

7 872748



## Universidad Don Vasco, A.C.

---- INCORPORACIÓN No. 8727-48 ----  
a la Universidad Nacional Autónoma de México  
Escuela de Informática

*Desarrollo de un sistema de evaluación  
para el desempeño académico de los  
docentes de las escuelas de Administración,  
Contaduría e Informática de la  
Universidad Don Vasco, A.C.*

### T E S I S

Que para obtener el título de:

**LICENCIADO EN INFORMÁTICA**

presenta:

***Rafael Ignacio Farfán Navarro***



*Uruapan, Michoacán, Julio del 2000.*



Universidad Nacional  
Autónoma de México



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

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

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

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

# Agradecimientos

---

Si me preguntasen cual fue la cuestión de la tesis que más me llevó tiempo en determinar, podría asegurar que fue precisamente esta.

Este es mi primer trabajo a gran escala donde tengo que aplicar los conocimientos adquiridos en mi vida como estudiante. Esta es la primera hoja que se leerá a cerca de mi trabajo, el cual efectivamente en parte es mío, pero logrado gracias a un gran número de personas que con su dedicación, enseñanza, apoyo y cariño me formaron hasta llegar a ser un profesionista. Por ello veo necesario mencionar, si no a todos por que me falla la memoria, si a la mayoría de las personas que tuvieron que ver en mi formación.

Primero que nada quiero agradecer a Dios por que me ha dado la oportunidad de terminar mis estudios y poder ser útil a la sociedad.

A mis padres, Rafael Farfán Soria y Luz. Ma. Navarro Barragán. Por que no solo me dieron la vida, si no me brindaron la oportunidad de superarme como persona.

Ignacio Navarro Quiroz y a Elvira Barragán, por ser un ejemplo a seguir, por ser una guía inicial en mi vida, gracias por ser los mejores abuelitos del mundo.

Erika Amezcua, porque con su apoyo y amor cualquier dificultad parece simple.

A mi asesor de tesis, el Ing. Manuel López Gallegos, por sus acertadas instrucciones, por el conocimiento que compartió conmigo y sobre todo por ser mi amigo.

Lic. Sergio Quintero, un gran amigo y excelente asesor de tesis, siempre afable y sobre todo, muy profesional. Lic. Catalina Núñez, gracias por pulir mi tesis y por tu amistad. Lic. Martín Sánchez, muchas gracias por tu apoyo que hizo que mi proyecto se llevara a cabo sin tropiezos mayores y a Laura Mata por ayudarme, una y otra vez hasta el final, a darle la mejor presentación a este proyecto.

*-Rafael Farfán Navarro*

# Índice General

---

|  |           |
|--|-----------|
| Índice   | 1         |
| Introducción   | 3         |
| <b>Capítulo I Informática</b>                                  | <b>6</b>  |
| 1.1 Antecedentes   | 7         |
| 1.2 Las generaciones de las computadoras                       | 9         |
| 1.2.1 Primera Generación                                       | 10        |
| 1.2.2 Segunda Generación                                       | 11        |
| 1.2.3 Tercera Generación                                       | 11        |
| 1.2.4 Cuarta Generación  | 12        |
| 1.2.5 Quinta Generación  | 13        |
| 1.3 Componentes de la computadora                              | 13        |
| 1.3.1 Hardware   | 16        |
| 1.3.2 Software   | 17        |
| 1.3.2.1 Sistemas operativos                                    | 18        |
| 1.3.2.2 Programas de Aplicación                                | 18        |
| 1.3.2.3 Utilerías  | 19        |
| 1.3.2.4 Lenguajes de programación                              | 19        |
| <b>Capítulo II Ciclo de vida de desarrollo de los sistemas</b> | <b>22</b> |
| 2.1 Definición e Importancia                                   | 23        |
| 2.2 Ciclo de vida de los sistemas                              | 25        |
| 2.2.1 Identificación del problema                              | 25        |
| 2.2.2 Oportunidades y Objetivos                                | 25        |
| 2.2.3 Determinación de requerimientos de información           | 27        |
| 2.2.4 Análisis de las necesidades del sistema                  | 30        |
| 2.2.5 Diseño del sistema recomendado                           | 31        |
| 2.2.6 Desarrollo y documentación del software                  | 33        |
| 2.2.7 Prueba y mantenimiento del sistema                       | 34        |
| 2.2.8 Implementación y evaluación del sistema                  | 35        |
| <b>Capítulo III Capítulo de Diseño</b>                         | <b>37</b> |
| 3.1 Documentación de la base de datos                          | 37        |
| 3.2 Pseudocódigo   | 46        |
| 3.3 Diseño de entradas   | 59        |
| 3.4 Diseño de Salidas  | 82        |
| 3.5 Diagrama de base de datos                                  | 86        |
| 3.6 Diagrama de Clases   | 87        |
| 3.7 Programación   | 88        |
| <b>Capítulo IV Lenguajes de programación</b>                   | <b>94</b> |
| 4.1 Tipos de lenguajes   | 94        |
| 4.1.1 Lenguaje máquina y lenguajes simbólicos                  | 94        |
| 4.1.2 Lenguajes imperativos                                    | 95        |

|                     |  |     |
|---------------------|--|-----|
| 4.1.3               | Lenguajes declarativos                           | 96  |
| 4.1.4               | Lenguajes de programación lógica                 | 96  |
| 4.1.5               | Lenguajes orientados a objetos                   | 97  |
| 4.1.6               | Lenguajes persistentes                           | 98  |
| 4.1.7               | Lenguajes concurrentes                           | 98  |
| 4.1.8               | Lenguajes visuales                               | 98  |
| 4.2                 | Intérprete y compilador                          | 99  |
| <b>Capítulo V</b>   | <b>Ingeniería de software</b>                    | 101 |
| 5.1                 | Análisis y diseño de sistema de información      | 101 |
| 5.2                 | Paradigmas de ingeniería                         | 102 |
| 5.2.1               | El ciclo de vida clásico                         | 102 |
| 5.2.2               | Creación de prototipos                           | 105 |
| 5.2.3               | El modelo en espiral                             | 106 |
| 5.2.4               | Técnicas de cuarta generación (T4G)              | 109 |
| 5.3                 | Fases de ingeniería de software                  | 112 |
| 5.4                 | Análisis de sistemas                             | 113 |
| 5.4.1               | Pasos para el análisis de los sistemas           | 114 |
| 5.5                 | Análisis Orientado a Objetos                     | 115 |
| 5.5.1               | Modelización de los datos                        | 117 |
| <b>Capítulo VI</b>  | <b>Estudio de caso del sistema de evaluación</b> | 122 |
| 6.1                 | Antecedentes de la Universidad Don Vasco A.C.    | 122 |
| 6.2                 | Metodología utilizada                            | 123 |
| 6.3                 | Procedimientos actuales de evaluación            | 124 |
| 6.4                 | Identificación de requerimientos                 | 126 |
| 6.5                 | Descripción del problema                         | 126 |
| 6.6                 | Estudio de viabilidad                            | 130 |
| 6.7                 | Análisis económico                               | 131 |
| 6.8                 | Definiendo los límites del sistema               | 140 |
| 6.9                 | Escenarios                                       | 143 |
| 6.10                | Diagramas de Gantt                               | 149 |
| 6.10.1              | Diagrama Proyectado                              | 149 |
| 6.10.2              | Diagrama Real                                    | 160 |
| 6.11                | Prueba de sistemas                               | 172 |
| 6.11.1              | Conclusiones                                     | 200 |
| <b>Conclusiones</b> |  | 204 |
| <b>Bibliografía</b> |  | 211 |
| <b>Hemerografía</b> |  | 216 |
| <b>Anexos</b>       |  | 217 |

# Introducción

---

Actualmente las necesidades de aprendizaje se han vuelto más complejas y diversas que antaño. No hace falta más que comparar las materias que se llevaban hace unos años con las del periodo escolar en curso. Las instituciones dedicadas a la enseñanza han de contar con diversos recursos para aprendizaje eficiente por parte del alumno.

Hoy en día la informática ha tomado una importancia impensable hasta hace unos 10 años, en algunas las empresas ya no se califican solamente los aspectos culturales en las personas, sino además sus conocimientos de cómputo.

La clasificación de analfabeta ahora se extiende y no sólo se aplica a aquellos que no saben escribir y leer sino también abarca a los que no saben manejar una computadora y también, aunque en menor grado, a los que no saben hablar el idioma Inglés puesto que en su mayoría los desarrollos de software están en dicho Idioma.

Los médicos utilizan sistemas de cómputo para realizar diagnosis, licenciados en derecho encuentran fácilmente casos parecidos y aspectos legales utilizando una base de datos, los ingenieros realizan sus diseños en simuladores de computadora para saber el comportamiento real de la estructura una vez terminada. en fin.

Es esencial para cualquier profesional, sin importar la especialización como hemos visto, poder hacer uso de las herramientas informáticas que tiene al alcance por medio de

la computadora, por lo cual es imposible que un profesional sea competitivo si no tiene las bases de informática necesarias para poder sacar un buen provecho y ser eficiente en sus actividades.

Aun cuando los conceptos informáticos no son complejos, muchos de ellos tienen orígenes extranjeros sin contar todos los acrónimos que abundan en el argot de la informática. Es necesario, para todo alumno o persona que pretenda alcanzar un conocimiento especializado, tener las bases necesarias para poder avanzar en su aprendizaje, existen hoy en día enciclopedias de informática pero son demasiado extensas y poco accesibles en precio.

Esta tesis tiene como objetivo probar que la implementación de un sistema informático para la aplicación de las evaluaciones del desempeño de los profesores es más eficaz y eficiente que el sistema manual. Es difícil plantear suposiciones sobre actividades no realizadas, pero lo es más plantar suposiciones de actividades ya establecidas y que cumplen con su fin, por ello es que este proyecto no solo debe cumplir con lo provisto con el actual sistema, sino mejorarlo.

La hipótesis que planteo en esta tesis es la siguiente: La implementación de un sistema informático para la aplicación de las evaluaciones del desempeño de los profesores es más eficaz y eficiente que el sistema manual.

La metodología utilizada para la obtención de información fue la entrevista directa con los encargados de elaborar, aplicar y analizar las evaluaciones del desempeño de los



profesores. Se hizo también una investigación documental para conocer las evaluaciones, el puntaje y la forma en que se aplicaban las evaluaciones. También se examinaron las técnicas de obtención de resultados y la valoración de las respuestas.

En esta tesis incluyo además del capítulo de los antecedentes de informática donde desarrollo los puntos principales de la historia de la computación y las diferentes generaciones existentes. Un capítulo de análisis y diseño de sistemas en el cual se plantea la importancia de estos aspectos. Un capítulo de ingeniería de software, sus bases y paradigmas, además de un abrevé descripción de cada uno, una comparativa entre ellos.

Un capítulo de lenguajes de programación donde se muestro las principales características de los diferentes paradigmas de programación existentes hasta hoy en día. El capítulo de estudio de caso presenta el desarrollo de proyecto de tesis, y todos los pasos llevados a cabo. Finalmente las conclusiones obtenidas durante el desarrollo y como resultado de la tesis.

# Capítulo I

## Informática

---

El auge de las computadoras la ha hecho una herramienta indispensable hoy en día para realizar casi cualquier tarea. Acostumbrados a las nuevas tecnologías la mayoría de las personas no consideran el gran avance tecnológico que supone el poder contar con ellas y lo increíble de su funcionamiento basado tan sólo en dos valores para realizar todas sus operaciones. Ningún conocimiento puede ser tan completo como aquel que considera los cimientos del tema en cuestión, es por ello que considero importante hacer mención de aquellos hombres que con su visión del futuro sentaron las bases para la construcción de las computadoras que tenemos hoy en día.

En este capítulo expongo un compendio de cómo se dio el desarrollo de la computadora, las generaciones que han existido y los principales componentes que la conforman. Basándome en la Enciclopedia Didáctica de Computación y en la Encarta 2000 logré reunir las fechas y sucesos más importantes en la historia de cómo las ideas de calcular de manera más rápida pasaron del ábaco a los modernos sistemas electrónicos que realizan millones de operaciones por segundo hoy en día.

El ábaco, quizá instrumento más antiguo construido por el hombre para realizar cálculos, se ha determinado su existencia desde el año 3000 a.C. En la antigua Roma estaban hechos de cera cubierta con arena o una tabla con surcos. Los Mongoles, a finales de la edad media llevaron el ábaco a Rusia, el cual provenía de Tartaria y China, aún se utiliza hoy en día en los pequeños comercios.

## 1.1 Antecedentes

- 1923. La primera máquina de calcular fue concebida por el alemán Wilhelm Schickard (1592-1635), 20 años Más tarde el filósofo francés Blaise Pascal (1623-1662) construyó una máquina que realizaba sumas y restas, a esta máquina le llamó la “Pascalina” la cual tenía el tamaño de una caja de zapatos y funcionaba con engranes de manera semejante a la que trabajan los odómetros en los autos hoy en día. Pascal la construyo para realizar más rápido los cálculos en la recaudación de impuestos, en la que trabajaba su padre (Enciclopedia Didáctica dela computación,1999:18).
- 1801. Joseph Marie Jacquard (1752-1834) de Francia diseña y construye un telar que realizaba su tarea de manera automática siguiendo instrucciones. Jacquard creó la máquina con la posibilidad de cambiarle las instrucciones, las cuales se le daban en forma de tarjetas perforadas, para que creara distintos tipos de tejidos o dibujos en las telas. Joseph Marie sienta las bases de un modelo de automatización, la codificación de la información y la programación de instrucciones que hacen posible que las máquinas realicen lo que se les pide (Ibíd.:23).
- 1811. El ingeniero inglés Charles Babbage (1791-1871), que es considerado el padre de la computación, hace una máquina para realizar cálculos llamada la “máquina de inferencias” la cual calculaba polinomios de segundo grado. Luego diseñó la “máquina analítica” la cual no pudo construir debido principalmente a la falta de recursos económicos, esta “máquina analítica” esboza lo que serían las computadoras en el futuro pues sería capaz de solucionar distinto tipo de problemas, tenía dispositivos de entrada y salida, una unidad de control, memoria y una unidad para realizar las operaciones conocida hoy en día como la

Unidad Lógica-Aritmética. Babbage fue el primero en pensar que las computadoras deberían reunir tanto lo físico como lo lógico para realizar sus funciones (Ibid.:25,26,27).

- 1840. Ada Augusta Byron (1815-1852) fue una colaboradora de Babbage e intervino en la publicación de sus ideas y proyectos. A raíz de sus exposiciones, un ingeniero Militar italiano escribió un artículo sobre la “máquina analítica” de Babbage. El cual fue traducido a Francés y luego al Inglés. Se considera a Ada Byron la primer mujer programadora ya que una de sus tareas como colaboradora de Babbage fue la de crear las primeras instrucciones para que funcionara la “máquina analítica”. En su honor, su nombre fue dado a un lenguaje de programación de alto nivel creado por el ejército de los estados Unidos el lenguaje ADA (Ibid.: 30).
- 1879. Herman Hollerith crea una máquina tabuladora para realizar el censo de los Estados Unidos, más que nada existía la necesidad de acelerar los procesos de conteo porque el censo de la década anterior había tardado tanto tiempo que apenas se había terminado antes de empezar el nuevo. La máquina de Hollerith era capaz de procesar más de 500 tarjetas con información por minuto, reduciendo en un 60% el tiempo que se tardaba en realizarlas con los antiguos métodos manuales (Ibid. :32, 33).
- 1939. Es Diseñada y construida la computadora MARK I, la cual fue instalada en Harvard en 1944. Sus creadores fueron Howard Aiken, Ecker y Mauchly, los cuales se basaron en el “analizador diferencial” de Bush. Y recibieron el apoyo económico de la IBM. El nombre del proyecto de desarrollo de la MARK I era ASCC (Automatic Sequence Controller Calculator) o Calculador Automático de Secuencia Controlada. La MARK I utilizaba el sistema decimal para realizar los cálculos, era capaz de sumar o restar

5 veces por segundo y multiplicaba cantidades de 11 números en 2 segundos. El dividir le tomaba el doble del tiempo que multiplicar (Ibíd.:39, 40).

- 1943. Es construida la ENIAC (Electronic Numerical Integrator and Computer) por Presper Eckert y John Mauchly la computadora más potente hasta entonces, trabajaba con bulbos electrónicos, pesaba 30 toneladas, utilizaba más 17,000 bulbos de 16 tipos diferentes y necesitaba de 100 a 140 Kilovatios para funcionar. Para darnos una idea ese consumo es el equivalente a la energía eléctrica que gastan unas mil viviendas. Los bulbos generaban tanto calor que era necesario refrigerarla con aire acondicionado, tenía 70,000 resistencias, 7500 interruptores y unas 500,000 conexiones según comenta Gregg Keiser en su artículo, donde la llama con toda razón "la madre de todas las computadoras". Esta computadora podía manejar instrucciones lógicas y utilizaba acumuladores como memoria. Se utilizó por el ejército en el cálculo de trayectorias de misiles balísticos, previsión climática, diseño de túneles de viento y para el desarrollo de los planos de la bomba de hidrógeno. Era 100 veces más rápida que la MARK-1 y trabajaba a 100. 000 Hz (Ibíd.:40,41,42). Como referencia la ENIAC necesitaba la misma cantidad de energía que 1000 casas.

## **1.2 Las generaciones de las computadoras**

Lamentablemente es necesario reconocer que es durante los conflictos bélicos cuando se dan los avances tecnológicos de manera más rápida y eficiente debido a la presión por ganar las batallas y superar al enemigo en todos los aspectos.

Tenemos un claro ejemplo de esto en la aviación, comunicación, navegación, etc. Así pues las computadoras no son la excepción y gran parte de su desarrollo inicial fue patrocinado con fines militares.

La segunda guerra creó una necesidad por información más rápida y precisa. En los Estados Unidos, la ENIAC fue construida para optimizar y agilizar los cálculos de trayectorias de misiles. En 1943 Inglaterra utilizó la computadora digital programable llamada Colossus I para descifrar los mensajes en clave que se interceptaban del enemigo, los cuales a su vez eran codificados por una computadora llamada Enigma que utilizaban los alemanes, y aunque la Colossus no tenía más que el propósito de descifrado, bien se puede clasificar como una computadora (Ibid.:47, 48), (Enciclopedia Britannica en línea: <http://ftp.arl.mil/~mike/comphist/46eniac-report/> 28 de Febrero de 2000).

### **1.2.1 Primera Generación: Las computadoras de tubos de vacío o bulbos (1951-1958)**

Algunas de las características de esta primera generación de computadoras son las siguientes:

- Los bulbos o tubos al vacío
- El calor, el cual era producido por los miles de bulbos utilizados por la computadora, era necesario tener un sistema de refrigeración constante mientras la computadora funcionara, normalmente se empleaban grandes ventiladores
- Gran consumo de energía.
- La información de entrada y salida por medio de tarjetas perforadas.
- Las instrucciones eran dadas en códigos que sólo podía entender la máquina, es decir se utilizaba el lenguaje máquina, el cual era muy difícil de comprender a menos que se

tuviera experiencia en el manejo(Enciclopedia Didáctica de la computación,1999: 49-52).

