seleccionando reactivos :
"+LTRIM(STR(k)) WINDOW NOWAIT
CREATE CURSOR Mientras (Clave C(5), Solucion C(5),
Tiempo N(2), Peso N(2))
SELECT Reactivo
SCAN FOR (EMPTY(nCarreras) Or Carreras = nCarreras)
And ;
                (EMPTY(nAreas) Or Areas =
nAreas) And ;
                (EMPTY(nSemestres) Or
Semestres <= nSemestres)
INSERT INTO Mientras VALUES
(STR(RecNo("Reactivo"),5), OrdenaRes(), Reactivo.Tiempo,
Reactivo.Peso)
ENDSCAN
SELECT Mientras
j = Numpreg
FOR i=RECCOUNT() TO 1 STEP -1
GOTO TOP
SKIP CEILING(RAND() * i) - 1
DELETE
j = j - 1
IF j <= 0
EXIT
ENDIF
ENDFOR
BLANK ALL FOR IDELETED()
RECALL ALL
SELECT Auxiliar
GOTO BOTTOM
nTemp = IIF(EOF(), 1, k)

```

```

SELECT Mientras
SCAN FOR !ISBLANK(Solucion)
INSERT INTO Auxiliar VALUES (nTemp,
Mientras.Clave, Mientras.Solucion, Mientras.Tiempo, Mientras.Peso, " ", "
", "")
ENDSCAN
ENDFOR
select auxiliar
INDEX ON EXANUM TAG AUXILIAR ADDITIVE

RETURN

FUNCTION OrdenaRes
PRIVATE cOrdena, i, nTemp, cCar
cOrdena = "1234"
FOR i=1 to 4
nTemp = CEILING(RAND() * i)
cOrdena = STUFF(cOrdena, nTemp, 1, "") +
SUBSTR(cOrdena, nTemp, 1)
ENDFOR
RETURN cOrdena

SET DELETED OFF
#define cFile 'EXA'+transform(n,'@L 9999')+'.DBF'
n=1

IF Inew
cAuxFile= '
FOR n=1 to 9999
IF FILE (cFile)
cAuxFile=cFile
ELSE
cAuxFile=cFile
n=VAL(SUBSTR(cAuxFile, 4, 4))
n=n+1
IF not FILE (cFile)
n=n-1
exit
ENDIF
NEXT n
ELSE
FOR n=1 to 9999
IF FILE (cFile)
cAuxFile=cFile
ELSE
cAuxFile=cFile
n=VAL(SUBSTR(cAuxFile, 4, 4))
n=n-1
exit
ENDIF
NEXT n
ENDIF

SELECT 2
SET ORDER TO reactivos

*STORE 0 TO mcount,Carrera,Area,Semestre,copias,;
*
* Tiempo,Peso,TiemTot,PesoTot,y,ReacTot,;
* x,z,reg

*STORE 0 TO
mcount,copias,Tiempo,Peso,TiemTot,PesoTot,y,ReacTot,x,z,reg

DO WHILE reg <= Copias

carrera=m.carrera

DO CASE
CASE m.semestre=1
semestre=7
CASE m.semestre=2
smestre=8
OTHERWISE
semestre=9
ENDCASE

IF m.pregunta6 <> 0
area=6
ENDIF

copias=m.pregunta6

* @ 3, 13 SAY 'Carrera : ' GET Carrera PICTURE '9'
* @ 4, 13 SAY 'Area : ' GET Area PICTURE '9'
* @ 5, 13 SAY 'Semestre : ' GET Semestre PICTURE '9'
* @ 6, 13 SAY 'Copias : ' GET Copias PICTURE '99'
* READ

```

```

Copias2=Copias

DIMENSION NoReg(Copias)
For a=1 to Copias
    NoReg(a)=0
    NoReg(copias)=0
Next a

SELECT CARRERAS, AREAS, SEMESTRES, NOMBRE, TIEMPO, PESO,;
    OP1, OP2, OP3, OP4, TEXTO;
FROM REACTIVO;
WHERE CARRERAS = Carrera;
    AND AREAS = Area;
    AND SEMESTRES <= Semestre;
INTO TABLE find.dbf

if RECCOUNT() < copias
    use in find
    DELETE FILE find.dbf
endif

reg=RECCOUNT()+1

ENDDO

DO WHILE Copias > 0
    z=INT(RAND()*reg)
    GO TOP
    SKIP z
    NoReg(Copias)=RECNO()
    For a=1 to copias2
        if NoReg(Copias) = NoReg(a)
            z=INT(RAND()*reg)
            GO TOP
            SKIP z
            NoReg(Copias)=RECNO()
        endif
    Next
    DELETE
    TiemTot=TiemTot+Tiempo
    PesoTot=PesoTot+Peso
    ? 'Tiempo:',Tiempo,'Peso:',Peso
    reg = reg - 1
    Copias = Copias - 1
ENDDO

IF FILE (cFile)
    SELECT 6
    USE cFile ALIAS exams
    SELECT find
    GO TOP
    SCAN FOR DELETED()
        SCATTER MEMVAR
        INSERT INTO exams FROM MEMVAR
    ENDSCAN
ELSE
    SELECT CARRERAS, AREAS, SEMESTRES, NOMBRE,
TIEMPO, PESO,;
    OP1, OP2, OP3, OP4, TEXTO;
    FROM FIND;
    WHERE DELETED();
    INTO TABLE cFile
ENDIF
RECALL ALL

*? 'Tiempo total:',TiemTot,'Peso Total:',PesoTot

USE IN find
DELETE FILE find.dbf
SET DELETED ON

RETURN cFile

Parameters Inew
#define cFile 'EXA'+transform(n,'@L 9999')+'.DBF'
n=1

IF Inew
    cAuxFile='
    FOR n=1 to 9999
        IF FILE (cFile)
            cAuxFile=cFile
        ELSE

```

```

        cAuxFile=cFile
        n=VAL(SUBSTR(cAuxFile, 4, 4))
        n=n+1
        IF not FILE (cFile)
            n=n-1
            exit
        ENDIF
    NEXT n
ENDIF

select auxiliar
copy structure to cFile
copy to cFile
USE cFile
SELECT cFile
INDEX ON CLAVE TAG CLAVE ADDITIVE
SET ORDER TO

SELECT reactivo
SET ORDER TO TAG clave
SELECT cFile
SET RELATION TO clave INTO reactivo

cArchivo=cFile
RETURN cArchivo

USE MAESTROS
SET ORDER TO CLAVES
GOTO TOP
LOCATE FOR CLAVE='KCO6'
BROWSE

PARAMETERS referencia

PUSH KEY CLEAR
PUSH MENU _Msysmenu

IF EMPTY(WONTOP())
    a_Ayuda = "Acerca de la Ayuda"
endif
SELECT 5
SET ORDER TO referencia
LOCATE FOR referencia=a_Ayuda
DO ayuda.spr

POP MENU _Msysmenu
POP KEY

RETURN

PARAMETERS Inew

#REGION 1

#REGION 0
REGIONAL m.currarea, m.talkstat, m.compstat

IF SET("TALK") = "ON"
    SET TALK OFF
    m.talkstat = "ON"
ELSE
    m.talkstat = "OFF"
ENDIF
m.compstat = SET("COMPATIBLE")
SET COMPATIBLE FOXPLUS

m.currarea = SELECT()

IF NOT WEXIST("captura_m");
    OR UPPER(WTITLE("CAPTURA_M")) == "CAPTURA_M.PJX"
;
    OR UPPER(WTITLE("CAPTURA_M")) ==
"CAPTURA_M.SCX";
    OR UPPER(WTITLE("CAPTURA_M")) ==
"CAPTURA_M.MNX";
    OR UPPER(WTITLE("CAPTURA_M")) ==
"CAPTURA_M.PRG";
    OR UPPER(WTITLE("CAPTURA_M")) == "CAPTURA_M.FRX"
;
    OR UPPER(WTITLE("CAPTURA_M")) ==
"CAPTURA_M.QPR"
    DEFINE WINDOW captura_m ;
        FROM INT((SROW()-8)/2),INT((SCOL()-57)/2) ;

```

```

        TO INT((SROW()-8)/2)+7,INT((SCOL()-57)/2)+56
        TITLE " Alta de Usuario ";
        FLOAT;
        NOCLOSE;
        SHADOW;
        NOMINIMIZE;
        DOUBLE;
        COLOR SCHEME 5
    ENDIF

```

```

#REGION 1
Private nOpcion, nombre, cTipo

```

```

If Inew
    SCATTER MEMVAR BLANK
    STORE 1 TO Tipo, nCatego
    m.clave = genpass(nCatego)
Else
    SCATTER MEMVAR
Endif
cTipo = TipoNom(m.Tipo)

```

```

#REGION 1
IF WWISIBLE("captura_m")
    ACTIVATE WINDOW captura_m SAME
ELSE
    ACTIVATE WINDOW captura_m NOSHOW
ENDIF

```

```

@ 3,3 SAY "Nombre .:" ;
    SIZE 1,8, 0
@ 1,36 SAY "Clave .:" ;
    SIZE 1,8, 0
@ 1,12 GET Dummy2 ;
    PICTURE "@*IVN " ;
    SIZE 1,16,1 ;
    DEFAULT 0 ;
    WHEN Inew ;
    VALID _rey1199d3()
@ 1,12 SAY cTipo ;
    SIZE 1,15
@ 1,46 SAY m.clave ;
    SIZE 1,4 ;
    PICTURE "@!"
@ 3,12 GET m.nombre ;
    SIZE 1,40 ;
    DEFAULT "" ;
    PICTURE "@!"
@ 5,6 GET nOpcion ;
    PICTURE "@*HN <Guardar,<Cancelar" ;
    SIZE 1,14,15 ;
    DEFAULT 1 ;
    VALID _rey1199zw()
@ 1,27 SAY "" ;
    SIZE 1,1, 0
@ 0,34 TO 2,51
@ 1,3 SAY "Usuario .:" ;
    SIZE 1,9, 0

```

```

IF NOT WWISIBLE("captura_m")
    ACTIVATE WINDOW captura_m
ENDIF

```

```

READ CYCLE MODAL ;
    WHEN _rey1199ajx() ;
    DEACTIVATE .t ;
    SHOW _rey1199akd()

```

```

RELEASE WINDOW captura_m
SELECT (m.currarea)

```

```

#REGION 0
IF m.talkstat = "ON"
    SET TALK ON
ENDIF
IF m.compstat = "ON"
    SET COMPATIBLE ON
ENDIF

```

```

* +-----+
FUNCTION _rey1199d3 && Dummy2 VALID
#REGION 1

```

```

    m.Tipo = Tipo(m.Tipo)
    cTipo = TipoNom(m.Tipo)

```

```

    If cTipo="Administrador"

```

```

        nCatego = 1
    Endif
    If cTipo="Maestro "
        nCatego = 2
    Endif
    If cTipo="Capturista "
        nCatego = 3
    Endif

```

```

    m.clave = genpass(nCatego)

```

```

    SHOW GETS OFF

```

```

*
FUNCTION _rey1199zw && nOpcion VALID
#REGION 1
If nOpcion = 1
    SELECT maestros
    SET ORDER TO orden
    GO TOP
    If IEmpty(m.Clave) And Seek(m.clave)
        WAIT "Seleccione Usuario Nuevamente" WINDOW NOWAIT
        _CUROBJ = ObjNum(m.nombre)
        RETURN .T.
    Endif
    If Empty(m.nombre)
        WAIT "Falta NOMBRE" WINDOW NOWAIT
        _CUROBJ = ObjNum(m.nombre)
        RETURN .T.
    Endif
    If Inew
        APPEND BLANK
    Endif
    GATHER MEMVAR
Else
    GOTO TOP
Endif

```