### **1.2.2 Segunda Generación: El transistor (1958-1964)**

Algunos aspectos que caracterizan a esta generación son:

- Se utiliza el transistor para su funcionamiento.
- El tamaño de las computadoras se reduce.
- Hay menos errores en los cálculos y son más rápidas
- El almacenamiento y salida de datos se da de forma magnética en cintas, discos y tambores. Impresoras más rápidas y eficientes
- Surgen propiamente los lenguajes de programación, que son más fáciles de entender y utilizar. El ensamblador es uno de ellos.

FORTRAN, COBOL y ALGOL son algunos lenguajes de programación que se utilizaron, FORTRAN se aplicaba en problemas científicos, COBOL en el ambiente comercial y ALGOL tenía ventajas que lo hacían aplicable a esas dos ramas. Eran los primeros lenguajes de alto nivel (Ibid.:52-55).

### **1.2.3 Tercera Generación. Los circuitos Integrados (1965-1970)**

Algunas de sus características son:

- La invención del Circuito Integrado.
- Reducción del consumo de energía.
- Reducción de las dimensiones de los componentes.

- Se fabrican computadoras compatibles, lo que permite compartir información y programas.
- Se conectan las primeras redes de computadoras, surge el tiempo compartido y el teleproceso además de nuevos periféricos.
- Los lenguajes de alto nivel como COBOL y FORTRAN alcanzan su máxima popularidad y los lenguajes máquina se dejan a un lado. (Ibíd.: 55-57), (SILBERSCHATZ, 1994: 18-22).

#### **1.2.4 Cuarta generación: Microprocesador (1971-1980)**

Los elementos importantes de esta generación son:

- El microprocesador, continúa la reducción en el tamaño de los componentes.
- Se comienzan a utilizar programas para el manejo de las bases de datos.
- Las aplicaciones abarcan a casi todas las áreas del conocimiento: medicina, matemáticas, ingeniería, administración, etc.

El microprocesador es comenzado a fabricarse por INTEL en 1971 bajo pedido de una compañía japonesa que se había dado cuenta del enorme potencial del mercado de las computadoras ( Enciclopedia Britannica en línea: [www.britannica.com/bcom/eb/article/2/0,5716.2242+1,00.html](http://www.britannica.com/bcom/eb/article/2/0,5716.2242+1,00.html) 28/01/2000). Los discos para almacenar la información incrementan su capacidad.

En esta generación es cuando las computadoras comienzan a invadir todos los campos en los que se les puede dar aplicación (Enciclopedia Didáctica de la computación, 1999:57-61).



### **1.2.5 Quinta Generación: Inteligencia Artificial (1982-1999)**

La quinta generación engloba a elementos como:

- Interfaz gráfica, facilita el manejo de la computadora.
- Inteligencia Artificial: Decisiones automáticas.
- Multimedia: Información que contiene imágenes, video y sonidos.
- Robots: Automatas que realizan las tareas más complicadas, tediosas y peligrosas.
- Sistemas expertos: Que tienen la capacidad de dar respuestas empleando los conocimientos de un experto humano (Enciclopedia Britannica en línea: [www.britannica.com/bcom/eb/article/0/0,5716,1640+1.00.html](http://www.britannica.com/bcom/eb/article/0/0,5716,1640+1.00.html) 28/01/2000).
- Lenguaje Natural. Para comunicarse en lenguaje común a la computadora.
- Realidad Virtual: Un entorno simulado por la computadora (Enciclopedia Didáctica de la computación, 1999:63-74).

### **1.3 Componentes de la computadora**

Una vez que hemos dado un vistazo general a la historia de las computadoras y como fue su evolución hasta nuestros días, es necesario entonces estudiarlas internamente para conocer sus componentes y su funcionamiento.

Veamos las definiciones de algunos autores:

Según June Jamrich Parson en su libro conceptos de la computación, “una computadora es un dispositivo que acepta entradas, procesa datos, los almacena y genera salidas.” (JAMRICH, 1997:4).

Para Alan Freedman, una computadora puede definirse como una máquina de propósito general que procesa los datos de acuerdo al conjunto de instrucciones que están almacenadas de manera interna, ya sea temporal o permanentemente. La computadora y todo el equipo conectado a ella se denomina hardware. Las instrucciones que le dicen lo que tiene que hacer se llaman software o programas (Diccionario de Computación, 1994).

Microsoft establece a la computadora como un dispositivo electrónico capaz de recibir un conjunto de instrucciones y ejecutarlas realizando cálculos sobre los datos numéricos, o bien compilando y correlacionando otros tipos de información (Encarta,2000).

Una máquina básica para tratar de forma automática la información, formada por una unidad central de proceso, memorias y unidades de entrada / salida de datos. Es otra definición que podremos encontrar en la enciclopedia Didáctica de la Computación (Enciclopedia Didáctica de la Computación,1999:301)

Ahora haciendo una compilación de dichos autores, mi definición de computadora, a la cual trataré de dar significado como un sistema de cómputo, es la siguiente: Una máquina capaz de procesar información, para lo cual maneja datos de entrada, almacenamiento de datos, procesamiento de datos y es capaz de dar una salida como resultado, por ello se considera un sistema.

En general las computadoras tienen una clasificación basada en la naturaleza de los componentes que los divide en:

- a) Hardware: Dispositivos Físicos (Unidad Central de proceso, memoria, Dispositivos de Entrada y Salida)
- b) Software: Instrucciones o programas que permiten dar órdenes a la computadora.

Haciendo una analogía de la computadora con el cuerpo humano (ver cuadro 1), yo propongo que podría decirse que el hardware son todos los huesos, músculos y tendones que nos permiten movernos; los dispositivos de entrada son todos nuestros sentidos que nos permiten conocer nuestro medio ambiente; los dispositivos de salida son los que nos permiten comunicar lo que pensamos, como nuestra boca.

El software son nuestros pensamientos e ideas que hacen que nuestros movimientos tengan un sentido o un objetivo.

Analogía de la Computadora y el Cuerpo humano

| Computadora             | Cuerpo Humano                      |
|-------------------------|------------------------------------|
| <u>Hardware</u>         | <u>Huesos, Músculos y tendones</u> |
| Dispositivos de entrada | Vista, tacto, gusto, oído, olfato. |
| Dispositivo de salida   | Boca                               |
| <u>Software</u>         | <u>Pensamientos e Ideas</u>        |
| Sistema Operativo       | Instintos básicos                  |

Cuadro 1

Fuente: Apreciación personal

### 1.3.1 Hardware

El hardware, (en inglés significa Materia dura), se puede definir como todo aquello que se puede tocar físicamente de una computadora y se refiere a todos los componentes materiales en el sistema. Las funciones que desempeña el hardware se pueden clasificar de acuerdo a la acción que realiza (ENCARTA,2000).

- Unidad Central de Proceso (CPU Central Process Unit)
- Dispositivos de entrada: Reciben datos o instrucciones para ser procesados en la computadora, entre los más conocidos se encuentran: teclado, ratón, escáner o digitalizador, micrófono, CD-ROM con 650 MB de capacidad, lectores de código de barras y últimamente el DVD el cual si se configuran dos capas por ambos lados la capacidad máxima de almacenamiento es de 17GB (8.5 de cada lado)(PCMAGAZINE Vol.10 No.5, 1999:32).
- Dispositivos de Salida: Son los resultados que arroja una computadora generados por sus operaciones. Permiten que el usuario puede recibir la información de la computadora. Algunos dispositivos de salida son: monitor, bocinas, cañón o proyector, impresoras y Plotters.
- Dispositivos de almacenamiento: Se encargan de guardar la información en la computadora para su uso posterior.
- Dispositivos de entrada y salida: Estos dispositivos reciben y envían información fuera de la computadora, algunos ejemplos son: Disquetes, CD-R, CD-RW, DVD-RAM, módem, disco duro, unidades de respaldo como cintas magnéticas.

El procesador puede ser considerado como el cerebro de la computadora, está formado por miles de microcircuitos (CHIPS) que pueden realizar operaciones con los datos que se le envían.

Cada uno de los dispositivos del hardware se conecta con el procesador de la computadora mediante una línea llamada bus de sistema, de esta manera el procesador recibe todas las entradas y provee todas las salidas.

### **1.3.2 Software**

El software (Materia suave en inglés) es aquello que no se puede tocar físicamente en la computadora, es decir los programas que se ejecutan en ella. Un programa es un conjunto de instrucciones que se le dan a la computadora para que ejecute una determinada acción, por ejemplo hay instrucciones para que la computadora reciba los datos del teclado, para que la computadora muestre la información en el monitor, etc. En cierta forma el software permite que se pueda utilizar el hardware traduciendo nuestras instrucciones a su "lenguaje".

El término software, también conocido como programa, es un término con una aplicación muy amplia y se puede referir tanto a un sólo programa como a varios programas. También se puede referir a aplicaciones que en realidad pueden integrar más de un sólo programa. Existen diferentes tipos de programas orientados cada uno a un área diferente, los más comunes en una computadora son: (Diccionario DYSON de computación, 1995)

### **1.3.2.1 Sistemas Operativos**

También conocidos como DOS (Disk Operating System).A. Silberschatz define a los sistemas operativos como Programas encargados de controlar y coordinar el uso del hardware entre los otros programas de aplicación de los distintos usuarios. “Un sistema operativo es similar a un gobierno, los componentes de un sistema de computación son el hardware, software y los datos.

El sistema operativo proporciona los medios para usar adecuadamente estos recursos durante el funcionamiento del sistema y, como un gobierno, no lleva a cabo ninguna función útil por si mismo, sólo proporciona un entorno dentro del cual otros programas pueden realizar trabajo útil” (SILBERSCHATZ,1994:5) Algunos de los sistemas operativos más comunes son: MS-DOS, UNIX, Windows, OS/2, Windows NT, VMS y MacOS (SILBERSCHATZ,1994:555,651), (PCMEDIA,1999 Año 5 No.10:16-21).

### **1.3.2.2 Programas de aplicación**

Programa de computación diseñado para realizar una tarea específica, tal como contabilidad, análisis científico, procesamiento de texto o diseño gráfico. Los programas de aplicación aquellos que utilizamos para realizar nuestro trabajo en la computadora. Como son: Microsoft Office, Microsoft Encarta 2000, Microsoft Project, Contpaq, SUA2000, Nominpaq, etc.

### **1.3.2.3 Utilerías**

Programas especiales que sirven para dar mantenimiento a otros programas, respaldar información, manejo de archivos, etc. Algunos de los programas más utilizados para este tipo de actividades es el Norton Utilities de Symantec, pero con el auge del Internet y la expansión de las comunicaciones han surgido nuevas necesidades y problemas, como lo es la compresión de archivos , la descarga archivos del Internet en varias sesiones, etc.

### **1.3.2.4 Lenguajes de programación**

Son programas utilizados para crear otros programas. Existen hoy en día más de 200, uno de los más populares es el lenguaje C, Los lenguajes de programación resultan muy convenientes cuando no existen programas de aplicación que cumplan nuestras necesidades (Enciclopedia Britannica en línea: [www.britannica.com/bcom/eb/article/8/0,5716,117728+5,00.html](http://www.britannica.com/bcom/eb/article/8/0,5716,117728+5,00.html) 29/01/2000).

- FORTRAN: Iniciado en 1954 y terminado en 1957, el FORTRAN (Formula Translator) fue el primer lenguaje de alto nivel que podía utilizarse con distintos fines.
- ALGOL: En 1958 surge el ALGOL (Algorithmic Language), otro lenguaje orientado a las necesidades de la ciencia.
- COBOL (Common Business Oriented Language) es un lenguaje de programación para uso comercial y empresarial especializado en la organización de datos y manipulación de archivos, y hoy día está muy difundido en el mundo empresarial.
- BASIC: El lenguaje BASIC (Código de Instrucciones Simbólicas de Uso General para Principiantes) desarrollado en el colegio Dartmouth a principios de la década

de 1960, está dirigido a los usuarios no profesionales (Enciclopedia Didáctica de la computación, 1999:299), (Greg Perry, 1999:8-11).

- PASCAL, diseñado en un principio como herramienta de enseñanza, hoy es uno de los lenguajes de computadora más populares.
- DELPHI: Utiliza un Entorno de Desarrollo Integrado, lo cual permite tener acceso al editor, compilador y demás Utilerías en una misma aplicación (PCMEDIA, 1998 año IV No.1:53).
- JAVA: Un lenguaje de programación que utiliza una “máquina virtual” la cual le permite ejecutarse en cualquier plataforma debido a que esta traduce el código fuente al ejecutarlo. Es ampliamente utilizado en las aplicaciones de Internet y fue creado para ser utilizado en aparatos electrónicos (LEMAY, 1997:1).
- El LISP y el PROLOG han alcanzado amplia difusión en el campo de la inteligencia artificial.
- Lenguaje C: Es de uso general desarrollado por Dennis Ritchie, en los Laboratorios Bell de American Telephone and Telegraph (AT&T), durante los primeros años de la década de 1970. El lenguaje C se originó como el lenguaje primario del sistema operativo UNIX, pero se utilizó ampliamente después de su estandarización por el comité ANSI X3J11 en 1980, y es actualmente el lenguaje de elección para el software comercial en casi todos los sistemas de cómputo (SCHILDT, 1996:3).

En este primer capítulo he querido presentar de manera muy general los antecedentes de la informática, la evolución de las computadoras y el desarrollo de sus componentes internos



así como de los lenguajes de programación más utilizados para crear los programas. Todos estos elementos nos ayudan a entender como es que la computadora se ha convertido poco a poco en una herramienta indispensable para realizar el trabajo del hombre, automatizando las tareas y dando mayor exactitud y rapidez a los cálculos.

## Capítulo II

---

### Ciclo de vida de desarrollo de los sistemas

“Conforme prolifera la información, es esencial un enfoque planeado y sistemático para la introducción, modificación y mantenimiento de los sistemas de información.”

(Kenneth E. Kendall)

Las necesidades de nuestros días son por muchos motivos más complejas: involucran más cantidad de datos, más fuentes de información, mayores distancias de la fuente a la base de datos, mejores interfaces, más reportes especializados, mayor facilidad, etc. El papel del analista en sistemas es muy importante pues identifica las oportunidades para implementar un sistema de información o para mejorar el existente. Además de que le marca los pasos para desarrollar completamente un sistema y las herramientas que le ayudarán a realizarlo. La diferencia entre utilizar y no el Análisis y Diseño de sistemas es obtener un buen sistema que funcione correctamente y cumpla con las especificaciones que existen de manera estándar en el mundo de la informática.

Para la realización del sistema, el diseño y análisis es muy importante pues me permite dirigir el programa hacia los principales puntos del sistema, además de que me da las herramientas y pauta a seguir para el desarrollo del programa. Los conceptos que a continuación se presentan fueron tomados del libro de Análisis y Diseño de Sistemas de Kenneth E. Kendall (KENDALL,1998:2).

## 2.1 Definición e importancia

Según Kenneth E., la información es uno de los elementos más importantes dentro de cualquier organización, por esto debe tenerse singular cuidado al obtenerla, almacenarla y manejarla porque de su buen uso y aplicación dependen una buena toma de decisiones y un éxito por consecuente.

La facilidad de uso y adquisición de las computadoras unidas a su rápida expansión ha permitido que el manejo de la información se haya incrementado drásticamente.

Generalmente los datos generados y utilizados por una organización son administrados por una base de datos la cual se encarga de captar, organizar, mantener y controlar los datos. Junto con la Base de datos, muchos otros sistemas trabajan para que se lleven a cabo las tareas:

- Sistema de procesamiento de transacciones: Maneja aquellas transacciones generadas que contienen grandes volúmenes de datos.
- Sistema de automatización: Programas que contienen todas las herramientas que generalmente son necesarias en una oficina para realizar las labores cotidianas, como son: procesadores de palabras, hojas de cálculo, agendas electrónicas, etc.
- Sistemas de trabajo para el conocimiento: Ayudan a los expertos en sus investigaciones en busca de nuevos conocimientos
- Sistema de apoyo para decisiones: Contiene datos de expertos que han llevado a cabo ciertas tareas y sus resultados, los cuales sirven para tomar decisiones que se basen en elementos similares.

- **Sistemas de Información Gerencial:** Se encargan de recuperar información de toda la organización para que se puedan tomar decisiones tomando en cuenta todos los elementos posibles y así tener mayor eficiencia en la planeación (Ibíd.: 2).

Para Kenneth E. el analista de sistemas debe desempeñarse en diferentes actividades durante sus actividades, ya sea como consultor externo, experto de soporte dentro del negocio y como un agente de cambio en situaciones tanto externas como internas de la organización. Es por ello que debe contar con las habilidades requeridas para solucionar problemas, tomar retos, analizar y encontrar respuestas, comunicarse con los demás y saber llevar su trabajo a las computadoras.

Para poder llevar a cabo el diseño es necesario seguir un conjunto de pasos de manera sistemática, a estos pasos se les llama ciclo de vida del desarrollo de los sistemas. El cual se divide en 7 actividades o secuencias, no todos los analistas están de acuerdo con el número de etapas en el ciclo de vida, pero en general concuerdan en que un enfoque de este tipo hace mucho más fácil el desarrollo de cualquier sistema. Podría pensarse que cada fase se ejecuta de manera independiente y secuencial, más sin embargo es posible que varias tareas se lleven a cabo al mismo tiempo y de manera repetida. Se puede ver al ciclo de vida de los sistemas como una receta que hay que consultar a la hora de realizar un sistema, de tal forma que todo marche bien y que al terminar no nos percatemos que lo que hemos realizado no es realmente efectivo o tenga nula utilidad debido a problemas que pueden ir desde datos no tomados en cuenta hasta información innecesaria.

## **2.2 Ciclo de vida de los sistemas propuesto por Kenneth E.**

### **2.2.1 Identificación de los problemas**

Esta es la primera etapa en el desarrollo de los sistemas y consiste básicamente en reunir todos los datos necesarios para determinar que necesidades existen en un sistema. La recaudación de datos se puede dar de distintas maneras, se puede utilizar:

- Entrevistas
- Cuestionarios
- Observación

Es necesario que el encargado de realizar el análisis sea una persona íntegra y capte la información de manera objetiva y verás para que los resultados de los datos sean significativos.

Una vez recabada toda la información que se considera pertinente, debe analizarse para buscar posibles problemas, los cuales pueden ir desde el retraso en la ejecución de una actividad hasta la existencia de actividades repetidas que pueden integrarse en una sola.

En este primer paso del sistema, es necesario involucrar en el análisis tanto a los usuarios como a los administradores de la organización. Hay que estudiar el tipo de información, si existe, las fuentes y cual es su aplicación dentro de la organización, para así conocer el alcance y la magnitud del sistema a proyectar (Ibid.:9).

### **2.2.2 Oportunidades y Objetivos**

El identificar los objetivos es muy importante pues de esto depende el que concentremos nuestros esfuerzos hacia una dirección y podamos definir las metas del sistema. Se pueden conocer los objetivos al estudiar la organización, entender sus metas y

así desarrollar un sistema encaminado a lograrlas. El analista de sistemas debe determinar la información requerida así como sus fuentes posibles (Ibíd.:8).

Se les llama oportunidades a aquellos aspectos que pueden ser mejorados gracias a un mejor sistema de información, es responsabilidad de los analistas encontrar cuales son estas oportunidades basándose en el análisis de los datos.

Una vez que se determinan los objetivos y las oportunidades, se puede decir que se tienen la definición del problema y la sumarización de los objetivos, los cuales se enunciarán en el estudio de factibilidad el cual está formado por:

- Factibilidad Técnica: Esta estudio nos dice si la organización tiene los recursos técnicos para el desarrollo y utilización del sistema, esta factibilidad se refiere a la conveniencia de que la organización invierta en un nuevo sistema, y si el gasto resultará en mayor eficiencia.
- Factibilidad económica: Determina si se cuentan con los recursos monetarios necesarios para desarrollar y utilizar el sistema propuesto.
- Factibilidad operativa: Nos dice si es posible que el sistema sea operado en la organización, esta factibilidad generalmente está dada por la capacidad del personal para manejar el sistema. Un ejemplo donde no se tiene factibilidad operativa es cuando los usuarios se encuentran renuentes a cambiar de sistema porque están acostumbrados al que ya está implantado o no tienen ni pueden o quieren adquirir los conocimientos necesarios para manejar el sistema propuesto.

En este punto los analistas tienen la información necesaria respecto al costo y dificultades para decidir si continúan con el proyecto o no. Incluso, en caso de que no se cuenten con los fondos necesarios, pueden darse soluciones alternativas para solucionar el problema de forma manual cuando esto sea posible. Se tiene una estimación de la duración total del proyecto y de cada una las fases.

### **2.2.3 determinación de los requerimientos de información**

Kenneth Establece que una vez que se tienen todos los elementos para poder continuar con el desarrollo del sistema, entonces se pasa a la determinación de las necesidades de información por parte de los usuarios, para lograr esto se echará mano de:

- Muestreo e investigación de los datos considerados indispensables, como pueden ser reportes y búsqueda de errores en la información.
- Entrevistas y cuestionarios con los usuarios para determinar sus necesidades específicas y generales.
- Comportamiento de la toma de decisiones, cultura organizacional, etc.

Esta es una de las etapas críticas en el ciclo de los sistemas, debe ser realizada con la cooperación constante de los usuarios pues su ayuda es indispensable tomando en consideración que es el analista el que debe ponderar que información es necesaria y cual incluso irrelevante. Se debe tener conocimientos precisos acerca de:

- El personal que está relacionado con la información (Quién) Nos daremos cuenta de la disposición del personal en la empresa para aceptar un sistema nuevo, los conocimientos, la dispersión y cuales son los usuarios de la información  
 ¿Quién utiliza este reporte?  
 ¿Quién introduce estos datos?  
 ¿Quién autoriza este envío?
- La operación de la organización (Qué) Nos dice cuales son las metas de la organización y la importancia inherente al tipo de datos que se maneja, por ejemplo no es lo mismo manejar la seguridad en un negocio comercial que en un sistema bancario.  
 ¿Qué datos utilizan en la factura?  
 ¿Qué cantidad de información se maneja?  
 ¿Qué es lo que pasa cuando...?
- El ambiente donde se llevan a cabo las operaciones (Dónde) Indica la diversidad de usuarios, los destinos posibles, cual será el alcance del sistema, que tipo de enlaces son utilizados y el tipo de usuarios.  
 ¿Dónde se llena esta forma?  
 ¿Dónde se revisa esta información?  
 ¿Dónde se requiere este reporte?
- En que momento se realizan o deben ser realizadas (Cuándo) Con ello podemos saber con que frecuencia se realizan ciertas actividades y la importancia de recibir un reporte en un determinado lapso de tiempo o a alguna hora específica del día.  
 ¿Cuándo se necesita este listado?  
 ¿Cuándo introducen estos datos?



¿Cuándo se actualiza esta información?

- La manera en que desarrollan las tareas actualmente (Cómo) Nos ayuda a saber cuales son los métodos empleados para realizar las actividades y en gran parte nos ayudan a comprender las tareas y los datos relacionados con ellas.

¿Cómo verifica que es correcta la información?

¿Cómo se llega a esta conclusión?

¿Cómo se puede saber si hay un retraso?

Es necesario también saber el porqué se utiliza el sistema actual para así tener algunas bases a la hora de desarrollar un sistema nueva, por ejemplo, puede ser que se utilice un sistema sobre la base de MS-DOS debido a que la mayoría de las estaciones de trabajo en la organización sean antiguas y no puedan ejecutar un sistema operativo visual. Por otro lado puede ser que el sistema actual se utilice sólo porque no ha existido ninguno otro, en este caso hay que plantear todo el tipo de mejoras posibles echando mano de lo más eficiente que se tenga.

TIP: Hay que lograr que los futuros usuarios se involucren con el desarrollo del nuevo sistema, con esto obtenemos dos grandes beneficios, el primero es que contamos información de primera fuente, y el otro es que por el simple hecho de tener una relación con el diseño del sistema, los usuarios lo aceptan de buena gana ya que sienten que ellos han contribuido para su desarrollo (Ibíd.: 9).

#### **2.2.4 Análisis de las necesidades del sistema**

En esta etapa se estudian las necesidades del sistema, una de las herramientas para este análisis son los diagramas de flujo, una vez que se realiza este diagrama se pasa a determinar el diccionario de datos en el cual se especifica los significados de cada uno de los datos descritos en el diagrama de flujo, en el diccionario también se especifica que tipo de datos es, si es alfanumérico por ejemplo, y la longitud que ocupa.

También se determinan las alternativas que se toman en el sistema y se traducen en decisiones. En algunos casos las decisiones involucran distintas variables, para estos casos hay distintos métodos, algunos de los más utilizados son:

- Proceso de pros y contras: Consiste en dividir las ventajas de las desventajas, luego se van eliminando aquellas ventajas y desventajas que tengan un mismo valor.
- Métodos ponderados: Es cuando se asigna un determinado porcentaje a cada elemento a considerar, dependiendo de su importancia.
- Eliminación secuencial por lexicografía: Se ordenan las características de acuerdo a su importancia, primero se descartan aquellos que queden al último, los demás se van depurando de acuerdo a la importancia de los diferentes atributos extras.

El analista tiene ahora la posibilidad de hacer un resumen de todo lo que ha analizado, puede hacer un análisis costo beneficio y puede recomendar alguna de las opciones disponibles (Ibíd.: 9).

### 2.2.5 Diseño del sistema recomendado

El diseño lógico del sistema de información es realizado tomando en consideración todos los datos recolectados.

En esta parte se realiza:

- Interfaz de usuario: Que es lo que verá el usuario al trabajar con el sistema, esto es muy importante pues depende de ella que el usuario maneje de buena gana el sistema y de que entienda lo que se le pide, entre otras cosas.

Algunos de los tipos de interfaz que existen son:

- a) Interfaz de Lenguaje natural: Son la interfaz con la que soñamos todos encontrar algún día, es en la que no se requieren habilidades para manejar un sistema, uno sólo le escribe la orden al sistema y este lo procesa, por ejemplo se le escribe: "lista los proveedores a los que les debemos" y el sistema dará un listado con sus nombres.
- b) Interfaz de Pregunta y respuesta: Esta interfaz es cuando el sistema nos presenta una pregunta a la que contestaremos seleccionando alguna de las opciones dadas, entonces el sistema actuará de acuerdo a lo contestado, un ejemplo de esta interfaz podría ser la de alguna instrucción que pregunta antes de ejecutar alguna acción.
- c) Menú: Este tipo de interfaz se usa comúnmente cuando se quiere presentar al usuario un conjunto de opciones disponibles, todos estamos familiarizados con ellos: Abrir, Edición, Ayuda, etc.

- d) Interfaz de Llenado de formas o Formas de entrada / salida: En esta interfaz tenemos en la pantalla algo parecido a un formulario donde hay casillas o cuadros de texto donde ponemos nuestras respuestas o instrucciones.
- e) Interfaz de Lenguajes de comandos: En ella controlamos la aplicación por medio de comandos o teclas, un ejemplo de este lenguaje puede ser el editor de DBASE con sus instrucciones de manejo de archivos de datos.
- f) Interfaz Gráfica de Usuario (GUI Graphics User Interface): Utiliza un ambiente gráfico que facilita la ejecución de los comandos, un ejemplo de esta interfaz es el sistema operativo Windows 95 y el MAC OS de Apple.
- Diseño de las entradas y salidas del sistema: Para validar la información que entra al sistema para minimizar errores en los cálculos y consistencia en la información. Las salidas se refieren a las pantallas o reportes que aportará el sistema.
  - Diseño de archivos y bases de datos: En estos archivos se guardaran los datos que ocupa el sistema para trabajar, de un buen análisis y diseño de la base de datos dependerá un buen funcionamiento y eficiencia en el sistema.
  - Diseño de procedimientos de control y respaldo: Aquí se establece la seguridad del sistema para evitar que sólo los usuarios autorizados puedan acceder a la información, el respaldo sirve para tener una copia de la información indispensable para disponer de ella en caso de que la información original se dañe o pierda por alguna razón. El analista debe formar paquetes con los módulos del sistema que más tarde serán programados, también debe planear la forma de conjuntar los módulos sin que se repitan funciones (Ibíd.: 10).

### 2.2.6 Desarrollo e implementación del software

El desarrollo implica en sí el trabajo de realizar el programa que soportará al sistema de información. La documentación se refiere a describir todo lo referente al programa que se desarrolla para que en un futuro sea posible entender lo que se hizo. Por ejemplo, en caso de requerirse una modificación se tengan las bases necesarias para saber en cual parte del programa está el módulo que hay que modificar.

En esta etapa es importante que el analista tenga el apoyo de los usuarios para realizar una documentación precisa, y para que los manuales de uso tengan toda la información necesaria para que el sistema pueda ser comprendido y operado. Los programadores son los encargados de traducir las especificaciones hechas por el analista en código de programación para crear el sistema. Algunas de las técnicas estructuradas para el diseño y documentación del software son:

- a) Diagramas estructurados: Se basa en la consecución de pasos llevados a cabo en el sistema.
- b) Diagramas de flujo: basados en el flujo de datos del sistema.
- c) Diagramas Nassi-Schneiderman.
- d) Warnier-Orr.
- e) Método HIPO
- f) Pseudo código: escribir con nuestras propias palabras lo que hace el sistema (Ibíd.: 10).

### 2.2.7 Pruebas y mantenimiento de sistema

Al terminar el desarrollo del sistema debe proseguir una etapa de prueba para confirmar que todo su funcionamiento cumple con las expectativas con las que fue creado y que no presentará problemas más tarde cuando entre en funcionamiento de manera definitiva. Si al llegar a este punto el sistema presenta cualquier tipo de problema es necesario estudiarlo para determinar si es por causa de la programación, de un mal diseño o porque no se tomaron en cuenta en el análisis los aspectos del problema, en este caso será necesario darle la solución requerida, ya sea desde volver a programar hasta volver a hacer el diseño.

Lo más lógico es que cuando uno termina un sistema es porque ha concluido con todas las tareas para echarlo a funcionar. Resulta ridículo pensar que al terminar el sistema será necesario darle un mantenimiento, lamentablemente debido a problemas con el diseño, cambios en los datos o simplemente porque se encontró algo más eficiente para el manejo de información algunas veces se hace un mantenimiento del programa, en realidad el mantenimiento da la idea de que es para corregir errores, pero también se puede dar cuando se incluye una mejora que puede ser por las siguientes causas:

- Solicitud de nuevas características por parte de los usuarios: cuando se requiere un reporte adicional.
- Cambios en la organización y su información. Cuando surge un nuevo departamento.
- Evolución del software o hardware: Cuando se usa un sistema operativo diferente.
- Cambios externos, como el acontecimiento del año 2000 que podrá traer algunas consecuencias en la forma de determinar las fechas. (Ibíd.: 11)

## 2.2.8 Implementación y evaluación del sistema

Una vez que se ha concluido con todo lo referente al desarrollo del sistema, se han quitado los errores y se ha probado su procesamiento y fiabilidad, hay que proceder a instalarlo de modo que realice su función. Este último paso no sólo se refiere a la implementación del nuevo sistema, sino que también abarca la capacitación que se debe dar a los usuarios para que sepan como utilizarlo. También es necesario que el analista tome en cuenta que la migración de un sistema a otro debe ser paulatina para que no existan problemas con la aceptación del nuevo sistema, esta actividad también engloba la utilización de los datos antiguos por el sistema nuevo, lo cual puede implicar transformar los datos a nuevos formatos o a reestructurarlos.

Según Universidad del Tolima, Instituto de Educación a Distancia la implementación de un sistema abarca elementos tales como:

- Niveles de capacitación.
- Métodos de capacitación
- Métodos de conversión de sistemas.
- Componentes de un plan de conversión.
- Revisiones requeridas después de la implantación de un sistema. (Ibid.: 11)

Por lo anterior podemos concluir que aunque es posible realizar un sistema sin seguir los pasos del ciclo de vida, el tomarlos en cuenta nos ahorra muchos dolores de cabeza, tiempo y

dinero, lo cual se traduce en un mejor servicio hacia los usuarios. que es al fin al cabo el fin de cualquier sistema.

Existen algunos programas que ayudan al analista y diseñador de sistemas a realizar estas tareas de manera automática, estos programas son denominados herramientas CASE (Computer Aided Software Engineering), algunas de las ventajas que se obtienen de su aplicación son: más y mejor comunicación entre usuarios y diseñador, integración de los pasos y por lo tanto mayor productividad para el diseñador.

CARE (Reingeniería asistida por computadora) es otra de las herramientas utilizada por los diseñadores para hacer las modificaciones (reingeniería) al software y este siga siendo eficiente satisfaciendo necesidades cambiantes. (Ibíd.: 13)



# Capítulo III

## Capítulo de diseño de sistemas

---

El diseño del sistema establece de manera formal las reglas que se siguen en la realización del sistema, en el diseño recae el que un sistema tenga un desempeño favorable.

En la documentación de los datos se justifica y explica el uso de cada campo de la base de datos. Así como su longitud y tipo de información contenida. En la sección del pseudocódigo se encuentra explicado de manera natural las funciones que realiza de manera interna el sistema en cada módulo para realizar sus operaciones. El diseño de entradas y salidas indica cómo el sistema obtiene la información necesaria y cuales son los resultados que puede arrojar.

### 3.1 Documentación de la base de datos

#### **Tablas de archivo profesor**

Esta base de datos contiene los datos del profesor y los relacionados directamente con este

Cada salón está formado por un semestre y un grupo (Primer semestre A)

Cada carrera tiene un conjunto de semestres y grupos

Cada semestre y grupo de cada carrera tiene un conjunto de materias asignadas

cada una de estas materias es impartida por un profesor

Tabla: profesor

En esta tabla se almacenan los datos del profesor

| Nombre del campo | Descripción                 | Tipo         | Longitud     |
|------------------|-----------------------------|--------------|--------------|
| clave_profesor   | clave principal de profesor | autonumérico | entero largo |
| nombre_profesor  | nombre del profesor         | texto        | 30           |

Tabla: carrera

En esta tabla se almacenan los datos de las carreras disponibles

| Nombre del campo    | Descripción                | Tipo         | Longitud     |
|---------------------|----------------------------|--------------|--------------|
| clave_carrera       | clave principal de carrera | autonumérico | entero largo |
| descripcion_carrera | nombre del profesor        | texto        | 20           |

Tabla: carrera\_semestre

En esta tabla se almacenan los datos de la relación carrera y semestre

| Nombre del campo       | Descripción                         | Tipo         | Longitud     |
|------------------------|-------------------------------------|--------------|--------------|
| clave_carrera_semestre | clave principal de carrera_semestre | autonumérico | entero largo |
| clave_semestre         | clave principal de semestre         | numérico     | entero largo |
| clave_carrera          | clave principal de carrera          | numérico     | entero largo |

Tabla: grupo

En esta tabla se almacenan los datos de los grupos

| Nombre del campo  | Descripción              | Tipo         | Longitud     |
|-------------------|--------------------------|--------------|--------------|
| clave_grupo       | clave principal de grupo | autonumérico | entero largo |
| descripcion_grupo | descripción del grupo    | numérico     | entero largo |

Tabla: materia

En esta tabla se almacenan los datos de las materias disponibles

| Nombre del campo    | Descripción                | Tipo         | Longitud     |
|---------------------|----------------------------|--------------|--------------|
| clave_materia       | clave principal de materia | autonumérico | entero largo |
| descripcion_materia | nombre de la materia       | texto        | 20           |

Tabla: semestre

En esta tabla se almacenan los datos de los semestres

| Nombre del campo     | Descripción                 | Tipo         | Longitud     |
|----------------------|-----------------------------|--------------|--------------|
| clave_semestre       | clave principal de semestre | autonumérico | entero largo |
| descripcion_semestre | numero de semestre          | texto        | 15           |

Tabla: semestre\_grupo

En esta tabla se almacenan los datos de la relación semestre y grupo

| Nombre del campo     | Descripción                       | Tipo         | Longitud     |
|----------------------|-----------------------------------|--------------|--------------|
| clave_semestre_grupo | clave principal de semestre_grupo | autonumérico | entero largo |
| clave_semestre       | clave principal de semestre       | numérico     | entero largo |
| clave_grupo          | clave principal de grupo          | numérico     | entero largo |

Tabla: materia\_carrera

En esta tabla se almacenan los datos de la relacion carrera\_semestre y materia

| Nombre del campo       | Descripción                         | Tipo         | Longitud     |
|------------------------|-------------------------------------|--------------|--------------|
| clave_materia_carrera  | clave principal de materia_carrera  | autonumérico | entero largo |
| clave_materia          | clave principal de materia          | numérico     | entero largo |
| clave_carrera_semestre | clave principal de carrera_semestre | numérico     | entero largo |

Tabla: profesor\_materia

En esta tabla se almacenan los datos de la relación de materia\_carrera y profesor

| Nombre del campo       | Descripción                         | Tipo         | Longitud     |
|------------------------|-------------------------------------|--------------|--------------|
| clave_profesor_materia | clave principal de profesor_materia | autonumérico | entero largo |
| clave_materia_carrera  | clave principal de materia_carrera  | Numérico     | entero largo |
| clave_profesor         | clave principal de profesor         | Numérico     | entero largo |

## Tablas de archivo evaluación

Esta Base de datos contiene los datos de las preguntas

Cada evaluación contiene preguntas, varias evaluaciones pueden tener la misma pregunta

Cada pregunta puede ser

de opción múltiple o

abierta

Cada una de las opciones múltiples tiene una letra o número de identificador, el texto de la opción y el valor

Una pregunta puede aceptar que se responda más de una opción

| Tabla: pregunta                                       |                                     |              |              |
|---|-------------------------------------|--------------|--------------|
| En esta tabla se almacenan los datos de las preguntas |                                     |              |              |
| Nombre del campo                                      | Descripción                         | Tipo         | Longitud     |
| clave_pregunta  | clave principal de pregunta         | autonumérico | entero largo |
| es_abierta  | SI= es abierta, NO= opciones        | texto        | 2            |
| multiple_opcion                                       | NO= una opción, SI= varias opciones | texto        | byte         |
| descripcion_pregunta                                  | texto de la pregunta                | texto        | 150          |

Tabla: opcion\_pregunta

En esta tabla se almacenan las opciones de las preguntas

| Nombre del campo      | Descripción                      | Tipo         | Longitud     |
|-----------------------|----------------------------------|--------------|--------------|
|                       | clave principal de               | autonumérico |              |
| clave_opcion_pregunta | opcion_pregunta                  | rico entero  | largo        |
| clave_pregunta        | clave principal de pregunta      | numérico     | entero largo |
| clave_opcion          | letra o numero de opción         | Texto        | 10           |
| descripcion_opcion    | descripción o texto de la opción | Texto        | 50           |
| valor_opcion          | texto de la pregunta             | Texto        | 75           |