```

a_Ayuda = c_Ayuda
CLEAR READ
RETURN

```

```

*
FUNCTION _rey1199ajx && Read Level When
* When Code from screen: CAPTURAM

```

```

#REGION 1
c_Ayuda = a_Ayuda
a_Ayuda="Maestros Captura"
ON KEY LABEL Shift+F1 do ayuda

```

```

*
FUNCTION _rey1199akd && Read Level Show
PRIVATE currwind
STORE WOUTPUT() TO currwind
*
* Show Code from screen: CAPTURAM

```

```

#REGION 1
IF SYS(2016) = "CAPTURA_M" OR SYS(2016) = ""
    ACTIVATE WINDOW captura_m SAME
    @ 1,12 SAY cTipo ;
        SIZE 1,15, 0
    @ 1,46 SAY m.clave ;
        SIZE 1,4, 0 ;
        PICTURE "@!"
ENDIF

```

```

IF NOT EMPTY(currwind)
    ACTIVATE WINDOW (currwind) SAME
ENDIF

```

```

PARAMETERS Inew, cont

```

```

*
#REGION 1
#REGION 0
REGIONAL m.currarea, m.talkstat, m.compstat

```

```

IF SET("TALK") = "ON"
    SET TALK OFF
    m.talkstat = "ON"
ELSE
    m.talkstat = "OFF"
ENDIF

```

```

m.compstat = SET("COMPATIBLE")
SET COMPATIBLE FOXPLUS

```

```

m.currarea = SELECT()

IF NOT WEXIST("captura_r") ;
  OR UPPER(WTITLE("CAPTURA_R")) == "CAPTURA_R.PJX"
  ;
  OR UPPER(WTITLE("CAPTURA_R")) == "CAPTURA_R.SCX"
  ;
  OR UPPER(WTITLE("CAPTURA_R")) == "CAPTURA_R.MNX"
  ;
  OR UPPER(WTITLE("CAPTURA_R")) == "CAPTURA_R.PRG"
  ;
  OR UPPER(WTITLE("CAPTURA_R")) == "CAPTURA_R.FRX"
  ;
  OR UPPER(WTITLE("CAPTURA_R")) == "CAPTURA_R.QPR"
  ;
  DEFINE WINDOW captura_r ;
    FROM INT((SROW()-23)/2),INT((SCOL()-76)/2) ;
    TO INT((SROW()-23)/2)+22,INT((SCOL()-
76)/2)+75 ;
    TITLE " Control de Reactivos " ;
    FLOAT ;
    NOCLOSE ;
    SHADOW ;
    NOMINIMIZE ;
    DOUBLE ;
    COLOR SCHEME 5
  ;
ENDIF
*

#REGION 1
Private nOpcion, Carreras, Areas, Nombre, Texto, nOpcion2, cCarrera,
cArea, Clave
If INew
  SCATTER MEMVAR BLANK MEMO
  STORE 1 TO Carreras, Areas, Semestres, nOpcion2
  GOTO BOTTOM
  Clave = STR(RECCOUNT()+1,5)
Else
  SCATTER MEMVAR MEMO
EndIf
cCarrera = CarrNom(m.Carreras)
cArea = AreaNom(m.Areas)
*

#REGION 1
IF WVISIBLE("captura_r")
  ACTIVATE WINDOW captura_r SAME
ELSE
  ACTIVATE WINDOW captura_r NOSHOW
ENDIF
@ 14,2 TO 14,71
@ 0,1 TO 19,72
@ 14,2 SAY " Opciones " ;
  SIZE 1,10, 0
@ 1,3 EDIT m.Texto ;
  SIZE 13,88,999 ;
  DEFAULT "" ;
  SCROLL
@ 15,6 GET m.op1 ;
  SIZE 1,29 ;
  DEFAULT ""
@ 16,6 GET m.op2 ;
  SIZE 1,29 ;
  DEFAULT ""
@ 17,6 GET m.op3 ;
  SIZE 1,29 ;
  DEFAULT ""
@ 18,6 GET m.op4 ;
  SIZE 1,29 ;
  DEFAULT ""
@ 15,3 SAY "a." ;
  SIZE 1,2, 0
@ 16,3 SAY "b." ;
  SIZE 1,2, 0
@ 17,3 SAY "c." ;
  SIZE 1,2, 0
@ 18,3 SAY "d." ;
  SIZE 1,2, 0
@ 16,38 SAY "Tiempo" ;
  SIZE 1,6, 0
@ 15,38 SAY "Nombre" ;
  SIZE 1,6, 0
@ 18,38 SAY "Area" ;
  SIZE 1,4, 0
@ 17,38 SAY "Carrera" ;
  SIZE 1,7, 0
@ 15,46 GET m.nombre ;
  SIZE 1,20 ;
  DEFAULT "" ;

```

```

  PICTURE "@!"
@ 16,46 GET m.tiempo ;
  SIZE 1,2 ;
  DEFAULT 0
@ 16,56 GET m.peso ;
  SIZE 1,2 ;
  DEFAULT 0
@ 16,51 SAY "Peso" ;
  SIZE 1,4, 0
@ 16,61 SAY "Semestre" ;
  SIZE 1,8, 0
@ 16,70 GET m.Semestres ;
  SIZE 1,1 ;
  DEFAULT 0 ;
  PICTURE "@KZ 9"
@ 17,46 GET Dummy1 ;
  PICTURE "@!VN " ;
  SIZE 1,25,1 ;
  DEFAULT 0 ;
  VALID _rfy0vj6md()
@ 17,46 SAY cCarrera ;
  SIZE 1,24
@ 17,70 SAY "" ;
  SIZE 1,1, 0
@ 18,46 GET Dummy2 ;
  PICTURE "@!VN " ;
  SIZE 1,25,1 ;
  DEFAULT 0 ;
  VALID _rfy0vj7di()
@ 18,46 SAY cArea ;
  SIZE 1,24
@ 20,17 GET nOpcion ;
  PICTURE "@!HN \\<Guardar,17<Cancelar" ;
  SIZE 1,16,7 ;
  DEFAULT 1 ;
  VALID _rfy0vj7yn()
@ 19,36 SAY "-." ;
  SIZE 1,1, 0
@ 15,36 TO 18,36
@ 14,36 SAY "-." ;
  SIZE 1,1, 0
@ 18,70 SAY "" ;
  SIZE 1,1, 0
@ 14,1 SAY "+." ;
  SIZE 1,1, 0
@ 14,72 SAY "" ;
  SIZE 1,1, 0

IF NOT WVISIBLE("captura_r")
  ACTIVATE WINDOW captura_r
ENDIF

READ CYCLE ;
  WHEN _rfy0vj9al() ;
  DEACTIVATE t ;
  SHOW _rfy0vj9cg()

RELEASE WINDOW captura_r
SELECT (m.currarea)

#REGION 0
IF m.talkstat = "ON"
  SET TALK ON
ENDIF
IF m.compstat = "ON"
  SET COMPATIBLE ON
ENDIF

#REGION 1

#REGION 1
PROCEDURE Aviso
  If INew
    @ 0, 2 SAY " Nuevo " COLOR ( Scheme(5, 7) )
  EndIf
  RETURN
*
FUNCTION _rfy0vj6md && Dummy1 VALID
#REGION 1
  m.Carreras = Carrera(m.Carreras)
  cCarrera = CarrNom(m.Carreras)
  SHOW GETS OFF
FUNCTION _rfy0vj7di && Dummy2 VALID
#REGION 1
  m.Areas = Area(m.Areas)
  cArea = AreaNom(m.Areas)

```

```

SHOW GETS OFFI
*
FUNCTION _rfy0vj7yn  && nOpcion VALID
#REGION 1
DO CASE
  Case nOpcion != 1
    CLEAR READ

  CASE Empty(m.Texto)
    WAIT "Debe proporcionar el Reactivo" WINDOW NOWAIT
      _CUIROBJ = ObjNum(m.Texto)

  Case Empty(m.Op1)
    WAIT "Deben ser CUATRO Opciones" WINDOW NOWAIT
      _CUIROBJ = ObjNum(m.op1)

  Case Empty(m.Op2)
    WAIT "Deben ser CUATRO Opciones" WINDOW NOWAIT
      _CUIROBJ = ObjNum(m.op2)

  Case Empty(m.Op3)
    WAIT "Deben ser CUATRO Opciones" WINDOW NOWAIT
      _CUIROBJ = ObjNum(m.op3)

  Case Empty(m.Op4)
    WAIT "Deben ser CUATRO Opciones" WINDOW NOWAIT
      _CUIROBJ = ObjNum(m.op4)

  Case Empty(m.Nombre)
    WAIT "Digite el Nombre del Reactivo" WINDOW NOWAIT
      _CUIROBJ = ObjNum(m.Nombre)

  Case m.tiempo <= 0
    WAIT "Debe ser mayor que cero" WINDOW NOWAIT
      _CUIROBJ = ObjNum(m.tiempo)

  Case m.peso <= 0
    WAIT "Debe ser mayor que cero" WINDOW NOWAIT
      _CUIROBJ = ObjNum(m.peso)

  Case IBetween(m.Semestres, 1, 9)
    WAIT "Digite un semestre entre 1 y 9" WINDOW NOWAIT
      _CUIROBJ = ObjNum(m.Semestres)

  Otherwise
    Carreras = IIF(Between(m.Carreras, 1, 3), m.Carreras, 1)
    Areas = IIF(Between(m.Areas, 1, 8), m.Areas, 1)

    If INew
      APPEND BLANK
    Endif

    GATHER MEMVAR MEMO
    CLEAR READ

  ENDCASE
a_Ayuda = c_Ayuda
RETURN
*
FUNCTION _rfy0vj9ai  && Read Level When
*
* When Code from screen: CAPTURAR
*
#REGION 1
DO Aviso
If INew
  c_Ayuda = a_Ayuda
  a_Ayuda="Reactivos Captura"
  ON KEY LABEL Shift+F1 do ayuda

Else
  c_Ayuda = a_Ayuda
  a_Ayuda="Reactivos Edita"
  ON KEY LABEL Shift+F1 do ayuda

Endif
*
+-----+
FUNCTION _rfy0vj9cg  && Read Level Show
PRIVATE curwind
STORE WOUTPUT() TO curwind
*
* Show Code from screen: CAPTURAR
*
#REGION 1
IF SYS(2016) = "CAPTURA_R" OR SYS(2016) = ""
  ACTIVATE WINDOW captura_r SAME
  @ 17,46 SAY cCarrera ;
    SIZE 1,24, 0
  @ 18,46 SAY cArea ;
    SIZE 1,24, 0
ENDIF

```

```

IF NOT EMPTY(curwind)
  ACTIVATE WINDOW (curwind) SAME
ENDIF

#REGION 1
Private nOpcion, Referencia, Texto

#REGION 0
REGIONAL m.currarea, m.talkstat, m.compstat

IF SET("TALK") = "ON"
  SET TALK OFF
  m.talkstat = "ON"
ELSE
  m.talkstat = "OFF"
ENDIF
m.compstat = SET("COMPATIBLE")
SET COMPATIBLE FOXPLUS

m.currarea = SELECT()

IF NOT WEXIST("ayuda") ;
  OR UPPER(WTITLE("AYUDA")) = "AYUDA.PJX" ;
  OR UPPER(WTITLE("AYUDA")) = "AYUDA.SCX" ;
  OR UPPER(WTITLE("AYUDA")) = "AYUDA.MNX" ;
  OR UPPER(WTITLE("AYUDA")) = "AYUDA.PRG" ;
  OR UPPER(WTITLE("AYUDA")) = "AYUDA.FRX" ;
  OR UPPER(WTITLE("AYUDA")) = "AYUDA.QPR"
  DEFINE WINDOW ayuda ;
    FROM INT((SROW()-15)/2),INT((SCOL()-73)/2) ;
    TO INT((SROW()-15)/2)+14,INT((SCOL()-
73)/2)+72 ;
    TITLE " AYUDA " ;
    FLOAT ;
    CLOSE ;
    SHADOW ;
    MINIMIZE ;
    SYSTEM ;
    COLOR SCHEME 8
ENDIF

#REGION 1
IF WVISIBLE("ayuda")
  ACTIVATE WINDOW ayuda SAME
ELSE
  ACTIVATE WINDOW ayuda NOSHOW
ENDIF
@ 0,2 EDIT texto ;
  SIZE 13,51,999 ;
  COLOR W+/BG,...,W+/B,...,W+/RB ;
  NOMODIFY ;
  DEFAULT "" ;
  SCROLL

@ 2,54 GET nOpcion ;
  PICTURE "@*VN |<Siguiente,<Anterior;Sat<lir" ;
  SIZE 1,16,3 ;
  DEFAULT 1 ;
  VALID _rfy0utx73) ;
  COLOR SCHEME 1

IF NOT WVISIBLE("ayuda")
  ACTIVATE WINDOW ayuda
ENDIF

READ CYCLE MODAL

RELEASE WINDOW ayuda
SELECT (m.currarea)

#REGION 0
IF m.talkstat = "ON"
  SET TALK ON
ENDIF
IF m.compstat = "ON"
  SET COMPATIBLE ON
ENDIF

*
FUNCTION _rfy0utx73  && nOpcion VALID
#REGION 1
DO CASE

```

```

CASE nOpcion=1
    DO siguiente
CASE nOpcion=2
    DO anterior
OTHERWISE
    CLEAR READ
    RETURN
ENDCASE
RETURN

```

```

PROCEDURE siguiente
SKIP
IF EOF()
    GO TOP
ENDIF
SHOW GET texto
RETURN

```

```

PROCEDURE anterior
SKIP -1
IF BOF()
    GO BOTTOM
ENDIF
SHOW GET texto
RETURN

```

PARAMETERS Inew, Itipo

#REGION 1

```

#REGION 0
REGIONAL m.currarea, m.talkstat, m.compstat

```

```

IF SET("TALK") = "ON"
    SET TALK OFF
    m.talkstat = "ON"
ELSE
    m.talkstat = "OFF"
ENDIF
m.compstat = SET("COMPATIBLE")
SET COMPATIBLE FOXPLUS
m.currarea = SELECT()

```

```

IF NOT WEXIST("creaexamen");
    OR UPPER(WTITLE("CREAEXAMEN")) ==
"CREAEXAMEN.PJX";
    OR UPPER(WTITLE("CREAEXAMEN")) ==
"CREAEXAMEN.SCX";
    OR UPPER(WTITLE("CREAEXAMEN")) ==
"CREAEXAMEN.MNX";
    OR UPPER(WTITLE("CREAEXAMEN")) ==
"CREAEXAMEN.PRG";
    OR UPPER(WTITLE("CREAEXAMEN")) ==
"CREAEXAMEN.FRX";
    OR UPPER(WTITLE("CREAEXAMEN")) ==
"CREAEXAMEN.QPR"
    DEFINE WINDOW creaexamen ;
        FROM 3, 12 ;
        TO 16,56 ;
        TITLE " GENERADOR DE EXAMENES " ;
        FLOAT ;
        NOCLOSE ;
        SHADOW ;
        NOMINIMIZE ;
        DOUBLE ;
        COLOR SCHEME 5
ENDIF

```

ENDIF

#REGION 1

```

Private nOpcion, Carrera, Semestre, Pregunta1, Pregunta2, Pregunta3,
Pregunta4, Pregunta5, Pregunta6, Pregunta7, Pregunta8, Fecha,
ClaveProf, Maestro

```

```

If !New
SCATTER MEMVAR BLANK
STORE 1 TO Carrera, Semestre, nOpcion
store ' ' to Maestro
claveprof=cKlave
Maestro=cMaestro
STORE 0 TO Pregunta1, Pregunta2, Pregunta3, Pregunta4,
Pregunta5, Pregunta6, Pregunta7, Pregunta8, nOpcion2
Else
SCATTER MEMVAR
Endf

```

```

#REGION 1
IF WVISIBLE("creaexamen")
    ACTIVATE WINDOW creaexamen SAME
ELSE
    ACTIVATE WINDOW creaexamen NOSHOW
ENDIF

```

```

ENDIF
@ 0,1 TO 10,41
@ 6,4 SAY "Carrera";
    SIZE 1,7, 0
@ 6,24 SAY "Semestre";
    SIZE 1,8, 0
@ 1,4 SAY "Maestro";
    SIZE 1,7, 0
@ 2,4 SAY m.maestro ;
    SIZE 1,35 ;
    PICTURE "@!"
@ 4,27 GET m.numexam ;
    SIZE 1,2 ;
    DEFAULT 0
@ 7,3 GET m.carrera ;
    PICTURE "@^ \<Administración;\<Contaduría";
    SIZE 3,18 ;
    DEFAULT "Administración";
    COLOR SCHEME 5, 6
@ 7,23 GET m.semestre ;
    PICTURE "@^ \<Septimo;\<Octavo;\<Noveno";
    SIZE 3,16 ;
    DEFAULT "Septimo";
    COLOR SCHEME 5, 6
@ 11,4 GET nOpcion2 ;
    PICTURE "@*HN \!\<Guión;\!<Cancelar";
    SIZE 1,14,7 ;
    DEFAULT 1 ;
    VALID _rfj15zwnh()
@ 3,25 TO 5,30
@ 4,4 SAY "No.Tipos de Exámenes";
    SIZE 1,20, 0

```

```

IF NOT WVISIBLE("creaexamen")
    ACTIVATE WINDOW creaexamen
ENDIF

```

```

READ CYCLE MODAL ;
    WHEN _rfj15zwh6() ;
    DEACTIVATE t.

```

```

RELEASE WINDOW creaexamen
SELECT (m.currarea)

```

#REGION 0

```

IF m.talkstat = "ON"
    SET TALK ON
ENDIF
IF m.compstat = "ON"
    SET COMPATIBLE ON
ENDIF

```

```

FUNCTION _rfj15zwnh && nOpcion2 VALID
#REGION 1
If nOpcion2 = 1

```

```

If !Between(m.Carrera, 1, 2)
WAIT "Debe especificar una Licenciara" WINDOW NOWAIT
_CUROBJ = ObjNum(m.Carrera)
RETURN .T.
Endf

```

```

If !Between(m.Semestre, 1, 3)
WAIT "Debe especificar un Semestre" WINDOW NOWAIT
_CUROBJ = ObjNum(m.Semestre)
RETURN .T.
Endf

```

```

If Empty (m.numexam) .OR. m.numexam <= 0
WAIT "Debe ser mayor a 0" WINDOW NOWAIT
_CUROBJ = ObjNum(m.numexam)
RETURN .T.
Else

```

```

If (m.numexam) > 5
WAIT "Como máximo 5 exámenes diferentes"
WINDOW NOWAIT
_CUROBJ = ObjNum(m.numexam)
RETURN .T.
Endf
Endf

```

```

DO CASE
CASE m.Carrera=2 AND m.Semestre=1

```

```

DO Guion6.spr
CASE m.Carrera=2 AND m.Semestre=2 OR m.Carrera=2
AND m.Semestre=3
DO Guion7.spr
OTHERWISE
DO Guion4.spr
ENDCASE

Else
a_Ayuda = c_Ayuda1
CLEAR READ
Endif

*
FUNCTION _rfj15zxh6 && Read Level When
*
* When Code from screen: GENEXAM
*
#REGION 1
c_Ayuda1 = a_Ayuda
a_Ayuda="Exámenes Genera"
ON KEY LABEL Shift+F1 do ayuda

#REGION 0
REGIONAL m.currarea, m.talkstat, m.compstat

IF SET("TALK") = "ON"
SET TALK OFF
m.talkstat = "ON"
ELSE
m.talkstat = "OFF"
ENDIF
m.compstat = SET("COMPATIBLE")
SET COMPATIBLE FOXPLUS

m.currarea = SELECT()

IF NOT WEXIST("guion4");
OR UPPER(WTITLE("GUION4")) == "GUION4.PJX";
OR UPPER(WTITLE("GUION4")) == "GUION4.SCX";
OR UPPER(WTITLE("GUION4")) == "GUION4.MNX";
OR UPPER(WTITLE("GUION4")) == "GUION4.PRG";
OR UPPER(WTITLE("GUION4")) == "GUION4.FRX";
OR UPPER(WTITLE("GUION4")) == "GUION4.QPR"
DEFINE WINDOW guion4;
FROM 8, 17;
TO 19,61;
TITLE " CAPTURA DEL GUION ";
FLOAT;
NOCLOSE;
SHADOW;
NOMINIMIZE;
DOUBLE;
COLOR SCHEME 5
ENDIF

#REGION 1
Private nOpcion, nPregunta, mcount, copias, Tiempo, Peso, TiemTot,
PesoTot, y, ReacTot, x, z, reg

STORE 0 TO
mcount,copias,Tiempo,Peso,TiemTot,PesoTot,y,ReacTot,x,z,reg

nPregunta = 0

#REGION 1
IF WVISIBLE("guion4")
ACTIVATE WINDOW guion4 SAME
ELSE
ACTIVATE WINDOW guion4 NOSHOW
ENDIF
@ 1,12 SAY "Area";
SIZE 1,4, 0
@ 1,16 SAY "s";
SIZE 1,1, 0
@ 7,4 SAY "Producción e l. de O.";
SIZE 1,21, 0
@ 3,4 SAY "Fundamentos de Admón.";
SIZE 1,21, 0
@ 4,4 SAY "Finanzas";
SIZE 1,8, 0
@ 6,4 SAY "Mercadotecnia";
SIZE 1,13, 0

```

```

@ 5,4 SAY "Admón. de Personal";
SIZE 1,18, 0
@ 0,1 TO 8,41
@ 3,26 TO 7,26
@ 2,2 TO 2,40
@ 2,1 SAY "+";
SIZE 1,1, 0
@ 2,41 SAY " ";
SIZE 1,1, 0
@ 8,1 TO 10,41
@ 8,26 SAY " ";
SIZE 1,1, 0
@ 8,1 SAY "+";
SIZE 1,1, 0
@ 8,41 SAY " ";
SIZE 1,1, 0
@ 1,28 SAY "No.Preguntas";
SIZE 1,12, 0
@ 1,26 TO 2,26
@ 0,26 SAY " ";
SIZE 1,1, 0
@ 2,26 SAY "+";
SIZE 1,1, 0
@ 3,33 GET m.pregunta6;
SIZE 1,2;
DEFAULT 0;
VALID _rf9176v9u()
@ 4,33 GET m.pregunta4;
SIZE 1,2;
DEFAULT 0;
VALID _rf9176vsc()
@ 5,33 GET m.pregunta1;
SIZE 1,2;
DEFAULT 0;
VALID _rf9176w5u()
@ 6,33 GET m.pregunta7;
SIZE 1,2;
DEFAULT 0;
VALID _rf9176wjl()
@ 7,33 GET m.pregunta8;
SIZE 1,2;
DEFAULT 0;
VALID _rf9176ww0()
@ 9,32 SAY nPregunta;
SIZE 1,3;
PICTURE "99"
@ 11,2 GET nOpcion;
PICTURE "@HN <Presentar;<Regresar";
SIZE 1,16,7;
DEFAULT 1;
VALID _rf9176xa6()
@ 9,4 SAY "Total: ";
SIZE 1,7, 0