Tabla: pregunta\_evaluación

En esta tabla se almacenan las relaciones de preguntas y

evaluaciones

| Nombre del campo          | Descripción                   | Tipo         | Longitud     |
|---------------------------|-------------------------------|--------------|--------------|
|                           | clave principal de            | Autonumérico |              |
| clave_pregunta_evaluación | pregunta_evaluacion           | érico entero | largo        |
| clave_pregunta            | clave principal de pregunta   | numérico     | entero largo |
| valor_evaluación          | clave principal de evaluación | numérico     | entero largo |

Esta tabla se justifica porque así podemos utilizar las mismas preguntas para distintas evaluaciones como es el caso de la evaluación de los profesores, una es la que está establecida y las respuestas serán asignadas a esa evaluación. Sólo una puede estar establecida a la vez

| Tabla: evaluación   |                                    |              |              |
|---|------------------------------------|--------------|--------------|
| En esta tabla se almacenan los datos de las evaluaciones de cada profesor |                                    |              |              |
| Nombre del campo  | Descripción                        | Tipo         | Longitud     |
| clave_evaluación  | clave principal de evaluación      | autonumérico | entero largo |
| clave_pregunta  | clave principal de pregunta        | numérico     | entero largo |
| fecha_evaluación  | fecha en que se crea la evaluación | texto        | 10           |

## Tablas de archivo de respuestas

Esta Base de datos contiene los datos de las preguntas

Cada evaluación contiene preguntas, varias evaluaciones pueden tener la misma pregunta

Cada pregunta puede ser de opción múltiple o abierta

Cada una de las opciones múltiples tiene una letra o número de identificador, el texto de la opción y el valor

| Tabla: respuesta                                      |  |              |              |
|---|--|--------------|--------------|
| En esta tabla se almacenan los datos de las preguntas |  |              |              |
| Nombre del campo                                      | Descripción                            | Tipo         | Longitud     |
| clave_respuesta                                       | clave principal de respuesta           | autonumérico | entero largo |
| clave_evaluado  | clave principal del que hace el examen | numérico     | entero largo |
| clave_profesor_evaluacion                             | clave principal de profesor_evaluacion | numérico     | entero largo |
| clave_pregunta_evaluacion                             | clave principal de pregunta_evaluacion | numérico     | entero largo |



Tabla: respuesta

En esta tabla se almacenan los datos de las preguntas

| Nombre del campo          | Descripción                            | Tipo         | Longitud     |
|---------------------------|--|--------------|--------------|
| clave_respuesta           | clave principal de respuesta           | autonumérico | entero largo |
| clave_pregunta_evaluacion | clave principal de pregunta_evaluacion | numérico     | entero largo |

Tabla: respuesta\_opcion

En esta tabla se almacenan los datos de las preguntas

| Nombre del campo       | Descripción                         | Tipo         | Longitud     |
|------------------------|-------------------------------------|--------------|--------------|
| clave_respuesta_opcion | clave principal de respuesta_opcion | autonumérico | entero largo |
| clave_respuesta        | clave principal de respuesta        |              |              |
| clave_pregunta_opcion  | clave principal de pregunta_opcion  | numérico     | entero largo |

Esta tabla se justifica pues puede haber preguntas que tengan como respuesta más de una opción

### 3.2 PseudoCódigo

El pseudocódigo es la explicación simple de las funciones que realiza el sistema, está redactado con palabras simples que ayudan al desarrollador a realizar la programación de los módulos del sistema.

#### Módulo de Profesores

Abrir base de datos

Mostrar lista de registros

Cerrar base de datos

Esperar a que el usuario ejecute una acción

Agregar registro

Obtener datos necesarios para agregar el registro

Verificar validez de los datos

Verificar que no exista un registro con los mismos datos

Si son válidos

Abrir base de datos

Agregar registro

Informar que se ha agregado el registro

Cerrar base de datos

Si no son válidos

Informar que hay datos faltantes o inválidos

Eliminar registro

Verificar que un registro a sido seleccionado de la lista

Si está seleccionado algún registro

Obtener clave del registro seleccionado

Mostrar mensaje de confirmación

Si hay confirmación para la eliminación

Abrir base de datos

Eliminar registro de la clave seleccionada

Informar que se ha eliminado el registro

Cerrar base de datos

Si no está seleccionado ningún registro

Informar que no hay ningún registro seleccionado para su eliminación

Modificar registro

Verificar que un registro a sido seleccionado de la lista

Si está seleccionado algún registro

Obtener clave del registro seleccionado

Mostrar mensaje de confirmación

Si hay confirmación para la modificación

Obtener los nuevos datos del registro

Verificar validez de los nuevos datos

Verificar que no exista un registro con los mismos datos

Si son válidos

Abrir base de datos

Guardar las modificaciones al registro de la clave

seleccionada

Informar que se ha modificado el registro

Cerrar base de datos

Si no son válidos

Si no está seleccionado ningún registro

Informar que no hay ningún registro seleccionado para su eliminación

Cerrar módulo de profesores

### **Módulo de materias**

Abrir base de datos

Mostrar lista de registros

Cerrar base de datos

Esperar a que el usuario ejecute una acción

Agregar registro

Obtener datos necesarios para agregar el registro

Verificar validez de los datos

Verificar que no exista un registro con los mismos datos

Si son válidos

Abrir base de datos

Agregar registro

Informar que se ha agregado el registro

Cerrar base de datos

Si no son válidos

Informar que hay datos faltantes o inválidos

## Eliminar registro

Verificar que un registro a sido seleccionado de la lista

Si está seleccionado algún registro

Obtener clave del registro seleccionado

Mostrar mensaje de confirmación

Si hay confirmación para la eliminación

Abrir base de datos

Eliminar registro de la clave seleccionada

Informar que se ha eliminado el registro

Cerrar base de datos

Si no está seleccionado ningún registro

Informar que no hay ningún registro seleccionado para su eliminación

## Modificar registro

Verificar que un registro a sido seleccionado de la lista

Si está seleccionado algún registro

Obtener clave del registro seleccionado

Mostrar mensaje de confirmación

Si hay confirmación para la modificación

Obtener los nuevos datos del registro

Verificar validez de los nuevos datos

Verificar que no exista un registro con los mismos datos

Si son válidos

Abrir base de datos

Guardar las modificaciones al registro de la clave  
seleccionada

Informar que se ha modificado el registro

Cerrar base de datos

Si no son válidos

Si no está seleccionado ningún registro

Informar que no hay ningún registro seleccionado para su eliminación

Cerrar módulo de materias

### **Módulo de grupos**

Abrir base de datos

Mostrar lista de registros

Cerrar base de datos

Esperar a que el usuario ejecute una acción

Agregar registro

Obtener datos necesarios para agregar el registro

Verificar validez de los datos

Verificar que no exista un registro con los mismos datos

Si son válidos

Abrir base de datos

Agregar registro

Informar que se ha agregado el registro

Cerrar base de datos

Si no son válidos

Informar que hay datos faltantes o inválidos

Eliminar registro

Verificar que un registro a sido seleccionado de la lista

Si está seleccionado algún registro

Obtener clave del registro seleccionado

Mostrar mensaje de confirmación

Si hay confirmación para la eliminación

Abrir base de datos

Eliminar registro de la clave seleccionada

Informar que se ha eliminado el registro

Cerrar base de datos

Si no está seleccionado ningún registro

Informar que no hay ningún registro seleccionado para su eliminación

Modificar registro

Verificar que un registro a sido seleccionado de la lista

Si está seleccionado algún registro

Obtener clave del registro seleccionado

Mostrar mensaje de confirmación

Si hay confirmación para la modificación

Obtener los nuevos datos del registro

Verificar validez de los nuevos datos

Verificar que no exista un registro con los mismos datos

Si son válidos

Abrir base de datos

Guardar las modificaciones al registro de la clave  
seleccionada

Informar que se ha modificado el registro

Cerrar base de datos

Si no son válidos

Si no está seleccionado ningún registro

Informar que no hay ningún registro seleccionado para su eliminación

Cerrar módulo de grupos

### **Módulo de semestres**

Abrir base de datos

Mostrar lista de registros

Cerrar base de datos

Esperar a que el usuario ejecute una acción

Agregar registro

Obtener datos necesarios para agregar el registro

Verificar validez de los datos

Verificar que no exista un registro con los mismos datos

Si son válidos

Abrir base de datos

Agregar registro



Informar que se ha agregado el registro

Cerrar base de datos

Si no son válidos

Informar que hay datos faltantes o inválidos

Eliminar registro

Verificar que un registro a sido seleccionado de la lista

Si está seleccionado algún registro

Obtener clave del registro seleccionado

Mostrar mensaje de confirmación

Si hay confirmación para la eliminación

Abrir base de datos

Eliminar registro de la clave seleccionada

Informar que se ha eliminado el registro

Cerrar base de datos

Si no está seleccionado ningún registro

Informar que no hay ningún registro seleccionado para su eliminación

Modificar registro

Verificar que un registro a sido seleccionado de la lista

Si está seleccionado algún registro

Obtener clave del registro seleccionado

Mostrar mensaje de confirmación

Si hay confirmación para la modificación

Obtener los nuevos datos del registro

Verificar validez de los nuevos datos

Verificar que no exista un registro con los mismos datos

Si son válidos

Abrir base de datos

Guardar las modificaciones al registro de la clave  
seleccionada

Informar que se ha modificado el registro

Cerrar base de datos

Si no son válidos

Si no está seleccionado ningún registro

Informar que no hay ningún registro seleccionado para su eliminación

Cerrar módulo de semestres

### **Módulo de carreras**

Abrir módulo de carreras

Abrir base de datos

Mostrar lista de registros

Cerrar base de datos

Esperar a que el usuario ejecute una acción

Agregar registro

Obtener datos necesarios para agregar el registro

Verificar validez de los datos

Verificar que no exista un registro con los mismos datos

Si son válidos

Abrir base de datos

Agregar registro

Informar que se ha agregado el registro

Cerrar base de datos

Si no son válidos

Informar que hay datos faltantes o inválidos

Eliminar registro

Verificar que un registro a sido seleccionado de la lista

Si está seleccionado algún registro

Obtener clave del registro seleccionado

Mostrar mensaje de confirmación

Si hay confirmación para la eliminación

Abrir base de datos

Eliminar registro de la clave seleccionada

Informar que se ha eliminado el registro

Cerrar base de datos

Si no está seleccionado ningún registro

Informar que no hay ningún registro seleccionado para su eliminación

Modificar registro

Verificar que un registro a sido seleccionado de la lista

Si está seleccionado algún registro

Obtener clave del registro seleccionado

Mostrar mensaje de confirmación

Si hay confirmación para la modificación

Obtener los nuevos datos del registro

Verificar validez de los nuevos datos

Verificar que no exista un registro con los mismos datos

Si son válidos

Abrir base de datos

Guardar las modificaciones al registro de la clave  
seleccionada

Informar que se ha modificado el registro

Cerrar base de datos

Si no son válidos

Si no está seleccionado ningún registro

Informar que no hay ningún registro seleccionado para su eliminación

Cerrar módulo de carreras

### **Módulo de asignación de grupos a semestres**

Abrir Módulo de asignación de grupos a semestres

Abrir base de datos

Mostrar lista de semestres

Mostrar lista de grupos

Mostrar lista de grupos y semestres asignados

Cerrar base de datos

Esperar a que suceda un evento

    Seleccionar semestre

        Establecer clave del semestre seleccionado

    Seleccionar grupo

        Establecer clave del semestre seleccionado

    Asignar grupo a semestre

    Validar que estén seleccionados

    Validar que no haya duplicados

    Si están validados

        Abrir base de datos

            Agregar nuevo registro de asignación de grupo y semestre

            Mostrar lista de grupos y semestres asignados

        Cerrar base de datos

        Informar que se agregó la asignación de grupo y semestre

    Si no están validados

        Informar que no están seleccionados los registros necesarios

    Eliminar Asignación

Cerrar Módulo de asignación de grupos a semestres

## **Módulo de asignación de profesores**

Abrir Módulo de asignación de profesores

Abrir base de datos

    Mostrar lista de semestres

Mostrar lista de carreras

Mostrar lista de materias

Esperar a que suceda un evento

    Seleccionar semestre

        Establecer clave del semestre seleccionado

    Seleccionar carreras

        Establecer clave de la carrera seleccionada

        Mostrar lista de semestres asignados a la carrera

    Seleccionar semestres asignados a la carrera

        Establecer clave del semestre asignado a la carrera seleccionado

    Seleccionar materia

        Establecer clave de la materia seleccionada

    Agregar semestre a la carrera

    Eliminar semestre de la carrera

    Asignar materia al semestre de la carrera

    Eliminar materia al semestre de la carrera

Mostrar lista de semestres asignados a la carrera

Mostrar lista de materias asignadas al semestre de la carrera

Cerrar base de datos

Cerrar módulo de asignación de profesores

Como se puede apreciar en el pseudocódigo, cada uno de los módulos se encarga de realizar una tarea específica, de tal forma que cualquier modificación necesaria tenga un menor impacto en el sistema general.

El pseudocódigo es una herramienta muy útil al programar, sobre todo en rutinas complejas donde, por lo complicado de las operaciones, una operación involucra un considerable número de variables o recibe información de múltiples fuentes.

### 3.3 Diseño de entradas:

Búsquedas dentro del sistema:

Este programa cuenta con un sistema de búsquedas muy útil que hacen mucho más fácil el encontrar registros en particular.

En la mayoría de los casos en los que se involucra una búsqueda que contenga más de una palabra, el sistema muestra una lista que presenta los registros coincidentes con los caracteres tecleados, conforme se teclca el texto completo la lista muestra los registros filtrados.

Ejemplo de búsqueda de profesor

1. Se comienza a teclear el nombre del profesor. En la lista aparecen todos los nombres de profesores que comiencen con la primera letra capturada

Nombre Profesor

|   |
|---|
| R   |
| RICARDO ROMERO PÉREZ<br>ROBERTO MORENO P.<br>ROSA ELENA NÚÑEZ |

2. Conforme se teclea el nombre completo del profesor sólo aparecen los registros concuerdan con el texto.

Nombre Profesor

|                                      |
|--------------------------------------|
| RO                                   |
| ROBERTO MORENO P<br>ROSA ELENA NÚÑEZ |

3 Una vez que se visualiza el nombre del profesor buscado, se selecciona de la lista haciendo clic en ella o presionando la tecla de movimiento hacia abajo y presionando [ENTER].

Nombre Profesor

|                  |
|------------------|
| ROBERTO          |
| ROBERTO MORENO P |
|                  |

Esta forma de realizar búsquedas elimina la necesidad de teclear una clave la cual debe memorizarse o escribir un nombre y luego hacer clic en un botón buscar. También hace innecesario tener botones de desplazamiento por los registros para una búsqueda secuencial. Aunque no por esto no se puede realizar, aunque de una manera distinta: al dejar un espacio en blanco, en la lista aparecerán todos los registros en orden alfabético en los cuales se puede desplazar de manera vertical hasta encontrar el deseado

Hace más fácil las actividades del sistema

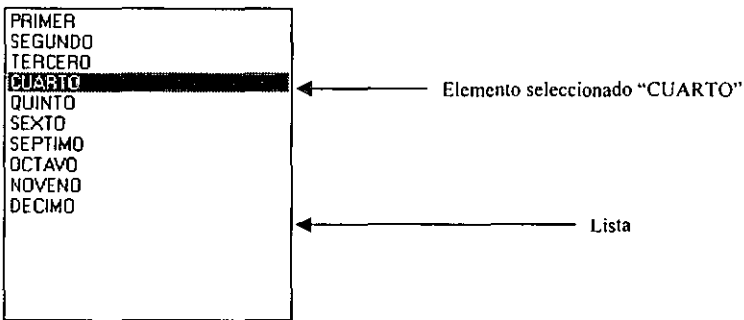
Cabe mencionar que cada cuadro de búsqueda tiene su propia validación y valida que la entrada sea correcta, evitando búsquedas inútiles, por ejemplo al buscar números en campos alfabéticos.



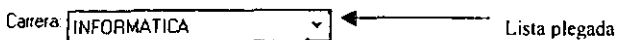
## Selección de datos dentro del sistema

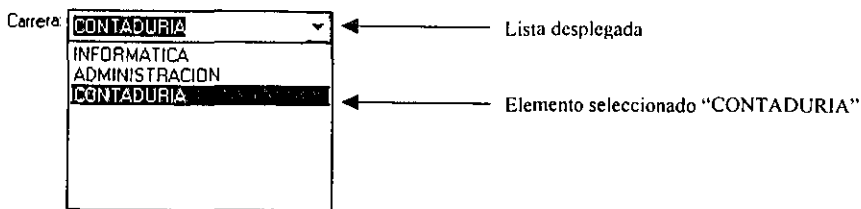
La selección de los datos dentro del sistema cuando provienen de alguna tabla de la base de datos se hace mediante la utilización de listas, combos o tablas de consulta. Esto facilita la selección de los registros y evita la captura errónea de claves o descripciones.

Listas: En ellas aparecen los datos de manera vertical, un dato en cada renglón y generalmente siguiendo un orden. Para seleccionar un elemento basta con hacer clic en él y aparecerá remarcado. Estas listas se manejan cuando hay una gran cantidad de datos que presentar.



Listas desplegables: Al igual que las listas muestran los datos de manera vertical, uno en cada renglón, la diferencia es que están "ocultos" y sólo aparece un elemento, al hacer clic en la flecha de despliegue, aparece una lista con los demás elementos, este tipo de listas se utiliza cuando los datos contenidos en ellas no son numerosos y se quiere ahorrar algo de espacio en la pantalla del programa.





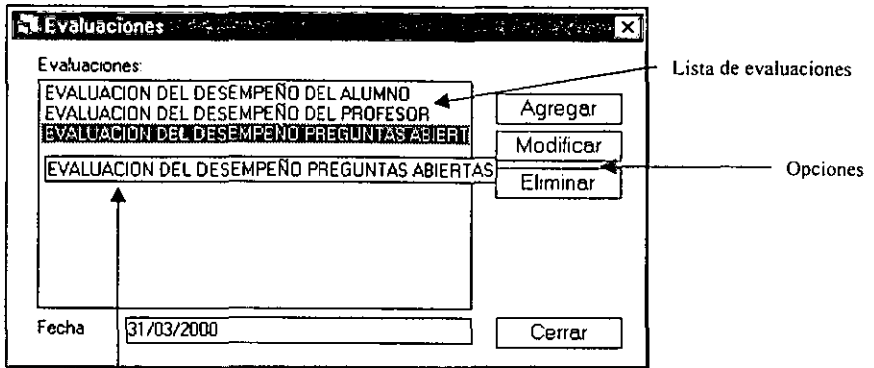
Tablas de consulta: Las tablas de consulta son muy parecidas a las listas normales, pero incluyen más columnas para tener vista a más información, en la figura del ejemplo se tienen dos columnas, en la primera se muestra el semestre y en la segunda el grupo, cada elemento de la lista puede ser seleccionado haciendo clic en él. Este tipo de lista permite seleccionar registros que comparten algunas características, por ejemplo en este caso el semestre y el grupo.

| Semestre | Grupo |
|----------|-------|
| PRIMER   | A     |
| PRIMER   | B     |
| SEGUNDO  | A     |
| SEGUNDO  | B     |
| TERCERO  | A     |
| TERCERO  | B     |
| CUARTO   | A     |
| CUARTO   | B     |
| QUINTO   | B     |
| SEXTO    | A     |
| SEXTO    | B     |
| SEPTIMO  | A     |

### Módulo de evaluaciones

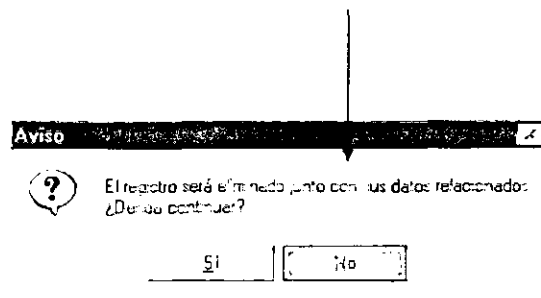
El Módulo de evaluaciones contiene las opciones necesarias para agregar, eliminar y modificar, además muestra una lista de las evaluaciones existentes en el sistema.