IF NOT WVISIBLE("guion4")
ACTIVATE WINDOW guion4
ENDIF

READ CYCLE MODAL;
WHEN _rf9176xzl();
DEACTIVATE t.;
SHOW _rf9176y0d()

RELEASE WINDOW guion4
SELECT (m.currarea)

#REGION 0
IF m.talkstat = "ON"
SET TALK ON
ENDIF
IF m.compstat = "ON"
SET COMPATIBLE ON
ENDIF

*
FUNCTION _rf9176v9u && m.pregunta6 VALID
#REGION 1
If m.pregunta6 >= 0

nPregunta=m.pregunta6+m.pregunta4+m.pregunta1+m.pregunta7+m.preg
unta8
@ 9,25 SAY nPregunta
* @ 3,38 say TiemTot
* @ 3,49 say PesoTot
Else
WAIT "No puede ser Negativo" WINDOW NOWAIT
_CUROBJ = ObjNum(m.pregunta6)
m.pregunta6=0
RETURN .T.
Endif

```

```

*
FUNCTION _rf9176vsc  && m.pregunta4 VALID
#REGION 1
If m.pregunta4 >= 0

nPregunta=m.pregunta6+m.pregunta4+m.pregunta1+m.pregunta7+m.preg
unta8
  @ 9,25 SAY nPregunta
  * @ 3,38 say TiemTot
  * @ 3,49 say PesoTot
Else
  WAIT "No puede ser Negativo" WINDOW NOWAIT
  _CUROBJ = ObjNum(m.pregunta4)
  m.pregunta=0
  RETURN .T.
Endif
*

FUNCTION _rf9176w5u  && m.pregunta1 VALID
#REGION 1
If m.pregunta1 >= 0

nPregunta=m.pregunta6+m.pregunta4+m.pregunta1+m.pregunta7+m.preg
unta8
  @ 9,25 SAY nPregunta
  * @ 3,38 say TiemTot
  * @ 3,49 say PesoTot
Else
  WAIT "No puede ser Negativo" WINDOW NOWAIT
  _CUROBJ = ObjNum(m.pregunta1)
  m.pregunta=0
  RETURN .T.
Endif
*

FUNCTION _rf9176wji  && m.pregunta7 VALID
#REGION 1
If m.pregunta7 >= 0

nPregunta=m.pregunta6+m.pregunta4+m.pregunta1+m.pregunta7+m.preg
unta8
  @ 9,25 SAY nPregunta
  * @ 3,38 say TiemTot
  * @ 3,49 say PesoTot
Else
  WAIT "No puede ser Negativo" WINDOW NOWAIT
  _CUROBJ = ObjNum(m.pregunta7)
  m.pregunta=0
  RETURN .T.
Endif
*

FUNCTION _rf9176wv0  && m.pregunta8 VALID
#REGION 1
If m.pregunta8 >= 0

nPregunta=m.pregunta6+m.pregunta4+m.pregunta1+m.pregunta7+m.preg
unta8
  @ 9,25 SAY nPregunta
  * @ 3,38 say TiemTot
  * @ 3,49 say PesoTot
Else
  WAIT "No puede ser Negativo" WINDOW NOWAIT
  _CUROBJ = ObjNum(m.pregunta8)
  m.pregunta=0
  RETURN .T.
Endif
*

FUNCTION _rf9176xa6  && nOpcion VALID
#REGION 1
SHOW GETS
DO CASE
  CASE nOpcion=1 .AND. ITipo  && Exámenes Impresos
    If
      m.pregunta6+m.pregunta4+m.pregunta1+m.pregunta7+m.pregunta8 <= 0
      WAIT "No se pueden generar
exámenes sin preguntas" WINDOW NOWAIT
      _CUROBJ =
ObjNum(m.pregunta6)
      RETURN .T.
    Else
      DO exams
      WAIT " Puede continuar " WINDOW
NOWAIT
      DO presenta.spr      with .T.
    Endif
  CASE nOpcion=1 .AND. ITipo=.F.  && Exámenes
Interactivos
    If
      m.pregunta6+m.pregunta4+m.pregunta1+m.pregunta7+m.pregunta8 <= 0
      WAIT "No se pueden generar
exámenes sin preguntas" WINDOW NOWAIT

```

```

      _CUROBJ =
ObjNum(m.pregunta6)
      RETURN .T.
    Else
      DO exams
      WAIT " Puede continuar " WINDOW
NOWAIT
      DO Preseint.spr with .T.
    Endif
  OTHERWISE
    a_Ayuda = c_Ayuda2
    CLEAR READ
  ENDCASE
*

FUNCTION _rf9176xzl  && Read Level When
*
* When Code from screen: GUION4
*
#REGION 1
c_Ayuda2 = a_Ayuda
a_Ayuda="Exámenes Guion"
ON KEY LABEL Shift+F1 do ayuda

FUNCTION _rf9176y0d  && Read Level Show
PRIVATE currwind
STORE WOUTPUT() TO currwind
*
* Show Code from screen: GUION4
*
#REGION 1
IF SYS(2016) = "GUION4" OR SYS(2016) = ""
  ACTIVATE WINDOW guion4 SAME
  @ 9,32 SAY nPregunta ;
  SIZE 1,3, 0 ;
  PICTURE "99"
ENDIF
IF NOT EMPTY(currwind)
  ACTIVATE WINDOW (currwind) SAME
ENDIF
*

#REGION 1
Private nOpcion, nPregunta, mcount, copias, Tiempo, Peso, TiemTot,
PesoTot, y, ReacTot, x, z, reg

STORE 0 TO
mcount,copias,Tiempo,Peso,TiemTot,PesoTot,y,ReacTot,x,z,reg

nPregunta = 0

#REGION 0
REGIONAL m.currarea, m.talkstat, m.compstat

IF SET("TALK") = "ON"
  SET TALK OFF
  m.talkstat = "ON"
ELSE
  m.talkstat = "OFF"
ENDIF
m.compstat = SET("COMPATIBLE")
SET COMPATIBLE FOXPLUS

m.currarea = SELECT()
*

IF NOT WEXIST("guion6") ;
  OR UPPER(WTITLE("GUION6")) == "GUION6.PJX" ;
  OR UPPER(WTITLE("GUION6")) == "GUION6.SCX" ;
  OR UPPER(WTITLE("GUION6")) == "GUION6.MNX" ;
  OR UPPER(WTITLE("GUION6")) == "GUION6.PRG" ;
  OR UPPER(WTITLE("GUION6")) == "GUION6.FRX" ;
  OR UPPER(WTITLE("GUION6")) == "GUION6.QPR" ;
  DEFINE WINDOW guion6 ;
  FROM 6, 17 ;
  TO 17,61 ;
  TITLE " CAPTURA DEL GUION " ;
  FLOAT ;
  NOCLOSE ;
  SHADOW ;
  NOMINIMIZE ;
  DOUBLE ;

```

COLOR SCHEME 5

```

ENDIF
#REGION 1
IF WVISIBLE("guion6")
    ACTIVATE WINDOW guion6 SAME
ELSE
    ACTIVATE WINDOW guion6 NOSHOW
ENDIF
@ 3,3 SAY "Contabilidad y Costos";
    SIZE 1,21, 0
@ 5,3 SAY "Fiscal";
    SIZE 1,6, 0
@ 4,3 SAY "Finanzas";
    SIZE 1,8, 0
@ 0,1 TO 6,41
@ 3,25 TO 5,25
@ 2,1 TO 2,40
@ 2,25 SAY " ";
    SIZE 1,1, 0
@ 2,1 SAY "+";
    SIZE 1,1, 0
@ 2,41 SAY " ";
    SIZE 1,1, 0
@ 6,1 TO 8,41
@ 6,25 SAY " ";
    SIZE 1,1, 0
@ 6,1 SAY "+";
    SIZE 1,1, 0
@ 1,11 SAY "Area";
    SIZE 1,4, 0
@ 1,15 SAY "s";
    SIZE 1,1, 0
@ 3,32 GET m.pregunta3;
    SIZE 1,2;
    DEFAULT 0;
    VALID _rfc0uo9oq()
@ 4,32 GET m.pregunta4;
    SIZE 1,2;
    DEFAULT 0;
    VALID _rfc0uoa31()
@ 5,32 GET m.pregunta5;
    SIZE 1,2;
    DEFAULT 0;
    VALID _rfc0uoaeh()
@ 7,32 SAY nPregunta;
    SIZE 1,2;
    PICTURE "99"
@ 9,2 GET nOpcion;
    PICTURE "@*HN \<Presentar;\<Regresar";
    SIZE 1,16,7;
    DEFAULT 1;
    VALID _rfc0uocar()
@ 1,27 SAY "No.Preguntas";
    SIZE 1,12, 0
@ 7,3 SAY "Total :";
    SIZE 1,7, 0
@ 6,41 SAY " ";
    SIZE 1,1, 0
IF NOT WVISIBLE("guion6")
    ACTIVATE WINDOW guion6
ENDIF
READ CYCLE MODAL;
    WHEN _rfc0uobgz();
    DEACTIVATE .t;
    SHOW _rfc0uobhm()
RELEASE WINDOW guion6
SELECT (m.currarea)
#REGION 0
IF m.talkstat = "ON"
    SET TALK ON
ENDIF
IF m.compstat = "ON"
    SET COMPATIBLE ON
ENDIF
*
FUNCTION _rfc0uo9oq && m.pregunta3 VALID
#REGION 1
If m.pregunta3 >= 0
    nPregunta=m.pregunta3+m.pregunta4+m.pregunta5
    @ 7,24 SAY nPregunta
Else
    WAIT "No puede ser Negativo" WINDOW NOWAIT
    _CUROBJ = ObjNum(m.pregunta3)

```

```

m.pregunta3=0
RETURN .T.
EndIf
FUNCTION _rfc0uoa31 && m.pregunta4 VALID
#REGION 1
If m.pregunta4 >= 0
    nPregunta=m.pregunta3+m.pregunta4+m.pregunta5
    @ 7,24 SAY nPregunta
Else
    WAIT "No puede ser Negativo" WINDOW NOWAIT
    _CUROBJ = ObjNum(m.pregunta4)
    m.pregunta4=0
    RETURN .T.
EndIf
*
FUNCTION _rfc0uoaeh && m.pregunta5 VALID
#REGION 1
If m.pregunta5 >= 0
    nPregunta=m.pregunta3+m.pregunta4+m.pregunta5
    @ 7,24 SAY nPregunta
Else
    WAIT "No puede ser Negativo" WINDOW NOWAIT
    _CUROBJ = ObjNum(m.pregunta5)
    m.pregunta5=0
    RETURN .T.
EndIf
*
FUNCTION _rfc0uocar && nOpcion VALID
#REGION 1
SHOW GETS
DO CASE
    CASE nOpcion=1 .AND. ITipo
        If m.pregunta3+m.pregunta4+m.pregunta5 <= 0
            WAIT "No se pueden generar
            exámenes sin preguntas" WINDOW NOWAIT
            _CUROBJ =
            ObjNum(m.pregunta3)
            RETURN .T.
        Else
            DO Exams1
            WAIT " Puede continuar " WINDOW
            NOWAIT
            DO Presenta.spr with .T.
        EndIf
    CASE nOpcion=1 .AND. ITipo=F.
        If m.pregunta3+m.pregunta4+m.pregunta5 <= 0
            WAIT "No se pueden generar
            exámenes sin preguntas" WINDOW NOWAIT
            _CUROBJ =
            ObjNum(m.pregunta3)
            RETURN .T.
        Else
            DO Exams1
            WAIT " Puede continuar " WINDOW
            NOWAIT
            DO Preint.spr with .T.
        EndIf
    OTHERWISE
        a_Ayuda = c_Ayuda2
        CLEAR READ
ENDCASE
*
FUNCTION _rfc0uobgz && Read Level When
*
* When Code from screen: GUION6
*
#REGION 1
c_Ayuda2 = a_Ayuda
a_Ayuda="Exámenes Guion"
ON KEY LABEL Shift+F1 do ayuda
*
FUNCTION _rfc0uobhm && Read Level Show
PRIVATE currwind
STORE WOUTPUT() TO currwind
*
* Show Code from screen: GUION6
*
#REGION 1
IF SYS(2016) = "GUION6" OR SYS(2016) = ""
    ACTIVATE WINDOW guion6 SAME
    @ 7,32 SAY nPregunta;
    SIZE 1,2, 0;
    PICTURE "99"
ENDIF
IF NOT EMPTY(currwind)
    ACTIVATE WINDOW (currwind) SAME

```

ENDIF

#REGION 1

Private nOpcion, nPregunta, mcount, copias, Tiempo, Peso, TiemTot, PesoTot, y, ReacTot, x, z, reg

STORE 0 TO

mcount, copias, Tiempo, Peso, TiemTot, PesoTot, y, ReacTot, x, z, reg

nPregunta = 0

#REGION 0

REGIONAL m.currarea, m.talkstat, m.compstat

IF SET("TALK") = "ON"

SET TALK OFF
m.talkstat = "ON"

ELSE

m.talkstat = "OFF"

ENDIF

m.compstat = SET("COMPATIBLE")
SET COMPATIBLE FOXPLUS

m.currarea = SELECT()

IF NOT WEXIST("guion7"):

OR UPPER(WTITLE("GUION7")) = "GUION7.PJX";
OR UPPER(WTITLE("GUION7")) = "GUION7.SCX";
OR UPPER(WTITLE("GUION7")) = "GUION7.MNX";
OR UPPER(WTITLE("GUION7")) = "GUION7.PRG";
OR UPPER(WTITLE("GUION7")) = "GUION7.FRX";
OR UPPER(WTITLE("GUION7")) = "GUION7.QPR"

DEFINE WINDOW guion7
FROM 8, 17;
TO 18, 61;
TITLE " CAPTURA DEL GUION ";
FLOAT;
NOCLOSE;
SHADOW;
NOMINIMIZE;
DOUBLE;
COLOR SCHEME 5

ENDIF

#REGION 1

IF WVISIBLE("guion7")

ACTIVATE WINDOW guion7 SAME

ELSE

ACTIVATE WINDOW guion7 NOSHOW

ENDIF

@ 7,1 TO 9,41

@ 2,2 TO 2,40

@ 3,3 SAY "Contabilidad y Costos";
SIZE 1,21, 0

@ 5,3 SAY "Fiscal";
SIZE 1,6, 0

@ 6,3 SAY "Auditoria";
SIZE 1,9, 0

@ 4,3 SAY "Finanzas";
SIZE 1,8, 0

@ 1,13 SAY "Area";
SIZE 1,4, 0

@ 1,17 SAY "s";
SIZE 1,1, 0

@ 1,27 SAY "No Preguntas";
SIZE 1,12, 0

@ 0,1 TO 7,41

@ 2,25 SAY ".";
SIZE 1,1, 0

@ 7,25 SAY ".";
SIZE 1,1, 0

@ 3,25 TO 8,25

@ 7,41 SAY "m";
SIZE 1,1, 0

@ 7,1 SAY "+";
SIZE 1,1, 0

@ 3,32 GET m.pregunta3;
SIZE 1,2;
DEFAULT 0;

VALID _rfc0upzgw()

@ 4,32 GET m.pregunta4;
SIZE 1,2;
DEFAULT 0;

VALID _rfc0uq0e9()

@ 5,32 GET m.pregunta5;
SIZE 1,2;
DEFAULT 0;

VALID _rfc0uq0r7()

@ 6,32 GET m.pregunta2;
SIZE 1,2;
DEFAULT 0;
VALID _rfc0uq12r()

@ 8,32 SAY nPregunta;
SIZE 1,2;
PICTURE "99"

@ 10,2 GET nOpcion;
PICTURE "@*HN <Presentar,<Regresar";
SIZE 1,16,7;
DEFAULT 1;
VALID _rfc0uq1fz()

@ 2,41 SAY "m";
SIZE 1,1, 0

@ 2,1 SAY "+";
SIZE 1,1, 0

@ 8,3 SAY "Total .";
SIZE 1,7, 0

IF NOT WVISIBLE("guion7")

ACTIVATE WINDOW guion7

ENDIF

READ CYCLE MODAL;

WHEN _rfc0uq25i();
SHOW _rfc0uq262()

RELEASE WINDOW guion7

SELECT (m.currarea)

#REGION 0

IF m.talkstat = "ON"

SET TALK ON

ENDIF

IF m.compstat = "ON"

SET COMPATIBLE ON

ENDIF

+-----+

FUNCTION _rfc0upzgw && m.pregunta3 VALID

#REGION 1

If m.pregunta3 >= 0

nPregunta=m.pregunta2+m.pregunta3+m.pregunta4+m.pregunta5
@ 8,24 SAY nPregunta

Else

WAIT "No puede ser Negativo" WINDOW NOWAIT

_CUROBJ = ObjNum(m.pregunta3)

m.pregunta3=0

RETURN .T.

Endif

+-----+

FUNCTION _rfc0uq0e9 && m.pregunta4 VALID

#REGION 1

If m.pregunta4 >= 0

nPregunta=m.pregunta2+m.pregunta3+m.pregunta4+m.pregunta5
@ 8,24 SAY nPregunta

Else

WAIT "No puede ser Negativo" WINDOW NOWAIT

_CUROBJ = ObjNum(m.pregunta4)

m.pregunta4=0

RETURN .T.

Endif

FUNCTION _rfc0uq0r7 && m.pregunta5 VALID

#REGION 1

If m.pregunta5 >= 0

nPregunta=m.pregunta2+m.pregunta3+m.pregunta4+m.pregunta5
@ 8,24 SAY nPregunta

Else

WAIT "No puede ser Negativo" WINDOW NOWAIT

_CUROBJ = ObjNum(m.pregunta5)

m.pregunta5=0

RETURN .T.

Endif

FUNCTION _rfc0uq12r && m.pregunta2 VALID

#REGION 1

If m.pregunta2 >= 0

nPregunta=m.pregunta2+m.pregunta3+m.pregunta4+m.pregunta5
@ 8,24 SAY nPregunta

Else

WAIT "No puede ser Negativo" WINDOW NOWAIT

_CUROBJ = ObjNum(m.pregunta2)

m.pregunta2=0

RETURN .T.

Endif

```

*
FUNCTION _rfc0uq1fz  && nOpcion VALID
#REGION 1
SHOW GETS
DO CASE
    CASE nOpcion=1 .AND. ITipo
        if
            m.pregunta3+m.pregunta4+m.pregunta5+m.pregunta2 <= 0
                WAIT "No se pueden generar
exámenes sin preguntas" WINDOW NOWAIT
                _CUROBJ =
ObjNum(m.pregunta3)
                RETURN .T.
            Else
                DO Exams2
                WAIT " Puede continuar " WINDOW
NOWAIT
                DO presenta.spr      with .T.
            Endif
        CASE nOpcion=1 .AND. ITipo=.F.
            if
                m.pregunta3+m.pregunta4+m.pregunta5+m.pregunta2 <= 0
                    WAIT "No se pueden generar
exámenes sin preguntas" WINDOW NOWAIT
                    _CUROBJ =
ObjNum(m.pregunta3)
                    RETURN .T.
                Else
                    DO Exams2
                    WAIT " Puede continuar " WINDOW
NOWAIT
                    DO Preseint.spr with .T.
            Endif
        OTHERWISE
            a_Ayuda = c_Ayuda2
            CLEAR READ
ENDCASE

```

```

*
FUNCTION _rfc0uq25i  && Read Level When
*
* When Code from screen: GUION7
*
#REGION 1
c_Ayuda2 = a_Ayuda
a_Ayuda="Exámenes Guion"
ON KEY LABEL Shift+F1 do ayuda

```

```

*
FUNCTION _rfc0uq262  && Read Level Show
PRIVATE curwind
STORE WOUTPUT() TO curwind
*
* Show Code from screen: GUION7
*
#REGION 1
IF SYS(2016) = "GUION7" OR SYS(2016) = ""
    ACTIVATE WINDOW guion7 SAME
    @ 8,32 SAY nPregunta ;
    SIZE 1,2, 0 ;
    PICTURE "99"
ENDIF
IF NOT EMPTY(curwind)
    ACTIVATE WINDOW (curwind) SAME
ENDIF

```

```

*
#REGION 1
Private nOpcion, Texto, nPreg
STORE RECNO() TO nPreg

```

```

#REGION 0
REGIONAL m.currarea, m.talkstat, m.compstat
IF SET("TALK") = "ON"
    SET TALK OFF
    m.talkstat = "ON"
ELSE
    m.talkstat = "OFF"
ENDIF
m.compstat = SET("COMPATIBLE")
SET COMPATIBLE FOXPLUS
m.currarea = SELECT()

```

```

IF NOT WEXIST("exam_inter") ;
    OR UPPER(WTITLE("EXAM_INTER")) ==
"EXAM_INTER.PJX" ;
    OR UPPER(WTITLE("EXAM_INTER")) ==
"EXAM_INTER.SCX" ;
    OR UPPER(WTITLE("EXAM_INTER")) ==
"EXAM_INTER.MNX" ;
    OR UPPER(WTITLE("EXAM_INTER")) ==
"EXAM_INTER.PRG" ;
    OR UPPER(WTITLE("EXAM_INTER")) ==
"EXAM_INTER.FRX" ;
    OR UPPER(WTITLE("EXAM_INTER")) ==
"EXAM_INTER.QPR" ;
    DEFINE WINDOW exam_inter ;
    FROM 1, 1 ;
    TO 23,76 ;
    TITLE "EXAMEN INTERACTIVO " ;
    FLOAT ;
    NOCLOSE ;
    SHADOW ;
    NOMINIMIZE ;
    DOUBLE ;
    COLOR SCHEME 1
ENDIF