Al Agregar un nuevo registro se validan los datos, se valida la longitud de los datos y se pasan a mayúsculas.



Cuando el nombre de una evaluación no puede ser mostrado totalmente en la lista basta con hacer clic en ella para que aparezca una etiqueta con el nombre completo

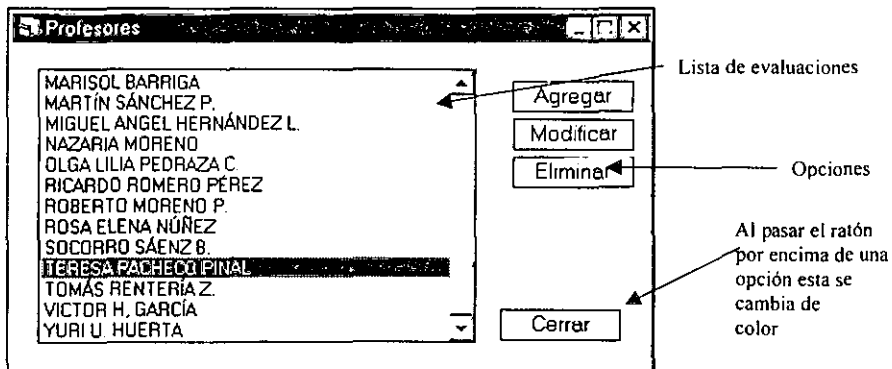
Al realizar una modificación o eliminación aparece un cuadro de diálogo para confirmar la acción y asegurar que no ha sido ejecutada por error. Además en la pantalla de confirmación está seleccionada por default la acción con menos impacto para el sistema.



**Módulo de profesores, Módulo de materias, Módulo de grupos, Módulo de semestres y  
Módulo de carrera**

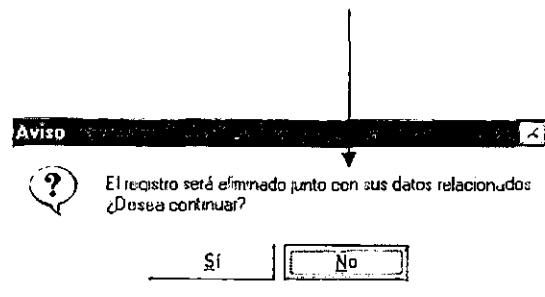
Cada uno de los módulos de profesores, materias, grupos, semestres y carreras contiene las opciones necesarias para agregar, eliminar y modificar registros, además muestra una lista de los profesores, materias, grupos, semestres o carreras del registro existentes en el sistema, dependiendo del módulo seleccionado.

Al agregar un nuevo registro o modificar uno existente se validan los datos, se valida la longitud de los datos y se pasan a mayúsculas.



Cuando el dato de nombre del registro no puede ser mostrada totalmente en la lista basta con hacer clic en ella para que aparezca una etiqueta con el nombre completo

Al realizar una modificación o eliminación aparece un cuadro de diálogo para confirmar la acción y asegurar que no ha sido ejecutada por error. Además en la pantalla de confirmación está seleccionada por default la acción con menos impacto para el sistema.



Cuadro de diálogo de confirmación

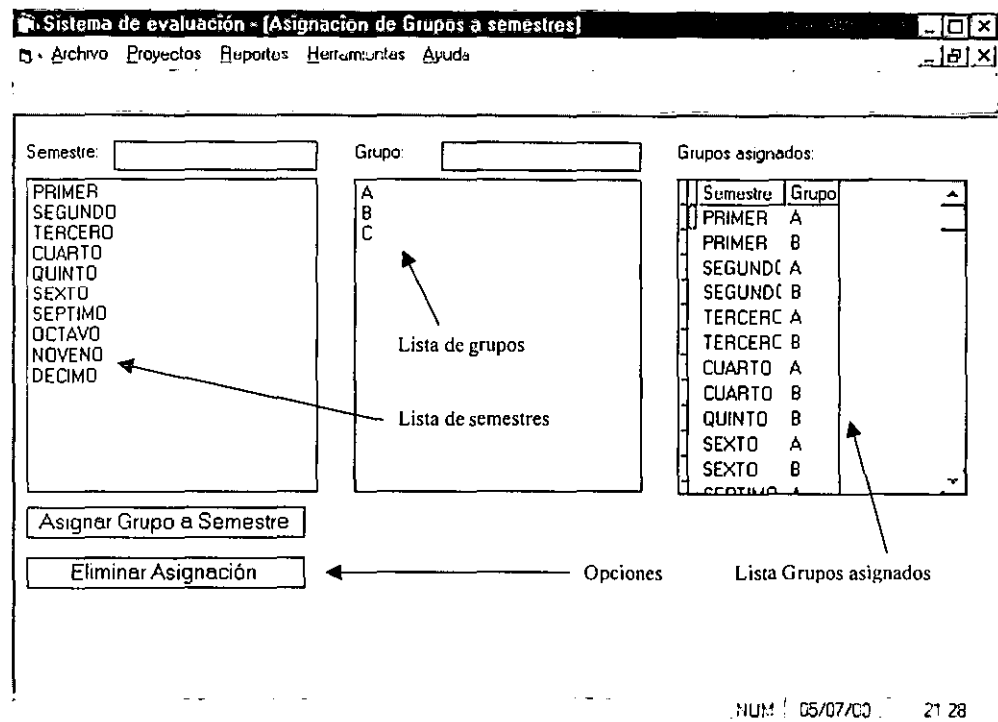
### **Módulo de asignación de semestres y grupos**

Datos capturados en el sistema necesarios para utilizar este módulo:

1. Datos de Semestres
2. Datos de Grupos

Una vez que se encuentran datos de alta los registros de profesores, materias, grupos, semestres y carreras debe procederse a la asignación de semestres y grupos, en esta parte se van a asignar grupos a semestres para conformar grupos de Primero A, Tercero B, etc.

Este Módulo muestra tres listados, uno de semestres, otro de grupos y en el tercer listado llamado "grupos asignados" se muestran las asignaciones existentes de semestres y grupos.



Pantalla de módulo de asignación de semestres a grupos

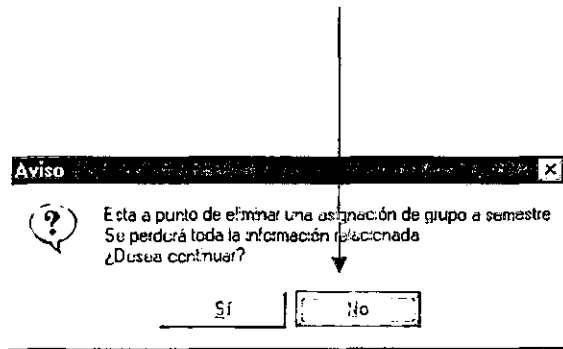
Este módulo tiene dos opciones:

- Asignar grupo a semestre: Esta opción crea un grupo al asignar un semestre y un grupo en un nuevo registro.
  1. Seleccionar un semestre a asignar de la lista de semestres
  2. Seleccionar el grupo a asignar de la lista de grupos
  3. Hacer clic en el recuadro [asignar grupo a semestre]
- Eliminar Asignación: Esta opción elimina un grupo, el cual debe ser seleccionado de la lista de grupos asignados.

1. Seleccionar un grupo de la lista de grupos asignados
2. Hacer clic en la opción [eliminar asignación]
3. Confirmar en el cuadro de diálogo

Se validan los datos antes de agregar el registro para no crear duplicados.

Al igual que en los módulos anteriores, se cuenta con mensajes de confirmación antes de realizar acciones de eliminación, por default se encuentra seleccionada la opción de no eliminar el registro.



Cuadro de diálogo de confirmación

## Módulo de asignación de profesores

Datos capturados en el sistema necesarios para utilizar este módulo:

- Datos de Profesores
- Datos de Materias
- Datos de Carreras
- Datos de asignación de semestres y grupos

Sistema de evaluación - [Form1]

Archivo Proyecto Reportes Herramientas Ayuda

Asignación de Profesores-Materia >>

Semestres:

| descripcion | descri |  |
|-------------|--------|--|
| PRIMER A    |        |  |
| PRIMER B    |        |  |
| SEGUNDO A   |        |  |
| SEGUNDO B   |        |  |
| TERCERO A   |        |  |
| TERCERO B   |        |  |
| CUARTO A    |        |  |
| CUARTO B    |        |  |
| QUINTO A    |        |  |
| QUINTO B    |        |  |
| SEXTO A     |        |  |
| SEXTO B     |        |  |
| SEPTIMO A   |        |  |
| SEPTIMO B   |        |  |

Carreras:

|                |
|----------------|
| INFORMATICA    |
| ADMINISTRACION |
| CONTADURIA     |

Materias:

|                         |
|-------------------------|
| ADMINISTRACIÓN II       |
| ADMINISTRACIÓN IV       |
| ADMINISTRACIÓN IX       |
| ADMINISTRACIÓN VI       |
| ADMINISTRACIÓN VIII     |
| ADMINISTRACIÓN X        |
| ADMÓN. BÁSICA II        |
| ADQS. Y ABASTECIMIENTOS |
| ALGORITMOS              |
| PRO. DE COMPUTADORAS    |
| AUDITORÍA III           |
| AUDITORÍA IV            |
| BASES DE DATOS          |
| CÓDIGO FISCAL FED       |
| COMP. HUMANO EN ORG     |

Agregar Semestre a la Carrera

Eliminar Semestre de la carrera

Asignar Materia al Semestre Carrera

Eliminar Materia al Semestre Carrera

Semestres Asignados la carrera:

| Semestre  | Grupo |  |
|-----------|-------|--|
| PRIMER A  |       |  |
| PRIMER B  |       |  |
| SEGUNDO A |       |  |
| SEGUNDO B |       |  |

Materias Asignadas al Semestre de la Carrera:

|                   |
|-------------------|
| ADMÓN. BÁSICA II  |
| CONTAB. BÁSICA II |
| INFORMATICA II    |
| MATEMÁTICAS I     |
| METODOLOGÍA II    |

NUM. CS/07/00 22 14

Pantalla de asignación de materias a semestres



En este módulo se llevan a cabo tres actividades muy importantes que relacionan todos los datos agregados en el sistema en los módulos de profesores, materias, carreras, semestres, grupos y materias. Consta de dos pantallas, una de asignación de materias a semestres y otra de asignación de profesores a materias, la pantalla con la que siempre inicia el módulo es con la de asignación de materias a semestres.

Las actividades son

- Asignar los grupos a las carreras para obtener grupos como Primero A de Informática, Quinto A de Administración, etc.

Para esto se tienen las siguientes opciones:

1. Agregar semestre a la carrera:

- Seleccionar un semestre de la lista de semestres
- Seleccionar la carrera a la que se desea asignar
- Verificar que el semestre no se encuentre en la lista de semestres asignados
- Hacer clic en la opción [asignar semestre] / Hacer doble clic en el semestre

2. Eliminar Asignación de semestre a la carrera

- Seleccionar un semestre de la lista de semestres
- Hacer clic en [eliminar semestre de la carrera]
- Confirmar en el cuadro de diálogo

Las acciones de asignación de semestre a las carreras están validadas y no permiten la creación de duplicados

- Asignar las materias que se imparten en cada uno de los grupos de las carreras. Tiene las siguientes opciones:

1. Agregar materia al grupo

- Seleccionar el semestre de la carrera al que se asignarán las materias
- Seleccionar la materia buscada de la lista
- Hacer clic en la opción [asignar semestre de la carrera] / hacer doble clic en la materia / presionar la tecla [ENTER]

2. Eliminar materia del grupo

- Seleccionar materia de la lista de materias
- Seleccionar la materia buscada de la lista
- Hacer clic en la opción [eliminar materia del semestre de la carrera] / presionar [DEL]

Las acciones de asignación de materias a semestre están validadas y no permiten la creación de duplicados.

- Asignar el profesor que imparte cada una de esas materias: Para ejecutar esta acción se hace clic en la opción asignación de profesor-materia que se encuentra en la esquina superior derecha. Entonces aparecerá la pantalla de asignación de profesores a materias.

Tiene las siguientes opciones:

1. Asignar materia al profesor: La cual sirve para establecer que profesor da la materia de matemáticas, por ejemplo, al grupo de primer semestre grupo A de la carrera de informática:
  - Seleccionar carrera del grupo
  - Seleccionar semestre
  - Seleccionar la materia a asignar de la lista de materias asignadas a la carrera.
  - Seleccionar el profesor poniendo su nombre en el cuadro de texto
  - Hacer clic en la opción de [asignar materia al profesor] / Doble clic en la materia a asignar

Las acciones de asignación de materias a profesor están validadas y no permiten la creación de duplicados.

2. Eliminar Asignación del profesor:
  - Seleccionar el profesor poniendo su nombre en el cuadro de texto
  - Seleccionar la materia asignada del recuadro de materias asignadas al profesor
  - Hacer clic en la opción [eliminar asignación de materia]
  - Confirmar en el cuadro de diálogo

Las acciones de eliminación de asignaciones de materias a profesor están validadas y no permiten la eliminación por descuido mediante el despliegue de un cuadro de diálogo de confirmación.

Sistema de evaluación - [Form1]

Archivo Proyectos Reportes Herramientas Ayuda

<< Asignación de Carrera-Materia

Carreras

|                       |
|-----------------------|
| INFORMÁTICA           |
| <b>ADMINISTRACIÓN</b> |
| CONTADURÍA            |

Nombre Profesor

ROBERTO MORENO P.

Materias Asignadas al profesor

| Carrera    | Semestre | Grupo | Materia       |
|------------|----------|-------|---------------|
| CONTADURÍA | CUARTO   | A     | MICROECONOMÍA |
| CONTADURÍA | CUARTO   | B     | MICROECONOMÍA |

Semestres Asignados a la carrera

| Semestre | Grupo |
|----------|-------|
| PRIMER   | A     |
| SEGUNDO  | A     |
| TERCERO  | A     |
| CUARTO   | A     |

Materias Asignadas al Semestre de la Carrera:

|                          |
|--------------------------|
| <b>ADMINISTRACIÓN II</b> |
| ADMÓN. BÁSICA II         |
| CONTAB. BÁSICA II        |
| INFORMÁTICA II           |
| MATEMÁTICAS I            |

Asignar Materia al profesor

Eliminar Asignación de materia

Capturando datos

NUM. 05/07/03 23 17

Pantalla de asignación de profesores a materia

Nota: La lista de semestres contiene los grupos asignados a cada semestre creados en el módulo de asignación de grupos a semestres.

## Módulo de Preguntas

Datos capturados en el sistema necesarios para utilizar este módulo:

- Datos de evaluaciones

Proyecto de evaluación:

Preguntas del proyecto: 8

| Pregunta  | Abierta   | Multip    |
|---|-----------|-----------|
| ASISTES REGULARMENTE A CLASES                       | NO        | NO        |
| ERES PUNTUAL  | NO        | NO        |
| <b>TIENES INTERES POR APRENDER LA MATERIA</b>       | <b>NO</b> | <b>NO</b> |
| TU NIVEL DE PARTICIPACION ES BUENO                  | NO        | NO        |
| TU NIVEL DE COMPROMISO CON EL CURSO ES BUENO        | NO        | NO        |
| TIENES UN BUEN NIVEL DE PREPARACION PARA COMPREN    | NO        | NO        |
| LAS HORAS QUE DEDICAS AL ESTUDIO (FUERA DEL AULA) S | NO        | NO        |

Opciones de la pregunta:

| Opcion | Texto | Val |
|--------|-------|-----|
| 1      |       | 1   |
| 2      |       | 2   |
| 3      |       | 3   |
| 4      |       | 4   |
| 5      |       | 5   |
| 6      |       | 6   |
| 7      |       | 7   |

NUM: 06/07/00 025

### Pantalla de módulo de respuestas

Las preguntas de esta evaluación pueden ser abiertas o cerradas (de opción) las preguntas abiertas permiten al usuario teclear su respuesta la cual no puede ser procesada para obtener calificaciones por el sistema, una pregunta de este tipo puede ser "Explica brevemente lo aprendido durante el curso". Las respuestas de opciones permiten al sistema obtener calificaciones de acuerdo al valor asignado a cada opción, un ejemplo puede ser "¿Cuál de los siguientes valores es negativo?". Cada respuesta puede tener más de una opción como

respuesta. Cada pregunta de opciones tiene además la posibilidad de recibir como respuesta una o más opciones, supongamos que se tiene una pregunta “¿Cuál de los siguientes es invertebrado?” y las opciones son A) pulpo, B) Caracol y C) Araña, las respuesta correcta sería Pulpo y Araña, entonces el usuario debe tener la opción de seleccionar dos opciones (en este caso el valor de la respuesta a la pregunta debe dividirse entre el número de opciones y asignársele como valor a cada opción)

En este módulo se realiza el manejo de las preguntas de las evaluaciones, tiene las siguientes opciones:

Para las opciones de preguntas es necesario seleccionar un proyecto de evaluación tecleando el nombre en el cuadro de texto

- Agregar Nueva pregunta: Agrega una pregunta al proyecto:
  1. Hacer clic en la opción de [agregar nueva pregunta]
  2. Establecer la descripción de la pregunta. Por ejemplo ¿Qué es un mamífero?
  3. Decidir si la respuesta será tomada de un grupo de opciones o será contestada de manera abierta.
  4. En caso de que sea establecida como de opción definir si hay más de una respuesta correcta para esa pregunta
- Agregar pregunta existente
  1. Hacer clic en la opción de [agregar pregunta existente]
  2. Seleccionar la pregunta tecleando la descripción en el cuadro de texto
  3. Hacer clic en la opción [agregar]
  4. Confirmar en el cuadro de diálogo

- **Modificar pregunta**
  1. Seleccionar pregunta
  2. Hacer clic en la opción de [modificar pregunta]
  3. Seleccionar la pregunta tecleando la descripción en el cuadro de texto
  4. Hacer las modificaciones necesarias
  5. Hacer clic en [aceptar]
  6. Confirmar en el cuadro de diálogo
  
- **Eliminar Pregunta**
  1. Seleccionar Pregunta
  2. Hacer clic en la opción de [eliminar pregunta]
  3. Seleccionar la pregunta tecleando la descripción en el cuadro de texto
  4. Confirmar en el cuadro de diálogo

Para el manejo de opción es necesario seleccionar una pregunta de la lista de preguntas del proyecto.