#REGION 1
IF WVISIBLE("exam_inter")
    ACTIVATE WINDOW exam_inter SAME
ELSE
    ACTIVATE WINDOW exam_inter NOSHOW
ENDIF
@ 0,1 TO 13,72 ;
    COLOR W+/B
@ 18,1 TO 20,72 ;
    COLOR W+/B
@ 13,2 SAY " Selecciona la opción correcta " ;
    SIZE 1,32, 0 ;
    COLOR W+/B
@ 0,2 SAY " 1. " ;
    SIZE 1,5, 0 ;
    COLOR W+/B
@ 13,70 SAY ". " ;
    SIZE 1,1, 0
@ 0,70 SAY "- " ;
    SIZE 1,1, 0
@ 0,3 SAY nPreg ;
    SIZE 1,2
@ 1,3 EDIT reactivo.texto ;
    SIZE 12,68,999 ;
    DEFAULT " " ;
    SCROLL ;
    COLOR SCHEME 10
@ 14,1 GET cRespuesta ;
    PICTURE "@*RVN \<a .\<b .\<c .\<d " ;
    SIZE 1,6,0 ;
    DEFAULT 1 ;
    COLOR SCHEME 8
@ 14,7 SAY IIF(SUBSTR(solucion, 1, 1) == "1", reactivo.op1,
IIF(SUBSTR(solucion, 1, 1) == "2", reactivo.op2, IIF(SUBSTR(solucion, 1, 1)
== "3", reactivo.op3, IIF(SUBSTR(solucion, 1, 1) == "4", reactivo.op4, " " )))
;
    SIZE 1,66 ;
    COLOR SCHEME 10
@ 15,7 SAY IIF(SUBSTR(solucion, 2, 1) == "1", reactivo.op1,
IIF(SUBSTR(solucion, 2, 1) == "2", reactivo.op2, IIF(SUBSTR(solucion, 2, 1)
== "3", reactivo.op3, IIF(SUBSTR(solucion, 2, 1) == "4", reactivo.op4, " " )))
;
    SIZE 1,66 ;
    COLOR SCHEME 10
@ 16,7 SAY IIF(SUBSTR(solucion, 3, 1) == "1", reactivo.op1,
IIF(SUBSTR(solucion, 3, 1) == "2", reactivo.op2, IIF(SUBSTR(solucion, 3, 1)
== "3", reactivo.op3, IIF(SUBSTR(solucion, 3, 1) == "4", reactivo.op4, " " )))
;
    SIZE 1,66 ;
    COLOR SCHEME 10
@ 17,7 SAY IIF(SUBSTR(solucion, 4, 1) == "1", reactivo.op1,
IIF(SUBSTR(solucion, 4, 1) == "2", reactivo.op2, IIF(SUBSTR(solucion, 4, 1)
== "3", reactivo.op3, IIF(SUBSTR(solucion, 4, 1) == "4", reactivo.op4, " " )))
;
    SIZE 1,66 ;
    COLOR SCHEME 10
@ 19,7 GET nOpcion ;
    PICTURE "@*HN \<Siguiente;\<Anterior;\<Finalizar" ;
    SIZE 1,14,9 ;
    DEFAULT 1 ;
    VALID _rfq13t9h5() ;
    COLOR SCHEME 11
IF NOT WVISIBLE("exam_inter")

```

```

ACTIVATE WINDOW exam_inter
ENDIF
READ CYCLE MODAL ;
  SHOW _rfq13tbqx()
RELEASE WINDOW exam_inter
SELECT (m.currarea)

#REGION 0
IF m.talkstat = "ON"
  SET TALK ON
ENDIF
IF m.compstat = "ON"
  SET COMPATIBLE ON
ENDIF

*
FUNCTION _rfq13t9h5  && nOpcion VALID
#REGION 1
DO CASE
  CASE nOpcion=1
    DO siguiente
  CASE nOpcion=2
    DO anterior
  OTHERWISE
    CLEAR READ
ENDCASE
RETURN

PROCEDURE siguiente
SKIP
nPreg=nPreg+1
IF EOF()
  GO TOP
  nPreg=1
ENDIF
@ 0,3 SAY nPreg SIZE 1,2
SHOW GET reactivo.texto
DO CASE
  CASE SUBSTR(solucion, 1, 1) == "1"
    @ 14,7 SAY reactivo.op1 SIZE 1,66 COLOR SCHEME 10
  CASE SUBSTR(solucion, 1, 1) == "2"
    @ 14,7 SAY reactivo.op2 SIZE 1,66 COLOR SCHEME 10
  CASE SUBSTR(solucion, 1, 1) == "3"
    @ 14,7 SAY reactivo.op3 SIZE 1,66 COLOR SCHEME 10
  OTHERWISE
    @ 14,7 SAY reactivo.op4 SIZE 1,66 COLOR SCHEME 10
ENDCASE
DO CASE
  CASE SUBSTR(solucion, 2, 1) == "1"
    @ 15,7 SAY reactivo.op1 SIZE 1,66 COLOR SCHEME 10
  CASE SUBSTR(solucion, 2, 1) == "2"
    @ 15,7 SAY reactivo.op2 SIZE 1,66 COLOR SCHEME 10
  CASE SUBSTR(solucion, 2, 1) == "3"
    @ 15,7 SAY reactivo.op3 SIZE 1,66 COLOR SCHEME 10
  OTHERWISE
    @ 15,7 SAY reactivo.op4 SIZE 1,66 COLOR SCHEME 10
ENDCASE
DO CASE
  CASE SUBSTR(solucion, 3, 1) == "1"
    @ 16,7 SAY reactivo.op1 SIZE 1,66 COLOR SCHEME 10
  CASE SUBSTR(solucion, 3, 1) == "2"
    @ 16,7 SAY reactivo.op2 SIZE 1,66 COLOR SCHEME 10
  CASE SUBSTR(solucion, 3, 1) == "3"
    @ 16,7 SAY reactivo.op3 SIZE 1,66 COLOR SCHEME 10
  OTHERWISE
    @ 16,7 SAY reactivo.op4 SIZE 1,66 COLOR SCHEME 10
ENDCASE
DO CASE
  CASE SUBSTR(solucion, 4, 1) == "1"
    @ 17,7 SAY reactivo.op1 SIZE 1,66 COLOR SCHEME 10
  CASE SUBSTR(solucion, 4, 1) == "2"
    @ 17,7 SAY reactivo.op2 SIZE 1,66 COLOR SCHEME 10
  CASE SUBSTR(solucion, 4, 1) == "3"
    @ 17,7 SAY reactivo.op3 SIZE 1,66 COLOR SCHEME 10
  OTHERWISE
    @ 17,7 SAY reactivo.op4 SIZE 1,66 COLOR SCHEME 10
ENDCASE
RETURN

PROCEDURE anterior
SKIP -1
nPreg=nPreg-1
IF BOF()
  GO BOTTOM
  nPreg=RECCOUNT()
ENDIF
@ 0,3 SAY nPreg SIZE 1,2
SHOW GET reactivo.texto
DO CASE
  CASE SUBSTR(solucion, 1, 1) == "1"

```

```

    @ 14,7 SAY reactivo.op1 SIZE 1,66 COLOR SCHEME 10
  CASE SUBSTR(solucion, 1, 1) == "2"
    @ 14,7 SAY reactivo.op2 SIZE 1,66 COLOR SCHEME 10
  CASE SUBSTR(solucion, 1, 1) == "3"
    @ 14,7 SAY reactivo.op3 SIZE 1,66 COLOR SCHEME 10
  OTHERWISE
    @ 14,7 SAY reactivo.op4 SIZE 1,66 COLOR SCHEME 10
ENDCASE
DO CASE
  CASE SUBSTR(solucion, 2, 1) == "1"
    @ 15,7 SAY reactivo.op1 SIZE 1,66 COLOR SCHEME 10
  CASE SUBSTR(solucion, 2, 1) == "2"
    @ 15,7 SAY reactivo.op2 SIZE 1,66 COLOR SCHEME 10
  CASE SUBSTR(solucion, 2, 1) == "3"
    @ 15,7 SAY reactivo.op3 SIZE 1,66 COLOR SCHEME 10
  OTHERWISE
    @ 15,7 SAY reactivo.op4 SIZE 1,66 COLOR SCHEME 10
ENDCASE
DO CASE
  CASE SUBSTR(solucion, 3, 1) == "1"
    @ 16,7 SAY reactivo.op1 SIZE 1,66 COLOR SCHEME 10
  CASE SUBSTR(solucion, 3, 1) == "2"
    @ 16,7 SAY reactivo.op2 SIZE 1,66 COLOR SCHEME 10
  CASE SUBSTR(solucion, 3, 1) == "3"
    @ 16,7 SAY reactivo.op3 SIZE 1,66 COLOR SCHEME 10
  OTHERWISE
    @ 16,7 SAY reactivo.op4 SIZE 1,66 COLOR SCHEME 10
ENDCASE
DO CASE
  CASE SUBSTR(solucion, 4, 1) == "1"
    @ 17,7 SAY reactivo.op1 SIZE 1,66 COLOR SCHEME 10
  CASE SUBSTR(solucion, 4, 1) == "2"
    @ 17,7 SAY reactivo.op2 SIZE 1,66 COLOR SCHEME 10
  CASE SUBSTR(solucion, 4, 1) == "3"
    @ 17,7 SAY reactivo.op3 SIZE 1,66 COLOR SCHEME 10
  OTHERWISE
    @ 17,7 SAY reactivo.op4 SIZE 1,66 COLOR SCHEME 10
ENDCASE
RETURN

*
FUNCTION _rfq13tbqx  && Read Level Show
PRIVATE currwind
STORE WOUTPUT() TO currwind
*
* Show Code from screen: EXAINTER
*
#REGION 1
IF SYS(2016) = "EXAM_INTER" OR SYS(2016) = ""
  ACTIVATE WINDOW exam_inter SAME
  @ 14,7 SAY IIF(SUBSTR(solucion, 1, 1) == "1", reactivo.op1,
  IIF(SUBSTR(solucion, 1, 1) == "2", reactivo.op2, IIF(SUBSTR(solucion, 1, 1)
  == "3", reactivo.op3, IIF(SUBSTR(solucion, 1, 1) == "4", reactivo.op4, " "))))
  SIZE 1,66, 0 ;
  COLOR SCHEME 10
  @ 15,7 SAY IIF(SUBSTR(solucion, 2, 1) == "1", reactivo.op1,
  IIF(SUBSTR(solucion, 2, 1) == "2", reactivo.op2, IIF(SUBSTR(solucion, 2, 1)
  == "3", reactivo.op3, IIF(SUBSTR(solucion, 2, 1) == "4", reactivo.op4, " "))))
  SIZE 1,66, 0 ;
  COLOR SCHEME 10
  @ 16,7 SAY IIF(SUBSTR(solucion, 3, 1) == "1", reactivo.op1,
  IIF(SUBSTR(solucion, 3, 1) == "2", reactivo.op2, IIF(SUBSTR(solucion, 3, 1)
  == "3", reactivo.op3, IIF(SUBSTR(solucion, 3, 1) == "4", reactivo.op4, " "))))
  SIZE 1,66, 0 ;
  COLOR SCHEME 10
  @ 17,7 SAY IIF(SUBSTR(solucion, 4, 1) == "1", reactivo.op1,
  IIF(SUBSTR(solucion, 4, 1) == "2", reactivo.op2, IIF(SUBSTR(solucion, 4, 1)
  == "3", reactivo.op3, IIF(SUBSTR(solucion, 4, 1) == "4", reactivo.op4, " "))))
  SIZE 1,66, 0 ;
  COLOR SCHEME 10
ENDIF
IF NOT EMPTY(currwind)
  ACTIVATE WINDOW (currwind) SAME
ENDIF

#REGION 1
Private clave, IAutoriza

IAutoriza = .F.
store ' ' to clave

#REGION 0
REGIONAL m.currarea, m.talkstat, m.compstat
IF SET("TALK") = "ON"

```

```

SET TALK OFF
m.talkstat = "ON"
ELSE
    m.talkstat = "OFF"
ENDIF
m.compstat = SET("COMPATIBLE")
SET COMPATIBLE FOXPLUS

m.currarea = SELECT()

IF NOT WEXIST("clave");
    OR UPPER(WTITLE("CLAVE")) == "CLAVE.PJX";
    OR UPPER(WTITLE("CLAVE")) == "CLAVE.SCX";
    OR UPPER(WTITLE("CLAVE")) == "CLAVE.MNX";
    OR UPPER(WTITLE("CLAVE")) == "CLAVE.PRG";
    OR UPPER(WTITLE("CLAVE")) == "CLAVE.FRX";
    OR UPPER(WTITLE("CLAVE")) == "CLAVE.QPR"
    DEFINE WINDOW clave;
        FROM INT((SROW()-5)/2),INT((SCOL()-28)/2);
        TO INT((SROW()-5)/2)+4,INT((SCOL()-28)/2)+27