- **Agregar Nueva Opción:**
  1. Establecer el número o texto descriptivo de la opción
  2. Establecer un valor de calificación a la opción
  3. Hacer clic en [agregar opción]
  4. Confirmar en el cuadro de diálogo
  
- **Eliminar Opción**
  1. Seleccionar opción a eliminar
  2. Confirmar en el cuadro de diálogo

## Módulo de selección de evaluación

Datos capturados en el sistema necesarios para utilizar este módulo:

- Datos de evaluaciones
- Datos de asignación de semestres y grupos

**Sistema de evaluación - [Evaluación de desempeño del profesor]**

Archivo Proyectos Reportes Herramientas Ayuda

Proyecto de evaluación:

| Evaluación                                  |
|---|
| EVALUACION DEL DESEMPEÑO DEL ALUMNO         |
| EVALUACION DEL DESEMPEÑO DEL PROFESOR       |
| EVALUACION DEL DESEMPEÑO PREGUNTAS ABIERTAS |

Carreras

|                |
|----------------|
| INFORMATICA    |
| ADMINISTRACION |
| CONTADURIA     |

Evaluaciones que se aplicarán

Profesores que imparten en ese grupo

| Nombre                 | Materia             |
|------------------------|---------------------|
| ILDEFONSO HUANTE       | ADMÓN. BÁSICA II    |
| ENRIQUETA HERNÁNDEZ D. | CONTAB BÁSICA II    |
| CATALINA NUÑEZ         | INFORMATICA II      |
| ABEL GÁLVEZ A.         | MATEMÁTICAS I       |
| MA. CRISTINA CRUZ      | METODOLOGÍA II      |
| EFRAIN CABALLERO       | NOCIONES DE DERECHO |

Profesores que dan clases al grupo seleccionado

Grupos Asignados a la carrera:

| Semestre | Grupo |
|----------|-------|
| PRIMER   | A     |
| PRIMER   | B     |
| SEGUNDO  | A     |
| SEGUNDO  | B     |
| TERCERO  | A     |
| TERCERO  | B     |
| CUARTO   | A     |
| SEXTO    | A     |
| SEXTO    | B     |
| SEPTIMO  | A     |
| OCTAVO   | A     |

Establecer Evaluación

NÚM. (5/07/00) 2039

Pantalla de selección de evaluación

En este módulo se selecciona la evaluación (es) y se crea el proyecto de evaluación que se aplicará a un grupo de una carrera.

Los pasos a seguir son:

1. Seleccionar la(s) evaluación(es) a ser aplicada(s) seleccionándolas poniendo su nombre en el cuadro de texto.



2. Se selecciona la carrera a que corresponde el grupo en el que se aplicará la evaluación
3. Se selecciona el grupo de la lista de grupos asignados a la carrera
4. Se hace clic en la opción de [establecer evaluación]
5. Se confirma en el cuadro de diálogo

La acción de agregar evaluaciones está validada para evitar duplicados

La acción de establecer evaluación está validada para evitar evaluaciones nulas.

Las acciones de eliminación de asignaciones de materias a profesor están validadas y no permiten la eliminación por descuido mediante el despliegue de un cuadro de diálogo de confirmación.

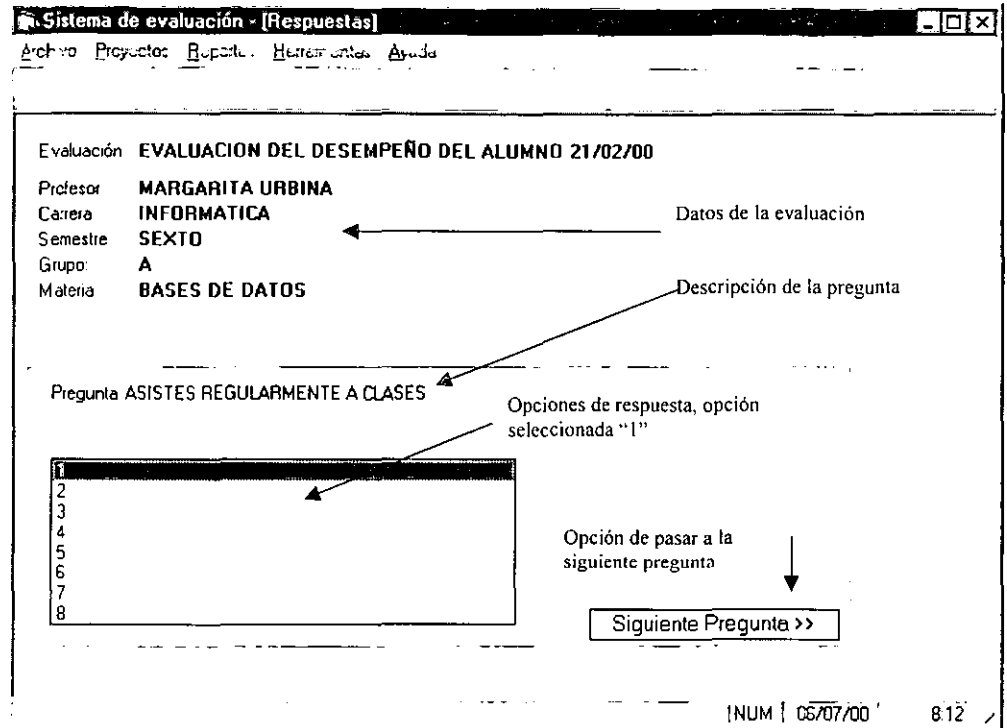
### **Módulo de aplicar evaluación**

Datos capturados en el sistema necesarios para utilizar este módulo:

- Datos de selección de evaluación

En este módulo se contesta el proyecto de evaluación seleccionado. Esta es la pantalla con la que el usuario, en el caso de los alumnos, evalúan el desempeño de los profesores. La pantalla de entrada de datos está dividida en dos sectores, el sector superior muestra el nombre de la evaluación, el nombre del profesor evaluado, la carrera, semestre y grupo en el que da clases (este grupo debe ser el que conteste la evaluación) y la materia que les imparte. En la parte inferior se encuentra el área de respuesta, delimitada con un rectángulo. En esta área aparece, si se trata de una pregunta abierta, un área de texto donde se puede escribir; o si se

trata de una pregunta de opciones, aparece una lista con las opciones de respuesta asignadas a esa pregunta.



Pantalla de respuesta

Las acciones en este módulo son:

1. Leer la pregunta
2. Seleccionar la opción(es) correctas
3. Hacer clic en la opción [siguiente pregunta>>].

El módulo está validado para no ejecutarse si no proyectos de evaluación seleccionados.

Las respuestas son almacenadas una vez que se da clic en la opción [siguiente pregunta>>], la entrada está validada para no permitir respuestas en blanco.

## Módulo de reportes Generales

Datos capturados en el sistema necesarios para utilizar este módulo:

- Datos de selección de evaluación
- Datos de módulo de respuestas
- Que las respuestas de la evaluación sean de opciones

Proyecto de evaluación:

Carrera:

Promedios Generales de profesores  
 Promedio General de Materias  
 Promedio General de Preguntas  
 Tabla general

Profesores Evaluados:

| Nombre             | Materia              | Semestre | Grupo | Carrera     | Promed |
|--------------------|----------------------|----------|-------|-------------|--------|
| ABEL GÁLVEZ A.     | MATEMÁTICAS I        | SEGUNDO  | A     | INFORMATICA | 5.46   |
| ABEL GÁLVEZ A.     | MATEMÁTICAS I        | SEGUNDO  | B     | INFORMATICA | 5.51   |
| ALFREDO HERRERA V. | FISCAL II            | SEXTO    | A     | CONTADURIA  | 5.32   |
| CARLOS HIDALGO A.  | AUDITORÍA III        | OCTAVO   | A     | CONTADURIA  | 5.33   |
| CARLOS HIDALGO A.  | AUDITORÍA III        | OCTAVO   | B     | CONTADURIA  | 5.32   |
| CARLOS HIDALGO A.  | CONTABILIDAD IV      | CUARTO   | B     | CONTADURIA  | 5.44   |
| CARMEN HERNÁNDEZ F | CONTAB BÁSICA II     | SEGUNDO  | B     | INFORMATICA | 5.37   |
| CATALINA NUÑEZ     | INFORMATICA II       | SEGUNDO  | A     | INFORMATICA | 5.49   |
| CATALINA NUÑEZ     | INFORMATICA II       | SEGUNDO  | B     | INFORMATICA | 5.53   |
| CATALINA NUÑEZ     | PROG. ORIENTADA A.O. | DECIMO   | A     | INFORMATICA | 4.92   |
| CATALINA NUÑEZ     | PROGRAMACIÓN O.O.    | OCTAVO   | A     | INFORMATICA | 5.53   |
| CATALINA NUÑEZ     | PROGRAMACIÓN O.O.    | OCTAVO   | B     | INFORMATICA | 5.40   |

Mostrar Gráfica      Imprimir Tabla

NUM 05/07/00      15 40

### Pantalla de reportes generales

En este módulo se tiene acceso a todos los reportes del sistema, los reportes se dividen en dos: Reportes por carrera y reportes generales. La diferencia entre ambos es la agrupación de los datos, y que los reportes por carrera pueden graficarse.

ESTA TESIS NO SALE  
DE LA BIBLIOTECA

Los pasos para obtener los reportes son los siguientes:

1. Seleccionar la evaluación introduciendo el nombre en el cuadro de texto.
2. Seleccionar la carrera de la lista desplegable en caso de que se requiera un reporte por carrera
3. Seleccionar el reporte que se quiere visualizar.

En los reportes generales, existen diferentes opciones para la obtención de los reportes deseados que permiten selección por:

- Profesores por carrera
- Promedio de Preguntas por carrera
- Promedio General de materias por carrera
- Promedio general de profesores
- Promedio General de materias
- Promedio General de preguntas
- Tabla general de información, que concentra toda la información.

En el caso de los reportes correspondientes a las respuestas abiertas, el número de reportes está limitado a respuestas por asignación de materia a profesor.

## Módulo de reportes de respuestas abiertas

Datos capturados en el sistema necesarios para utilizar este módulo:

- Datos de selección de evaluación
- Respuestas de evaluación abiertas

**Sistema de evaluación - [Reporte de Respuestas Abiertas]**

Archivo Proyectos Reportes Herramientas Ayuda

Proyecto de evaluación: **EVALUACION DEL DESEMPEÑO PREGUNTAS ABIERTAS**

Preguntas de la evaluación:  Mostrar respuestas de la pregunta seleccionada

| Pregunta  |
|---|
| ¿QUÉ FUÉ LO MAS IMPORTANTE PARA TI EN ESTE CURSO?               |
| COMENTARIOS Y SUGERENCIAS PARA MEJORAR EL DESEMPEÑO DEL MAESTRO |

Profesores:  Mostrar respuestas del profesor seleccionado

| Nombre              | Carrera     | Semestre | Grupo | Materia         |
|---------------------|-------------|----------|-------|-----------------|
| CARLOS HIDALGO A.   | CONTADURIA  | OCTAVO   | A     | AUDITORIA III   |
| CARLOS HIDALGO A.   | CONTADURIA  | OCTAVO   | B     | AUDITORIA III   |
| CARMEN HERNÁNDEZ F. | INFORMATICA | SEGUNDO  | B     | CONTAB BÁSICA I |
| CATALINA NUÑEZ      | INFORMATICA | OCTAVO   | A     | PROGRAMACIÓN C  |
| CATALINA NUÑEZ      | INFORMATICA | DECIMO   | A     | PROG ORIENTAD   |

Respuestas de la evaluación:

| Respuesta   |
|---|
| fue importante ya que me ayudo a aprender mas del tema de auditoria y de la elaboracion de los papeles d  |
| LA FORMA DE CONVIVIR CON SUS ALUMNOS  |
| Que entendí en buena parte lo que encierra la práctica de la materia, además creo que crecí un poco más   |
| Tal vez el haber podido apreciar más a fondo la práctica real de una auditoría con los aspectos más relev |

Imprimir respuestas del profesor

NUM : CS/07/CO 15 36

### Pantalla de módulo de reporte de respuestas abiertas

Los pasos a seguir son:

1. Seleccionar el proyecto de evaluación tecleando el nombre en el cuadro de texto
2. Seleccionar el nombre de profesor, grupo y materia que imparte.
3. Hacer clic en la opción [Imprimir respuestas del profesor]

El módulo de reporte de respuestas abiertas permite obtener reportes de las respuestas abiertas obtenidas en las evaluaciones.

### **3.4 Diseño de salidas**

El sistema de evaluación del desempeño de los profesores tiene las siguientes salidas: \_

#### **Reporte de preguntas abiertas por profesor a un archivo de texto.**

Los reportes de respuestas abiertas se envían de manera individual a un archivo, con el objeto de editar las respuestas del alumno, para eliminar comentarios desagradables o fuera de tono así como quitar respuestas que no tengan sentido. El guardar las respuestas en archivos separados permite además darle el formato adecuado para la impresión.

#### **Reportes generales de calificaciones**

El módulo de reportes contiene una lista de los reportes que pueden obtenerse del sistema, están divididos en dos columnas de opciones, la primera columna presenta reportes filtrados por carrera.

En la segunda columna se encuentran los reportes generales. El principal de estos reportes es el reporte de Tabla general, el cual recopila la información de todos los demás reportes de las opciones. Este reporte se muestra en una tabla que puede ser enviada directamente a la impresora.

Sistema de evaluación - [Reportes Generales]

Archivo Proyectos Reportes Herramientas Ayuda

Proyecto de evaluación: EVALUACION DEL DESEMPEÑO DEL ALUMNO

Carrera: Seleccionar Carrera

Promedios Generales de profesores  
 Promedio General de Materias  
 Promedio Profesores por Carrera  
 Promedio General de Preguntas  
 Promedio Preguntas por Carrera  
 Tabla general  
 Promedio General de materias por carrera

Profesores Evaluados:

| Nombre              | Materia              | Semestre | Grupo | Carrera     | Promed |
|---------------------|----------------------|----------|-------|-------------|--------|
| ABEL GÁLVEZ A.      | MATEMÁTICAS I        | SEGUNDO  | A     | INFORMATICA | 6.83   |
| ABEL GÁLVEZ A.      | MATEMÁTICAS I        | SEGUNDO  | B     | INFORMATICA | 6.89   |
| ALFREDO HERRERA V.  | FISCAL II            | SEXTO    | A     | CONTADURIA  | 6.65   |
| CARLOS HIDALGO A.   | AUDITORÍA III        | OCTAVO   | A     | CONTADURIA  | 6.66   |
| CARLOS HIDALGO A.   | AUDITORÍA III        | OCTAVO   | B     | CONTADURIA  | 6.65   |
| CARLOS HIDALGO A.   | CONTABILIDAD IV      | CUARTO   | B     | CONTADURIA  | 6.80   |
| CARMEN HERNÁNDEZ F. | CONTAB. BÁSICA II    | SEGUNDO  | B     | INFORMATICA | 6.71   |
| CATALINA NUÑEZ      | INFORMATICA II       | SEGUNDO  | A     | INFORMATICA | 6.86   |
| CATALINA NUÑEZ      | INFORMATICA II       | SEGUNDO  | B     | INFORMATICA | 6.91   |
| CATALINA NUÑEZ      | PROG. ORIENTADA A D. | DECIMO   | A     | INFORMATICA | 6.15   |
| CATALINA NUÑEZ      | PROGRAMACIÓN O.D.    | OCTAVO   | A     | INFORMATICA | 6.91   |
| FRANCA CAVALERO     | DESENGO MERCANTIL    | SEGUNDO  | A     | CONTADURIA  | 6.83   |

Mostrar Gráficas      Imprimir Tabla

Muestra la gráfica de los datos      [NUM | 06/07/00      22:53

### Pantalla de reportes generales

Antes de enviar a la impresora el sistema permite tener una vista previa del reporte solicitado.

En la vista previa puede seleccionarse el formato del reporte así como también darle una salida a un archivo de texto o HTML para poder presentarse en el Internet.

## Reporte de evaluación de desempeño del alumno

| Nombre         | Materia         | Semestre | Grupo | Carrera        | Promedio | Preguntas de la evaluación |      |      |      |      |
|----------------|-----------------|----------|-------|----------------|----------|----------------------------|------|------|------|------|
|                |                 |          |       |                |          | 1                          | 2    | 3    | 4    | 5    |
| ABEL GÁLVEZ A  | MATEMÁTICAS I   | SEGUNDO  | A     | INFORMATICA    | 6.83     | 7.61                       | 7.09 | 7.70 | 6.67 | 7.37 |
| ABEL GÁLVEZ A  | MATEMÁTICAS I   | SEGUNDO  | B     | INFORMATICA    | 6.89     | 7.58                       | 7.62 | 7.69 | 6.08 | 7.17 |
| ALFREDO        | FISCAL II       | SEXTO    | A     | CONTADURIA     | 6.65     | 6.92                       | 7.61 | 7.54 | 6.44 | 7.00 |
| CARLOS HDALGO  | AUDITORIA II    | OCTAVO   | A     | CONTADURIA     | 6.66     | 6.97                       | 7.10 | 6.89 | 6.82 | 6.86 |
| CARLOS HDALGO  | AUDITORIA II    | OCTAVO   | B     | CONTADURIA     | 6.65     | 6.97                       | 7.23 | 7.15 | 6.67 | 7.19 |
| CARLOS HDALGO  | CONTABILIDAD IV | CUARTO   | B     | CONTADURIA     | 6.80     | 7.07                       | 7.24 | 7.63 | 6.75 | 7.54 |
| CARMEN         | CONTAB. BÁSICA  | SEGUNDO  | B     | INFORMATICA    | 6.71     | 6.75                       | 6.77 | 7.38 | 7.15 | 6.80 |
| CATALINA NÚÑEZ | INFORMATICA II  | SEGUNDO  | A     | INFORMATICA    | 6.86     | 7.50                       | 7.15 | 7.91 | 6.86 | 7.05 |
| CATALINA NÚÑEZ | INFORMATICA II  | SEGUNDO  | B     | INFORMATICA    | 6.91     | 7.15                       | 7.00 | 8.00 | 6.31 | 7.33 |
| CATALINA NÚÑEZ | PROG. ORIENTADA | DECIMO   | A     | INFORMATICA    | 6.15     | 6.54                       | 6.31 | 7.23 | 5.83 | 6.46 |
| CATALINA NÚÑEZ | PROGRAMACIÓN    | OCTAVO   | A     | INFORMATICA    | 6.91     | 7.55                       | 7.63 | 7.70 | 6.70 | 7.42 |
| EFRAN          | DERECHO         | SEGUNDO  | A     | CONTADURIA     | 6.82     | 7.19                       | 7.31 | 6.88 | 6.94 | 7.03 |
| EFRAN          | DERECHO         | SEGUNDO  | B     | CONTADURIA     | 6.94     | 7.23                       | 7.43 | 7.55 | 6.52 | 7.10 |
| EFRAN          | DERECHO         | SEGUNDO  | A     | ADMINISTRACION | 6.27     | 7.35                       | 7.42 | 6.69 | 5.59 | 6.56 |
| EFRAN          | NOCIONES DE     | SEGUNDO  | A     | INFORMATICA    | 6.73     | 7.20                       | 7.15 | 7.05 | 6.37 | 6.68 |
| ENRIQUETA      | CONTAB. BÁSICA  | SEGUNDO  | A     | ADMINISTRACION | 6.52     | 7.38                       | 7.32 | 7.16 | 6.52 | 6.66 |
| ENRIQUETA      | CONTAB. BÁSICA  | SEGUNDO  | A     | INFORMATICA    | 6.33     | 7.05                       | 6.76 | 6.91 | 6.60 | 6.65 |

Pages 14

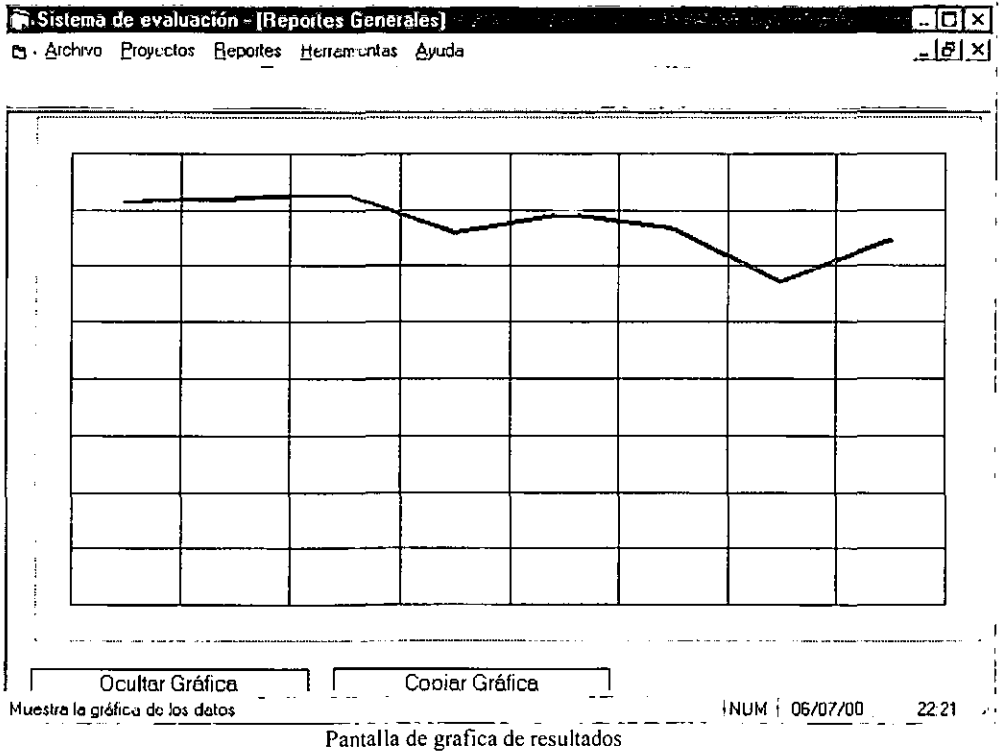
Pantalla de forma de impresión



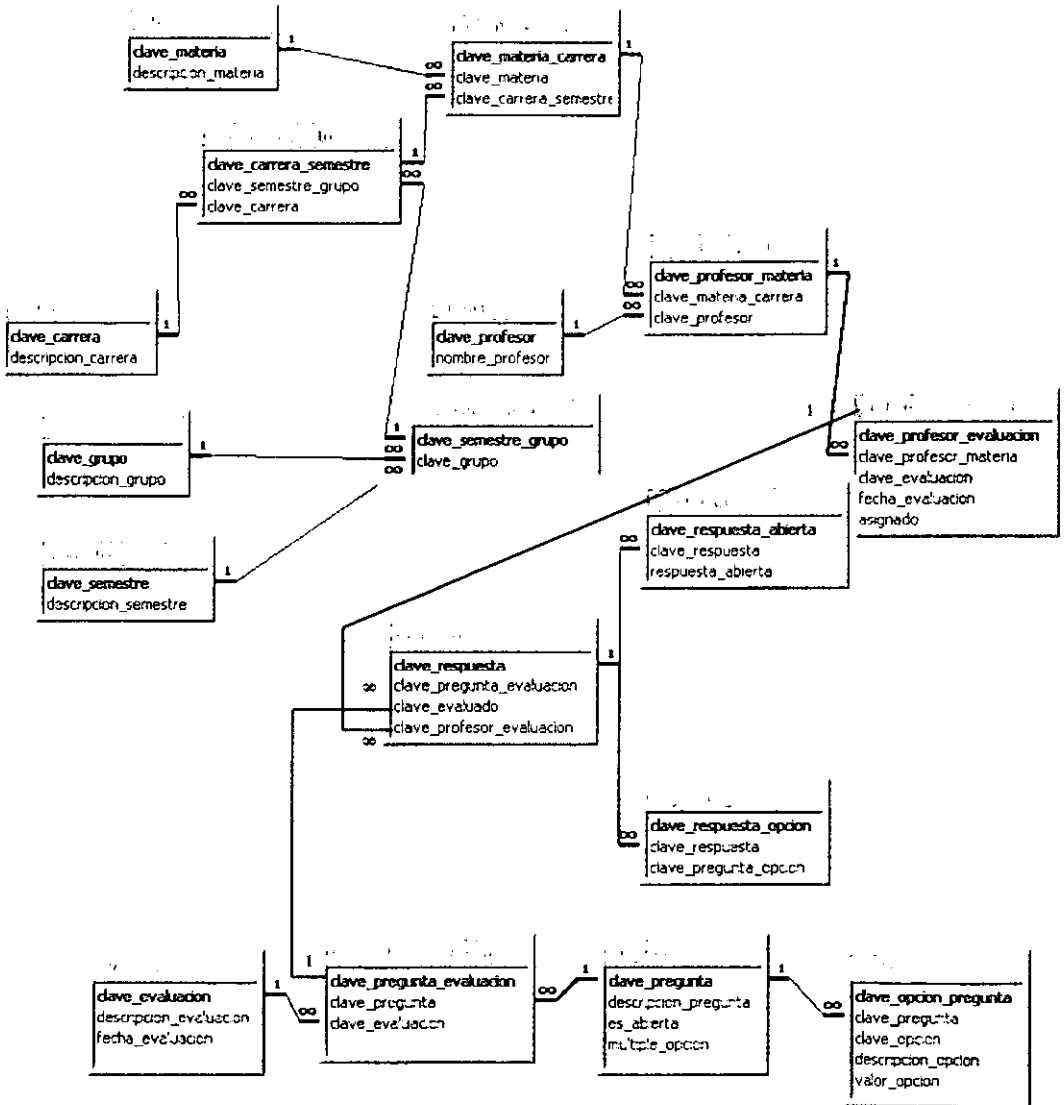
## Reportes por carrera de las calificaciones

Estos reportes muestran información filtrada por carrera, es posible visualizarla de manera de tabla o obtener un gráfico para comparar los valores.

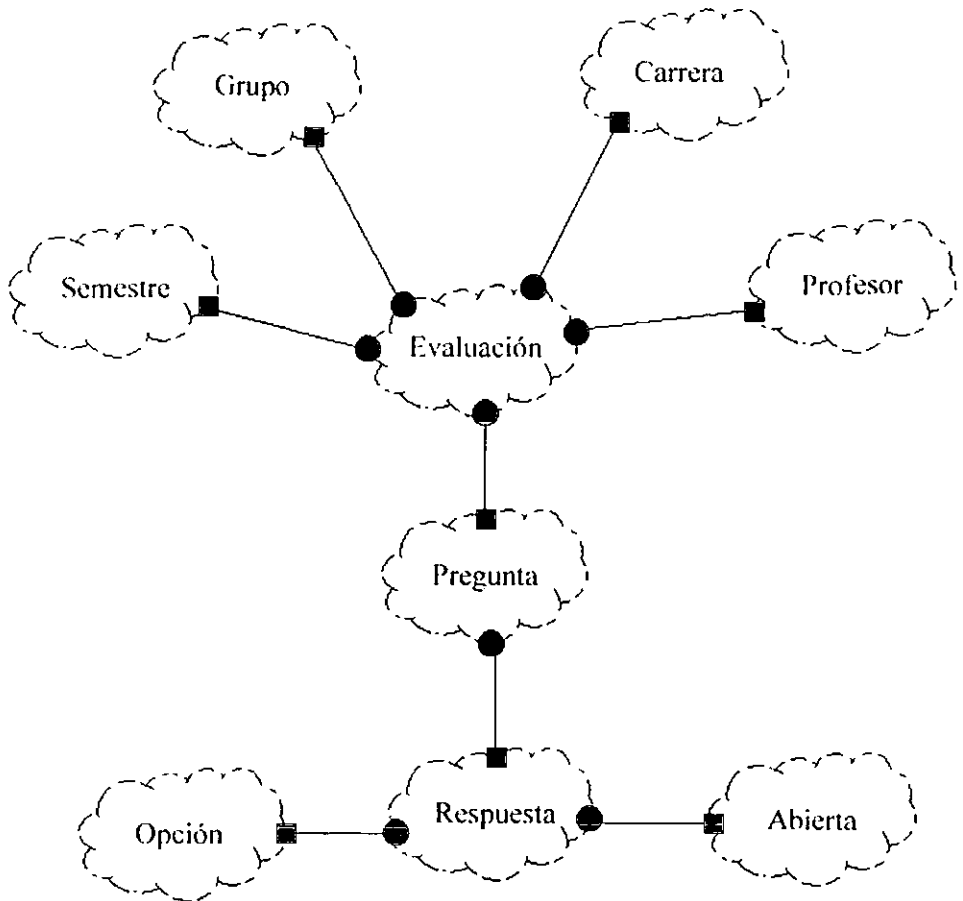
La gráfica creada puede importarse a los principales graficadores y hojas de cálculo como Excel o Lotus.



### 3.5 Diagrama de base de Datos



### 3.6 Diagrama de Clases



Dentro del diagrama de clases se muestran las relaciones todo parte con la conexión entre dos objetos, en los cuales uno está formado por otro u otros objetos.

En el caso de la presente tesis, se puede definir que el objeto evaluación esta formado por instancia de los objetos profesor, carrera, grupo y semestre.

Para simbolizar esta característica se conectan los objetos del diagrama con una línea terminada en dos conectores, cuadro significa “parte de”, y el círculo significa el “todo”.

Una relación todo-parte hace indispensable la existencia de los objetos “parte” para que pueda darse la creación del objeto “todo”.

Un diseño orientado a objetos especifica un conjunto de relaciones entre los objetos. Booch considera estas colaboraciones como “decisiones estructurales estratégicas” La conexión de los requerimientos a el diseño es soportado por el diseño orientado a objetos debido a la colaboración basada en modelos estáticos y dinámicos. El modelo dinámico describe la secuencia de mensajes que van de un objeto a otro, y el modelo estático describe los caminos que esos mensajes utilizan (CECIL;1995:28-29).

### 3.7 Programación

Al realizar el sistema de evaluación y del desempeño de los profesores y decidir el lenguaje de programación que utilizará para el desarrollo se tomaron en cuenta los siguientes aspectos.

Lugar donde sería aplicada la evaluación: **Laboratorios de la Universidad Don Vasco.**

Experiencia de los usuarios: **Alumnado de la Universidad**

Sistema operativo de los laboratorios: **Windows NT**

Facilidad de uso del lenguaje de programación

Capacidades de exportación / importación de datos.

Cantidad de usuarios que conocen la interfaz de los programas desarrollados en el lenguaje.

Disponibilidad del lenguaje de programación

Compatibilidad con el software de la red en el que será instalado

De acuerdo a la experiencia en los programas realizados, gran tiempo del desarrollo de programación es dedicado en crear la interfase del sistema para que sea más fácil al usuario, cada día las interfases de comandos son menos usuales y el uso del ratón que facilitar las tareas es casi un requisito.

El modelo de bases de datos que se maneja en Visual Basic de forma nativa (el mismo que utiliza Microsoft Access) y otras bases de datos que soportan SQL es un tanto diferente al de las bases de datos tradicionales como FoxPro, dBase, Clipper y Parados. Las bases de datos tradicionales están orientadas a registros. Las bases de datos del lenguaje de consulta estructurado (SQL) están orientadas a conjuntos de datos. Comprender la diferencia entre uno y otro me hizo decidirme por programar orientado a conjunto de datos.

En los sistemas orientados a registros, las operaciones que tienen que ver con la base de datos se realizan por un registro a la vez. En estos casos, la estructura del código que más se utiliza son los ciclos.

En el siguiente pseudocódigo, que sirve de ejemplo, se muestra la manera en que se incrementa un campo de precio en una tabla de un inventario contenido en una bases de datos orientada a registros, hay que prestar especial atención a que la misma operación se repite una y otra vez hasta que se cumple la condición buscada o se llega a los límites del archivo que contiene los registros :

BucleLectura:

Si FinDeArchivo

    IrA FinBucle

SiNo

    Leer Registro

    Si RegionDeVentas = 'Noreste' Entonces

        Precio= Precio \* 1.10

        Grabar Registro

    FinSi

FinSi

IrA BucleLectura

FinBucle:

Fin Programa

El procesamiento en sistemas orientados a registros por lo general trae consigo la generación de procedimientos que lean un registro, lo procesen, y vuelvan a leer o procesar hasta lograr el objetivo. Las bases de datos de este tipo utilizan índices para acelerar el proceso de localización de registros en las tablas.

En los sistemas orientados a conjuntos de datos las operaciones se llevan a cabo por cada conjunto de datos a la vez, y no por cada registro. En estos casos, la estructura que más se utiliza es la consulta SQL. En lugar de manejar ciclos para manejar cada registro, las bases de datos SQL pueden ejecutar operaciones con tablas enteras con una sola instrucción. El

siguiente pseudo código de ejemplo muestra la forma de alterar el mismo campo de precio visto antes, pero ahora en una base orientada a conjuntos de datos:

ACTUALIZAR Inventario

ESTABLECER Precio = Precio\*1.10

DONDE Inventario.RegionDeVentas = 'Noreste'

En este Pseudocódigo la palabra ACTUALIZAR equivale al comando UPDATE de SQL. UPDATE le indica al manejador de la base de datos que se va a hacer un cambio, específicamente en la tabla Inventario.

La palabra ESTABLECER equivale al comando SET de SQL, el cual indica el campo o los campos que serán modificados y su nuevo valor. La palabra DONDE equivale a WHERE de SQL, la cual establece el conjunto de datos sobre los cuales se aplicará la modificación, en este caso todos los registros cuya región de ventas sea Noreste.

Como se puede apreciar en el ejemplo, las bases orientadas a conjuntos de datos, se generan instrucciones sencillas para llevar a cabo las acciones deseadas. Con tan solo identificar el conjunto de datos se puede aplicar la operación en todos los registros contenidos en él. En sistemas de conjuntos de datos los índices se utilizan para control de los registros más que para acelerar los procesos.

(Programación de Bases de Datos,1998:45-46)

Cómo trabaja Visual Basic

- Programación dirigida por eventos. Un evento es una acción de algún tipo. Por ejemplo, mover el ratón, seleccionar un elemento de una lista o hacer clic en un botón

son eventos. Estos eventos dirigen lo que se ejecutará en visual Basic. Esta idea es muy diferente a la programación tradicional en la cual hay un programa principal el cual se ejecuta. En la programación tradicional, se escriben muchas líneas de código las cuales se ejecutan de manera secuencial. En la programación dirigida pro eventos, el código que se ejecuta está determinado por el evento ocurrido.

- Interfaz Gráfica de Usuario (GUI): Visual Basic está diseñado de tal manera que inmediatamente puede observarse lo que se está desarrollando. Esta utiliza dos clases de objetos:
  1. Formas: Que son las ventanas que uno y crea y modifica.
  2. Controles: Son objetos gráficos colocados en las formas, los cuales pueden ser: botones, cajas de textos, listas, controles de tiempo, etc..

Para programar, se agrega código a cada uno de los objetos para manejar los eventos que reciban.

- Archivos de un programa de Visual Basic:
  1. .mak: Archivos de proyecto: contienen una lista de todas las formas que conforman el proyecto y otros archivos de configuración, reportes, etc.
  2. .frm: Archivo de Forma: contiene a la forma y los objetos contenidos en ella
  3. .exe: Archivo ejecutable: compila toda la información del proyecto en formato que se pueda ejecutar sin visual Basic. Este archivo es el que comúnmente se distribuye una vez que la aplicación esta terminada completamente.



4. .frx: Incluye configuraciones de la forma y imágenes graficas puestas en ella o en los controles. (<http://www.vanderbilt.edu/Engineering/CIS/Sloan/web/es130/vbasic/howvb.html>, Junio del 2000)

En el presente capítulo se encuentra el diseño del sistema, lo cual es la base para un buen desarrollo, en mi opinión personal, más allá de las especificaciones de las entradas y salidas, la parte más importante en el diseño es el que corresponde a la base de datos, ya que ese diagrama muestra claramente las entidades y sus relaciones.

# Capítulo IV

---

## Lenguajes de Programación

Un programa es un conjunto de instrucciones secuenciales escritas en un lenguaje de programación que recibe la computadora para que sean procesadas y cumplan una tarea. Inicialmente los programas eran utilizados para manejar el hardware, pero hoy en día son utilizados con diversas aplicaciones, principalmente en el manejo de bases de datos y para realizar cálculos complejos (Enciclopedia Didáctica de la Computación;1999: 310).

### 4.1 Tipos de lenguajes

#### 4.1.1 Lenguajes máquina y primeros lenguaje simbólicos

Los lenguajes de programación son los lenguajes en los cuales un programador escribe las instrucciones (programa) que una computadora ejecutará. Los primeros lenguajes de programación eran lenguajes ensambladores, cuyas instrucciones no eran muy diferentes al código binario que ejecutaban las computadoras. Los lenguajes ensambladores eran muy difíciles de aprender, programar y depurar. Además de que el código se escribía sólo para una computadora en especial y para ejecutarlo en una computadora diferente tenía que re-escribirse completamente. Conforme se extendió el uso de las computadoras y se necesitó compartir los datos, se crearon nuevos lenguajes (a principios de 1955) Estos lenguajes se denominan simbólicos pues ya no se usan solamente números binarios, sino también mnemónicos. FORTRAN (Formula

Translator / Traductor de fórmulas) fue uno de los primeros lenguajes de programación y era muy parecido al lenguaje ensamblador, sin embargo permitía a los programas escribir expresiones algebraicas en vez de instrucciones codificadas para las operaciones aritméticas

(<http://www.britannica.com/bcom/eb/article/3/0,5716,117723+2,00.html> 28 de Febrero de 2000 ), (Enciclopedia Didáctica de computación, 1999:305, 310).

COBOL (Common Business-Oriented Language) fue desarrollado para manejar registros y archivos y las operaciones necesarias para aplicaciones de negocios simples. La tendencia desde entonces ha sido encaminada a desarrollar programas cada vez más abstractos que permitan escribir lenguajes más fáciles de entender, aprender y depurar. Cada vez más distintos al código binario de la máquina (<http://www.britannica.com/bcom/eb/article/3/0,5716,117723+2,00.html> 28 de Febrero de 2000).

#### **4.1.2 Lenguajes Imperativos**

COBOL y FORTRAN, y sus descendientes como PASCAL y C se conocen como lenguajes imperativos debido a que se especifican como una secuencia de ordenes de cómo la máquina va a resolver un problema, esto no es muy diferente a lo que sucede en el código binario que ejecuta la máquina (Ibíd.) (Enciclopedia Didáctica de la Computación;1999: 300).