        FLOAT;
        NOCLOSE;
        SHADOW;
        NOMINIMIZE;
        DOUBLE;
        COLOR SCHEME 5
    ENDIF

#REGION 1
IF WVISIBLE("clave")
    ACTIVATE WINDOW clave SAME
ELSE
    ACTIVATE WINDOW clave NOSHOW
ENDIF
@ 1,19 GET m.clave;
    SIZE 1,4;
    DEFAULT "";
    PICTURE "@!"
@ 1,1 SAY "CLAVE DE ACCESO";
    SIZE 1,15,0
@ 0,17 TO 2,24

IF NOT WVISIBLE("clave")
    ACTIVATE WINDOW clave
ENDIF

READ MODAL;
    SHOW .T.
@ 1, 19 SAY ""

RELEASE WINDOW clave
SELECT (m.currarea)

#REGION 0
IF m.talkstat = "ON"
    SET TALK ON
ENDIF
IF m.compstat = "ON"
    SET COMPATIBLE ON
ENDIF

#REGION 1
If !Empty(m.clave) And Seek(m.clave)
    IAutORIZA = T.
    clave=m.clave
Else
    WAIT WINDOW 'Clave no autorizada' NOWAIT
ENDIF
RETURN clave

+-----+

#define Falso .F.
#define Cierto .T.
..... TIPOS
#define MAT "1"
#define MAN "2"
#define EQU "3"
#define CUA "4"
#define MAQ "5"
#define BAS "6"
#define COM "7"
#define SUP "8"

```

```

#define TiposINS "123"
#define TiposMAT "45678"

SET SYSTEMU AUTOMATIC

DEFINE PAD Movimiento OF _MSYSMENU PROMPT "\<Movimientos"
COLOR SCHEME 3;
    KEY ALT+M, "ALT+M"
ON PAD Movimiento OF _MSYSMENU ACTIVATE POPUP movimiento

DEFINE POPUP movimiento MARGIN RELATIVE SHADOW COLOR
SCHEME 4
DEFINE BAR 1 OF movimiento PROMPT "\<Impresos";
    KEY ALT+F5, "ALT+F5";
    SKIP FOR !WonTop("Exámenes")
DEFINE BAR 2 OF movimiento PROMPT "\<Calificar";
    KEY ALT+F8, "ALT+F8";
    SKIP FOR !WonTop("Exámenes")
DEFINE BAR 3 OF movimiento PROMPT "\<";
DEFINE BAR 4 OF movimiento PROMPT "\<Interactivos";
    KEY ALT+F7, "ALT+F7";
    SKIP FOR !WonTop("Exámenes")
DEFINE BAR 5 OF movimiento PROMPT "\<Revisar";
    KEY ALT+F8, "ALT+F8";
    SKIP FOR !WonTop("Exámenes")
DEFINE BAR 6 OF movimiento PROMPT "\<";
DEFINE BAR 7 OF movimiento PROMPT "\<Termina";
    KEY CTRL+T, "CTRL+T";
    SKIP FOR Wexist("CreaExamen")
ON SELECTION BAR 1 OF movimiento DO GenExam.spr with .T.,.T.
ON SELECTION BAR 2 OF movimiento DO CalifImp.spr with .F.
ON SELECTION BAR 4 OF movimiento DO Interac.spr With .T.,.F.
ON SELECTION BAR 5 OF movimiento DO Interac.spr With .F.
ON SELECTION BAR 7 OF movimiento RELEASE WINDOW "Exámenes"

SET MARK OF MENU _MSYSMENU TO ""

#define Falso .F.
#define Cierto .T.
..... TIPOS
#define MAT "1"
#define MAN "2"
#define EQU "3"
#define CUA "4"
#define MAQ "5"
#define BAS "6"
#define COM "7"
#define SUP "8"
#define TiposINS "123"
#define TiposMAT "45678"

SET SYSTEMU AUTOMATIC

DEFINE PAD Movimiento OF _MSYSMENU PROMPT "\<Movimientos"
COLOR SCHEME 3;
    KEY ALT+M, "ALT+M"
ON PAD Movimiento OF _MSYSMENU ACTIVATE POPUP movimiento

DEFINE POPUP movimiento MARGIN RELATIVE SHADOW COLOR
SCHEME 4
DEFINE BAR 1 OF movimiento PROMPT "\<Aplicar";
    KEY SHIFT+F5, "SHIFT+F5";
    SKIP FOR !WonTop("Exámenes Interactivos")
DEFINE BAR 2 OF movimiento PROMPT "\<Revisar";
    KEY SHIFT+F8, "SHIFT+F8";
    SKIP FOR !WonTop("Exámenes Interactivos")
DEFINE BAR 3 OF movimiento PROMPT "\<";
DEFINE BAR 4 OF movimiento PROMPT "\<Termina";
    KEY CTRL+T, "CTRL+T";
    SKIP FOR Wexist("ExamenInt")
ON SELECTION BAR 1 OF movimiento DO Interac.spr With .T.
ON SELECTION BAR 2 OF movimiento DO Interac.spr With .F.
ON SELECTION BAR 4 OF movimiento RELEASE WINDOW "Exámenes
Interactivos"

SET MARK OF MENU _MSYSMENU TO ""

SET SYSTEMU AUTOMATIC

DEFINE PAD Movimiento OF _MSYSMENU PROMPT "\<Movimientos"
COLOR SCHEME 3;
    KEY ALT+M, "ALT+M"
ON PAD Movimiento OF _MSYSMENU ACTIVATE POPUP movimiento

DEFINE POPUP movimiento MARGIN RELATIVE SHADOW COLOR
SCHEME 4
DEFINE BAR 1 OF movimiento PROMPT "\<Agregar";

```

```

KEY ALT+F5, "ALT+F5";
SKIP FOR !WonTop("Maestros") or Wexist("Control_M") or
Wexist("Elimina_M") or Wexist("Captura_M")
DEFINE BAR 2 OF movimiento PROMPT "\<Editar";
KEY ALT+F8, "ALT+F8";
SKIP FOR !WonTop("Maestros") or Wexist("Control_M") or
Wexist("Elimina_M") or Wexist("Captura_M")
DEFINE BAR 3 OF movimiento PROMPT "El<liminar";
KEY ALT+F7, "ALT+F7";
SKIP FOR !WonTop("Maestros") or Wexist("Control_M") or
Wexist("Elimina_M") or Wexist("Captura_M")
DEFINE BAR 4 OF movimiento PROMPT "\<";
DEFINE BAR 5 OF movimiento PROMPT "\<Termina";
KEY CTRL+T, "CTRL+T";
ON SELECTION BAR 1 OF movimiento Do CapturaM.Spr With .T.
ON SELECTION BAR 2 OF movimiento Do EditaM.Spr
ON SELECTION BAR 5 OF movimiento RELEASE WINDOW "Maestros"

SET MARK OF MENU _MSYSMENU TO ""

```

```

#define Falso .F.
#define Cierto .T.
#define Activos .T.
..... TIPOS
#define MAT "1"
#define MAN "2"
#define EQU "3"
#define CJA "4"
#define MAQ "5"
#define BAS "6"
#define COM "7"
#define SUP "8"
#define TiposINS "123"
#define TiposMAT "45678"

```

SET SYSTEMU AUTOMATIC

```

DEFINE PAD Movimiento OF _MSYSMENU PROMPT "\<Movimientos"
COLOR SCHEME 3;
KEY ALT+M, "ALT+M"
ON PAD Movimiento OF _MSYSMENU ACTIVATE POPUP movimiento

```

```

DEFINE POPUP movimiento MARGIN RELATIVE SHADOW COLOR
SCHEME 4
DEFINE BAR 1 OF movimiento PROMPT "\<Agregar";
KEY ALT+F5, "ALT+F5";
SKIP FOR !WonTop("Reactivos") or Wexist("Captura_R") or
Wexist("Elimina_R")
DEFINE BAR 2 OF movimiento PROMPT "\<Editar";
KEY ALT+F8, "ALT+F8";
SKIP FOR !WonTop("Reactivos") or Wexist("Captura_R") or
Wexist("Elimina_R")
DEFINE BAR 3 OF movimiento PROMPT "El<liminar";
KEY ALT+F7, "ALT+F7";
SKIP FOR !WonTop("Reactivos") or Wexist("Captura_R") or
Wexist("Elimina_R")
DEFINE BAR 4 OF movimiento PROMPT "\<";
DEFINE BAR 5 OF movimiento PROMPT "\<Busca reactivo";
SKIP FOR !WonTop("Reactivos") or Wexist("Captura_R") or
Wexist("Elimina_R")
DEFINE BAR 6 OF movimiento PROMPT "\<Termina";
KEY CTRL+T, "CTRL+T";
SKIP FOR Wexist("Captura_R") or Wexist("Elimina_R")
ON SELECTION BAR 1 OF movimiento Do CapturaR.Spr With .T.
ON SELECTION BAR 2 OF movimiento Do CapturaR.Spr With .F.
ON SELECTION BAR 5 OF movimiento Do SelfReg.spr
ON SELECTION BAR 6 OF movimiento RELEASE WINDOW "Reactivos"

```

SET MARK OF MENU _MSYSMENU TO ""

SET SYSTEMU TO

SET SYSTEMU AUTOMATIC

```

DEFINE PAD _MSM_FILE OF _MSYSMENU PROMPT "\<Impresora"
COLOR SCHEME 3;
KEY ALT+F, ""
ON PAD _MSM_FILE OF _MSYSMENU ACTIVATE POPUP impresora

```

```

DEFINE POPUP impresora MARGIN RELATIVE SHADOW COLOR
SCHEME 4
DEFINE BAR _MFI_SETUP OF impresora PROMPT "Pr<inter Setup..."
DEFINE BAR _MFI_PRINT OF impresora PROMPT "\<Print..."

```

SET SYSTEMU TO

SET SYSTEMU AUTOMATIC

```

DEFINE PAD Sistema OF _MSYSMENU PROMPT "\<Sistema" COLOR
SCHEME 3;
KEY ALT+S, ""

```

```

DEFINE PAD Edición OF _MSYSMENU PROMPT "\<Edición" COLOR
SCHEME 3;
KEY ALT+E, ""

```

```

DEFINE PAD Ventana OF _MSYSMENU PROMPT "\<Ventana" COLOR
SCHEME 3;
KEY ALT+V, ""

```

```

ON PAD Sistema OF _MSYSMENU ACTIVATE POPUP sistema
ON PAD Edición OF _MSYSMENU ACTIVATE POPUP edición
ON PAD Ventana OF _MSYSMENU ACTIVATE POPUP ventana

```

```

DEFINE POPUP sistema MARGIN RELATIVE SHADOW COLOR SCHEME
4

```

```

DEFINE BAR 1 OF sistema PROMPT "\<S.E.E.C."

```

```

DEFINE BAR 2 OF sistema PROMPT "\<Ayuda";
KEY SHIFT+F1, "SHIFT+F1"

```

```

DEFINE BAR 3 OF sistema PROMPT "\<";

```

```

DEFINE BAR 4 OF sistema PROMPT "\<Maestros";
KEY ALT+M, "ALT+M";
SKIP FOR Wexist("Maestros") .OR. Wexist("Reactivos") .OR.
Wexist("Exámenes")

```

```

DEFINE BAR 5 OF sistema PROMPT "\<Reactivos";
KEY ALT+R, "ALT+R";
SKIP FOR Wexist("Maestros") .OR. Wexist("Reactivos") .OR.
Wexist("Exámenes")

```

```

DEFINE BAR 6 OF sistema PROMPT "\<";

```

```

DEFINE BAR 7 OF sistema PROMPT "\<Exámenes";
KEY ALT+E, "ALT+E";
SKIP FOR Wexist("Maestros") .OR. Wexist("Reactivos") .OR.
Wexist("Exámenes")

```

```

DEFINE BAR 8 OF sistema PROMPT "Report<es";
KEY ALT+T, "ALT+T";
SKIP FOR Wexist("Maestros") .OR. Wexist("Reactivos") .OR.
Wexist("Exámenes")

```

```

DEFINE BAR 9 OF sistema PROMPT "\<";

```

```

DEFINE BAR 10 OF sistema PROMPT "Sal<tir";
KEY CTRL+X, "CTRL+X";
SKIP FOR Wexist("Maestros") .OR. Wexist("Reactivos") .OR.
Wexist("Exámenes")

```

```

ON SELECTION BAR 1 OF sistema DO SEEC.SPR
ON SELECTION BAR 2 OF sistema DO AYUDA.PRG
ON SELECTION BAR 4 OF sistema DO MAESTROS.PRG
ON SELECTION BAR 5 OF sistema DO REACTIVO.PRG
ON SELECTION BAR 7 OF sistema DO EXAMENES.PRG
ON BAR 8 OF sistema ACTIVATE POPUP reportes
ON SELECTION BAR 10 OF sistema sigue=.F.

```

```

DEFINE POPUP reportes MARGIN RELATIVE SHADOW COLOR
SCHEME 4

```

```

DEFINE BAR 1 OF reportes PROMPT "\<Maestros";
SKIP FOR nTipo <> 1

```

```

DEFINE BAR 2 OF reportes PROMPT "\<Reactivos"

```

```

DEFINE BAR 3 OF reportes PROMPT "Result. <Interactivos"
ON SELECTION BAR 1 OF reportes DO ReporteM.PRG
ON SELECTION BAR 2 OF reportes DO Repor_R.spr

```

```

DEFINE POPUP edición MARGIN RELATIVE SHADOW COLOR SCHEME
4

```

```

DEFINE BAR _MED_CUT OF edición PROMPT "Co<rtar";
KEY CTRL+X, "AX"

```

```

DEFINE BAR 2 OF edición PROMPT "\<";

```

```

DEFINE BAR _MED_COPY OF edición PROMPT "\<Copiar";
KEY CTRL+C, "AC"

```

```

DEFINE BAR 4 OF edición PROMPT "\<";

```

```

DEFINE BAR _MED_PASTE OF edición PROMPT "Pe<gar";
KEY CTRL+V, "AV"

```

```

DEFINE POPUP ventana MARGIN RELATIVE SHADOW COLOR SCHEME
4

```

```

DEFINE BAR _MWI_SIZE OF ventana PROMPT "\<Tamaño";
KEY CTRL+F8, "AF8"

```

```

DEFINE BAR _MWI_ZOOM OF ventana PROMPT "Ma<ximizar □";
KEY CTRL+F10, "AF10"

```

```

DEFINE BAR _MWI_MIN OF ventana PROMPT "Mi<nimizar □";
KEY CTRL+F9, "AF9"

```

```

DEFINE BAR _MWI_COLOR OF ventana PROMPT "Co<lor..."

```

SET MARK OF MENU _MSYSMENU TO ""

PARAMETER prog

If an application READ is already active,

```

*
IF RDLEVEL() > 1
  tobedone = prog
  CLEAR READ
ELSE
*
  DO (prog)
ENDIF

PROCEDURE mexit
  dropdead = .T.
  CLEAR READ ALL

*SELECT 6
*GOTO TOP

do datos.spr
*Do exainter.spr

*CREATE CURSOR Personal (ExaNum N(4), Clave C(5), Solucion C(5),
Tiempo N(2), Peso N(2))

Parameters Inew
#define cFile 'EXA'+transform(n,'@L 9999')+'.DBF'
n=1

IF Inew
  cAuxFile=' '
  FOR n=1 to 9999
    IF FILE (cFile)
      cAuxFile=cFile
    ELSE
      cAuxFile=cFile
      n=VAL(SUBSTR(cAuxFile, 4, 4))
      n=n+1
      IF not FILE (cFile)
        n=n-1
        exit
      ENDIF
    ENDIF
  NEXT n
ENDIF

ENDIF

select auxiliar
copy structure to cFile
copy to cFile
USE cFile
SELECT cFile
INDEX ON CLAVE TAG CLAVE ADDITIVE
SET ORDER TO

SELECT reactivo
SET ORDER TO TAG clave
SELECT cFile
SET RELATION TO clave INTO reactivo

cArchivo=cFile
RETURN cArchivo

Parameters nBar2

DEFINE POPUP Area MARGIN RELATIVE SHADOW COLOR
SCHEME 4 && Create multi-choice popup
DEFINE BAR 1 OF Area PROMPT '\<Admón. de Personal'
DEFINE BAR 2 OF Area PROMPT 'Audí<toría'
DEFINE BAR 3 OF Area PROMPT '\<Contabilidad y Costos'
DEFINE BAR 4 OF Area PROMPT '\<Finanzas'
DEFINE BAR 5 OF Area PROMPT 'Fiscal<'
DEFINE BAR 6 OF Area PROMPT 'Fun<damentos de
Admón.'
DEFINE BAR 7 OF Area PROMPT '\<Mercadotecnia'
DEFINE BAR 8 OF Area PROMPT '\<Producción e I. de O.'

ON SELECTION POPUP Area DEACTIVATE POPUP Area
ACTIVATE POPUP Area BAR nBar2 AT Row(), Col()
RELEASE POPUP AREA
If Bar() > 0
  nBar2 = Bar()
Endif

Return nBar2

#define AREAS "Admón. de Personal Auditoría Contabilidad
y Costos Finanzas Fiscal Fundamentos de
Admón. Mercadotecnia Producción e Inv. de Op. "
Parameters nArea

```

```

RETURN IIF( Between(nArea, 1, 8), SubStr(AREAS, (nArea - 1) * 26 + 1,
26), Space(26) )

```

```

Parameters nBar1
DEFINE POPUP Carrera MARGIN RELATIVE SHADOW
COLOR SCHEME 4 && Create multi-choice popup
DEFINE BAR 1 OF Carrera PROMPT
'\<Administración'
DEFINE BAR 2 OF Carrera PROMPT
'\<Contabilidad'
DEFINE BAR 3 OF Carrera PROMPT 'Admón. \<y
Conta.'

ON SELECTION POPUP Carrera DEACTIVATE POPUP
Carrera
ACTIVATE POPUP Carrera BAR nBar1 AT Row(), Col()
RELEASE POPUP Carrera
If Bar() > 0
  nBar1 = Bar()
Endif

Return nBar1

```

```

#define CARRERAS "Administración Contabilidad Admón. y Conta. "
Parameters nCarrera
RETURN IIF( Between(nCarrera, 1, 3), SubStr(CARRERAS, (nCarrera - 1) *
16 + 1, 16), Space(16) )

```

```

#define Falso .F.
#define Cierto .T.
* ..... TIPOS
#define MAT "1"
#define MAN "2"
#define EQU "3"
#define CUA "4"
#define MAQ "5"
#define BAS "6"
#define COM "7"
#define SUP "8"
#define TiposINS "123"
#define TiposMAT "45678"

```

```

SET SYSTEMMENU AUTOMATIC

```

```

DEFINE PAD Movimiento OF _MSYSMENU PROMPT '\<Movimientos'
COLOR SCHEME 3 ;
KEY ALT+M, "ALT+M"
ON PAD Movimiento OF _MSYSMENU ACTIVATE POPUP movimiento

```

```

DEFINE POPUP movimiento MARGIN RELATIVE SHADOW COLOR
SCHEME 4
DEFINE BAR 1 OF movimiento PROMPT '\<Impresos' ;
KEY ALT+F5, "ALT+F5" ;
SKIP FOR !WonTop("Exámenes")
DEFINE BAR 2 OF movimiento PROMPT '\<Calificar' ;
KEY ALT+F6, "ALT+F6" ;
SKIP FOR !WonTop("Exámenes")
DEFINE BAR 3 OF movimiento PROMPT "\<"
DEFINE BAR 4 OF movimiento PROMPT "\<Interactivos" ;
KEY ALT+F7, "ALT+F7" ;
SKIP FOR !WonTop("Exámenes")
DEFINE BAR 5 OF movimiento PROMPT "\<Revisar" ;
KEY ALT+F8, "ALT+F8" ;
SKIP FOR !WonTop("Exámenes")
DEFINE BAR 6 OF movimiento PROMPT "\<"
DEFINE BAR 7 OF movimiento PROMPT "\<Termina" ;
KEY CTRL+T, "CTRL+T" ;
SKIP FOR Wexist("CreaExamen")
ON SELECTION BAR 1 OF movimiento DO GenExam.spr with .T.,.T.
ON SELECTION BAR 2 OF movimiento DO CalifImp.spr with .F.
ON SELECTION BAR 4 OF movimiento DO Interac.spr With .T.,.F.
ON SELECTION BAR 5 OF movimiento DO Interac.spr With .F.
ON SELECTION BAR 7 OF movimiento RELEASE WINDOW "Exámenes"

```

```

SET MARK OF MENU _MSYSMENU TO ""

```

```

PROCEDURE TERMINA
  CLEAR
  ON ERROR DO erhand WITH ERROR(), MESSAGE()

```

PROCEDURE TERMINA

CLEAR

CLOSE ALL

CLOSE ALTERNATE

CLOSE DATABASES

CLOSE FORMAT

CLOSE INDEXES

CLOSE PROCEDURE

QUIT

RETURN

RETURN

RODOLFO P. ...
Código de ...
Liderato ...
Miguel ...

...
...
...
...

EMERITO ...
...
...
...

JOSÉ ...
...
...
...

JUAN ...
...
...
...

BIBLIOGRAFIA

MARIO TAMAYO Y TAMAYO
El Proceso de la Investigación Científica
Fundamentos de investigación
Editorial LIMUSA.
México, D.F. 1986

NOE PÉREZ ÁVILA
Cómo hacer una investigación
Editorial EDIPSA. Primera edición
México, D.F. 1989

ARTURO ROSEABLUETH
El Método Científico
Editorial Mc. Graw Hill. Primera edición
México, D.F. 1985

UMBERTO ECO
Cómo hacer una Tesis
Editorial GEDISA
Barcelona, España, 1994

JOHN G. BURCH JR., FÉLIX R. STRATER JR.
Sistemas de Información Teoría y Práctica
Editorial LIMUSA.
México, D.F. 1992

JAMES A. SENN
Análisis y Diseño de Sistemas de Información
Editorial Mc. Graw Hill. Segunda edición
México, D.F. 1994

ROGER S. PRESMAN

Ingeniería del Software, un Enfoque Práctico

Editorial Mc. Graw Hill. Tercera edición

México, D.F. 1993

ANTONIO ARRANZ RAMONET

Administración de datos y archivos por computadora

Editorial LIMUSA. Primera edición

México, D.F. 1984

FORSYTHE, KEENAN, ORGANICK, STENBERG

Lenguajes y Diagramas de Flujo, Técnicas de Computación

Editorial LIMUSA Cuarta edición.

México, D.F. 1979

LELAND L. BECK

Software de Sistemas. Introducción a la Programación de Sistemas

Editorial Addison-Wesley Iberoamericana

México, D.F. 1988

JOSE JAVIER GARCIA BADEL

FoxPro 2.5 para MS-DOS y para WINDOWS a su alcance

Editorial Mc. Graw Hill. Primera edición

México, D.F. 1994

BILL CHAMBERS

Aplicaciones de Gestiones con FoxPro.

Metodologías para el Diseño de Aplicaciones en Empresas

Editorial Anaya Multimedia América. Tercera edición

México, D.F. 1994