### **4.1.3 Lenguajes declarativos**

En este caso la programación se realiza llamando o invocando funciones o procedimientos, los cuales son secciones de código que se ejecutan dentro de un programa. El lenguaje más conocido de este tipo es LISP (LIST Processing/ Procesador de LISTas), en el cual todo proceso es expresado como la aplicación de una función a uno o más objetos. Como dichos objetos pueden ser otras funciones así como datos individuales (variables) o estructuras de datos, un programador puede crear funciones que resuelvan todo un problema a un nivel muy alto de abstracción al sólo pasarle las variables necesarias.

### **4.1.4 Lenguajes de programación Lógica**

La característica funcional ha hecho a LISP un lenguaje muy popular para las aplicaciones de inteligencia artificial, más sin embargo ha sido relegado poco a poco por lenguajes de programación lógica como el PROLOG (PROgramming in LOGic / Programando en Lógica). Estos son lenguajes declarativos en los que el programador especifica los resultados que deben de obtenerse pero no los métodos que deben aplicarse para lograrlos. PROLOG está basado en los conceptos de resolución (relacionado con la deducción lógica) y la unificación (parecida a la búsqueda de patrones).

Los programas escritos en estos lenguajes son una secuencia de objetivos. Una reciente extensión de la programación lógica es una adición de restricciones a la programación lógica en la cual la búsqueda de patrones es reemplazada por el establecimiento de

restricciones que es más general. De nuevo, los programas son una secuencia de objetivos a realizar, pero ahora deben satisfacer las restricciones especificadas. Algunos lenguajes de esta categoría son: PROLOG, LISP, PROLOG III, CHIP, ECLIPSE y CLP(R) (<http://www.britannica.com/bcom/eb/article/3/0,5716,117723+2.00.html> 28 de Febrero de 2000).

#### **4.1.5 Lenguajes Orientados a Objetos**

Otras dos tendencias en los lenguajes de programación son el soporte de la encapsulación y la concurrencia. La encapsulación de datos es donde toda la programación se hace en términos de objetos. Un objeto consiste en un conjunto de datos junto con procedimientos para operar los datos. La encapsulación se refiere al hecho de que los datos del objeto sólo pueden accederse a través de los métodos o procedimientos establecidos (STROUSTRUP,1993:43). La programación es hecha creando objetos que se envían mensajes de unos a otros para realizar tareas invocando los métodos de los otros. El paradigma de la orientación a objetos ha sido muy influyente, recientemente ha influenciado a la tecnología de las bases de datos. Es una buena idea el crear objetos de los datos con los lenguajes de la programación orientada a objetos y entonces construir una base de datos orientada a objetos que pueda ser accedida y manipulada con las características del lenguaje orientado a objetos. El problema es que los datos deben ser almacenados en un medio no volátil, mientras que los lenguajes orientados a objetos manejan principalmente estructuras en memoria. Algunos lenguajes orientados a objetos son:

Lenguaje C++, PASCAL, ADA, FORTRAN y MODULA2  
<http://www.britannica.com/bcom/eb/article/3/0,5716.117723+2,00.html> 28 de Febrero de 2000), (Enciclopedia Didáctica de la computación, 1999: 311).

#### **4.1.6 Lenguajes Persistentes**

Como resultado de las debilidades de los lenguajes de programación orientados a objetos para el manejo de datos surgieron los lenguajes persistentes, en los cuales se integra la funcionalidad de los lenguajes tradicionales con las características de los sistemas de bases de datos en un mismo lenguaje (Ibíd.).

#### **4.1.7 Lenguajes Concurrentes**

Concurrencia se refiere a la ejecución de más de un procedimiento al mismo tiempo, por ejemplo el acceso a datos compartidos, ya sea verdaderamente simultáneo, en un sistema con múltiples procesadores, o de manera interpolada. Lenguajes como el ADA incluyen características como encapsulación y características que permiten al programador especificar las reglas de interacción entre los procedimientos o tareas concurrentes (Ibíd.).

Entre los lenguajes concurrentes se encuentran: PASCAL concurrente y ADA

#### **4.1.8 Lenguajes Visuales**

A un nivel superior de abstracción se encuentran los lenguajes de programación visual, en los cuales el usuario expresa gráficamente lo que quiere hacer mediante iconos que representan datos o instrucciones. Visual C++, Visual Dbase, Visual Basic,

Visual Java (posiblemente discontinuado debido a problemas legales de sus creadores) y Visual Fox son algunos de los lenguajes visuales más populares (Ibíd.).

## **4.2 Interprete y compilador**

Los programas de computadora escritos en otro lenguaje que no sea el lenguaje máquina deben ser interpretados o compilados. Un interprete es un programa que examina el código fuente de un programa instrucción tras instrucción y ejecuta las operaciones requeridas para esa instrucción. Este proceso hace la ejecución más lenta. Un compilador es un programa que traduce el código fuente del programa completamente a código máquina, el cual es guardado para que sea ejecutado (Ibíd.).

Los lenguajes de programación han sido una herramienta accesorio de la computadora, en un principio eran los propios diseñadores de las computadoras quienes creaban los lenguajes y programaban, después surgieron los programas comerciales que satisfacían algunas de las necesidades de los usuarios, pero conforme había un mayor conocimientos de las computadoras y los usuarios tuvieron acceso a las computadoras y su tecnología se hizo más barata, entonces se pudo obtener lenguajes de programación orientados a que el mismo usuario creara programas a su medida y satisficiera sus necesidades particulares.

En la actualidad se tienen múltiples opciones a la hora de llevar a cabo un sistema, y se puede decir que existe en lenguaje de programación específico para cada tipo de desarrollo. La facilidad de los lenguajes de programación actuales, como los visuales, compiten contra la robustez y generalidad de los antiguos. En mi apreciación personal creo que uno de los

lenguajes con más auge y vida por delante es el lenguaje Java, el cual debido a sus prestaciones para trabajar en Internet le da gran campo de acción.



# Capítulo V

---

## Ingeniería de software

Con el fin de obtener sistemas de información desarrollados con mejor calidad y que cumplan con los estándares establecidos en análisis y diseño, es necesario aplicar ciertas técnicas para asegurar una vida útil más extensa al sistema. La aplicación de la ingeniería de software no solamente garantiza que el sistema cumple con ciertas normas en su creación sino que además asegura que el manejo y mantenimiento del sistema puede llevarse a cabo por personas ajenas al desarrollo original, evitándose así la dependencia a los creadores del programa.

### 5.1 Análisis y diseño de sistemas de información

La construcción de un sistema de información cae dentro del dominio de la ingeniería. Al igual que otras disciplinas de ingeniería, la naturaleza y las herramientas de la ingeniería de información están relacionadas con desarrollos tecnológicos y una mejor percepción de las necesidades sociales de servicios de información.

Los primeros sistemas de información estaban diseñados para ser operados por expertos en información, y frecuentemente no cumplían con su de facilitar la obtención de información a los usuarios. Los sistemas de información modernos son cada vez más utilizados por personas que tienen poca o nula experiencia con la tecnología de información, pero poseen una mejor percepción acerca de lo que debe acompañar a la tecnología en el ambiente profesional y personal. Un entendimiento correcto de estos requerimientos, preferencias y “estilos de

información” de los usuarios es crucial para el diseño y el éxito de los sistemas de información de hoy en día.

La metodología involucrada en un sistema de información consiste en un conjunto de actividades interactivas conocidas como el ciclo de vida de los sistemas

(<http://www.britannica.com/bcom/eb/article/7/0,5716,109287+1+106312,00.html>).

Los programas se han convertido en el elemento clave de la evolución de los sistemas y productos informáticos. En las pasadas cuatro décadas, el software ha pasado de ser una resolución de problemas especializada y una herramienta de análisis de información, a ser una industria por si misma. Pero la temprana cultura e historia de la”programación” ha creado un conjunto de problemas que persisten todavía hoy.

La ingeniería del software es una disciplina que integra métodos, herramientas y procedimientos para el desarrollo del software de computadora. Se han propuesto varios paradigmas diferentes, cada uno exhibiendo ventajas y desventajas, pero todos tienen una serie de fases genéricas en común.

## **5.2 Paradigmas de Ingeniería**

### **5.2.1 El ciclo de vida Clásico**

El paradigma del ciclo de vida clásico plantea un enfoque sistemático y secuencial del software que comienza en el nivel del sistema y progresa a través del análisis, diseño, codificación, prueba y mantenimiento. Modelizado a partir del ciclo convencional de una ingeniería, el paradigma del ciclo de vida abarca las siguientes actividades:

- Ingeniería y análisis de sistema. Debido a que el software es siempre parte de un sistema mayor, el trabajo comienza estableciendo los requisitos de todos los elementos del sistema y luego asignando algún subconjunto de estos requisitos al software. Este planteamiento del sistema es esencial cuando el software debe interrelacionarse con otros elementos tales como el hardware, persona y bases de datos. La ingeniería y el análisis del sistema abarca los requisitos globales a nivel del sistema con una pequeña cantidad de análisis y diseño a un nivel superior.
- Análisis de los requisitos del software: el proceso de recopilación de los requisitos se centra e intensifica especialmente para el software. Para comprender la naturaleza de los programas que hay que construir, el ingeniero de software (“analista”) debe comprender el ámbito de la información del software, así como la función, el rendimiento y las interfaces requeridas. Los requisitos, tanto del sistema como del software se documentan y se revisan con el cliente.
- Diseño: el diseño de software es realmente un proceso multipaso que se enfoca sobre cuatro atributos distintos del programa: la estructura de los datos, la arquitectura del software, el detalle procedimental y la caracterización de la interfase. El proceso de diseño traduce los requisitos en una representación del software que pueda ser establecida de forma que obtenga la calidad requerida antes de que comience la codificación. Al igual que los requisitos, el diseño se documenta y forma parte de la configuración del software.

- **Codificación:** El diseño debe traducirse en una forma legible para la máquina. El paso de codificación realiza esta tarea. Si el diseño se realiza de un manera detallada, la codificación puede realizarse mecánicamente.
- **Mantenimiento:** Los programas, individualmente, sufrirán cambios después de que se entregue al cliente. Los cambios ocurrirán debido a que se hayan encontrado errores, a que los programas deban adaptarse a cambios del entorno externo, por ejemplo un cambio solicitado debido a que se tiene un nuevo sistema operativo o dispositivo periférico, o debido a que el cliente requiere aplicaciones funcionales o del rendimiento. El mantenimiento del software aplica cada uno de los pasos precedentes del ciclo de vida de un programa existente en vez de a uno nuevo. (Ingeniería de Software; PRESSMAN:29).

El ciclo de vida clásico es el paradigma más antiguo y ampliamente usado en la ingeniería del software. Sin embargo, con el paso de unos cuantos años se han producido críticas al paradigma, incluso por seguidores activos, que cuestionan su aplicabilidad a todas las situaciones. Entre los problemas que se presentan algunas veces cuando se aplica el paradigma del ciclo de vida clásico se encuentran:

1. Los proyectos en la realidad no siguen invariablemente el ciclo de vida que propone el modelo, regularmente se repite alguna etapa del ciclo y en diferente orden.
2. Para los usuarios es difícil identificar todas sus necesidades al principio, el modelo de ciclo de sistema los requiere todos en un principio y no plantea la posibilidad de incertidumbre en un comienzo.

3. El usuario del sistema debe esperar hasta que el proyecto esté casi terminado para poder probarlo. Siguiendo el ciclo de vida del modelo clásico no se encontrarán errores importantes hasta que ya casi todo está terminado.

Basándonos en los puntos anteriores podemos deducir que el modelo clásico tiene muchos aspectos que no pueden ser llevados de manera rígida a los sistemas reales, pero el modelo de ciclo de vida es ya reconocido ampliamente. Es importante porque da un panorama general de los métodos de análisis, diseño, codificación, prueba y mantenimiento. Y a pesar de sus inconveniencias es mucho mejor utilizarlo que desarrollar software sin guías (Ibid.:26,27).

### **5.2.2 Creación de prototipos**

Generalmente cuando un cliente hace la petición de un sistema no establece los requisitos detallados de entrada y salida, puede haber dudas respecto a los algoritmos o del funcionamiento con un sistema operativo en particular como es el caso del proyecto de esta tesis. En éste tipos de situaciones y otras que involucren incertidumbre, es conveniente utilizar la construcción de prototipos como método de ingeniería.

Los prototipos permiten la creación de un modelo del programa que se va a construir, existen tres tipos de prototipos:

1. Prototipo en papel o un modelo basado en pc que describa la interacción hombre-máquina, de forma que el usuario perciba como se manejará el sistema.

2. Un prototipo que implemente algunas funciones del sistema deseado.
3. Un programa que ejecute parte o todas las funciones deseadas, pero que tenga características que puedan ser mejoradas durante el desarrollo posterior.

La figura inferior muestra los pasos en el desarrollo de prototipos. (Ibíd.:28)

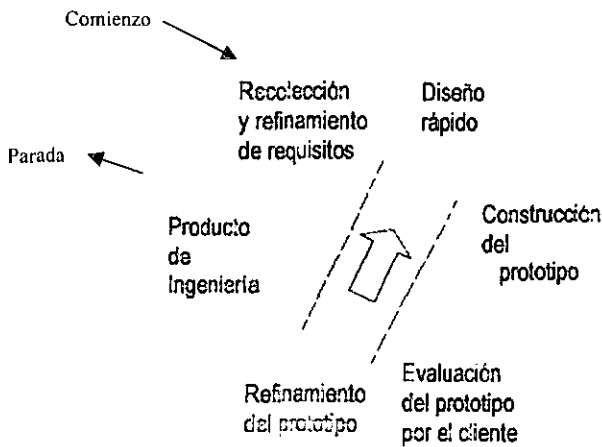


Figura de Creación Prototipos

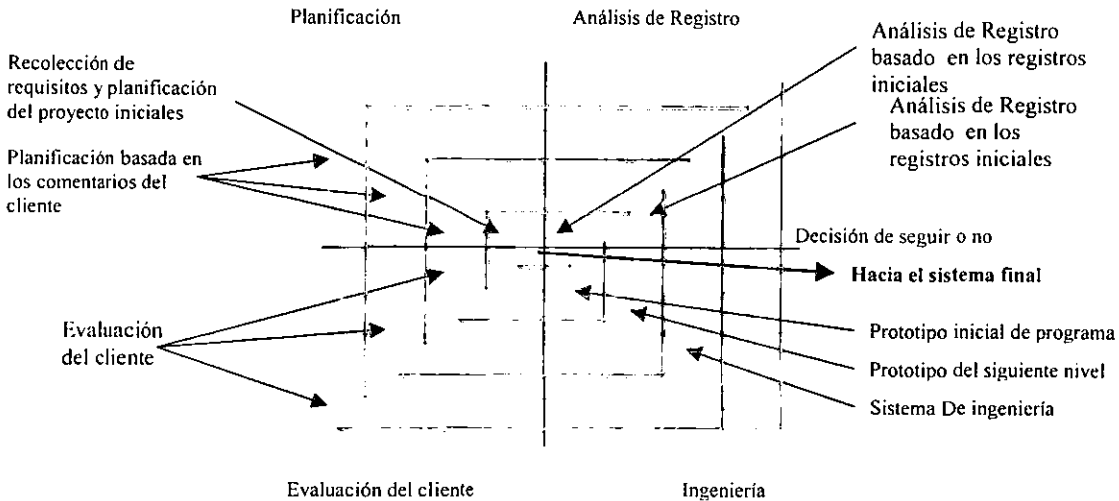
Fuente: Ingeniería de software; PRESSMAN (Pág. 29)

### 5.2.3 El modelo en espiral

Este modelo cumple con las características del ciclo de vida de los sistemas y con el de prototipos, pero añade un nuevo elemento faltante en dichos paradigmas: el riesgo.

Este modelo tiene cuatro actividades principales:

1. Planificación: Determinación de objetivos, alternativas y restricciones.
2. Análisis de riesgo: Análisis de las alternativas e identificación y resolución de los riesgos.
3. Ingeniería: desarrollo del producto de siguiente nivel o versión.
4. Evaluación del cliente: Valoración de los resultados de la ingeniería.



Fuente: Ingeniería de software PRESSMAN (Pág. 31)

Si analizamos la figura del espiral, podemos darnos cuenta que con cada vuelta alrededor del centro, se construyen nuevas versiones del sistema cada vez más completas.

En la primera vuelta se establecen los objetivos, alternativas, restricciones, objetivos y se analizan e identifican los riesgos.

Cuando durante el análisis se descubre un cierto grado de incertidumbre, se puede usar la creación de prototipos en el cuadrante de ingeniería para dar asistencia tanto al encargado del desarrollo como al cliente. Se pueden usar simulaciones y otros modelos para definir más el problema y refinar los requisitos. El cliente evalúa el trabajo de ingeniería (cuadrante de evaluación del cliente) y sugiere modificaciones. En base a los comentarios del cliente se produce la siguiente fase de planificación y de análisis de registro. En cada vuelta alrededor de la espiral, el fin del análisis resulta en una decisión de “seguir o no seguir”. Cuando hay ciertos detalles que aún no se solucionan, se sigue avanzando alrededor de la espiral, esto implica que los desarrolladores crean cada vez un modelo más completo del sistema y, al final, al propio sistema operacional.

Cada vuelta alrededor del espiral requiere ingeniería (cuadrante inferior derecho), que se puede llevar a cabo mediante el enfoque de vida clásico o de la creación de prototipos (Ibíd.:31).

El paradigma del modelo en espiral para la ingeniería del software es actualmente el enfoque más realista para el desarrollo del software y de sistemas a gran escala. Utiliza un enfoque “evolutivo” para la ingeniería de software, permitiendo al desarrollador y al cliente entender y reaccionar a los riesgos en cada nivel evolutivo. Utiliza la creación de prototipos como un mecanismo de reducción de riesgos, pero lo que es más importante, permite a quien lo desarrolle aplicar el enfoque de creación de



prototipos en cualquier etapa de la evolución del producto. Mantiene el enfoque sistemático correspondiente a los pasos sugeridos por el ciclo de la vida clásico, pero combinándolo con interacción, lo que refleja de forma más realista el mundo real. El modelo en espiral demanda una consideración directa de riesgos técnicos en todas las etapas del proyecto y, si se aplica adecuadamente, debe reducir los riesgos antes de que se conviertan en problemáticos.

Pero, al igual que otros paradigmas, el modelo en espiral puede no ser la mejor opción. Puede ser difícil convencer a grandes clientes (particularmente cuando se desarrolla bajo contrato) de que el enfoque evolutivo es controlable. Requiere una considerable habilidad para la valoración del riesgo y cuenta con esta habilidad para el éxito, si no se descubre un riesgo importante indudablemente surgirán problemas. Por último, el modelo en si mismo es relativamente nuevo y no se ha usado tanto como el ciclo de vida o la creación de prototipos (Ibíd.:32).

#### **5.2.4 Técnicas de Cuarta Generación (T4G)**

Este paradigma toma en consideración las herramientas de software que facilitan el desarrollo de software llevando a cabo actividades como:

- La especificación de algunas características de los programas de alto nivel.
- Generación de automática de código fuente basada en especificaciones.

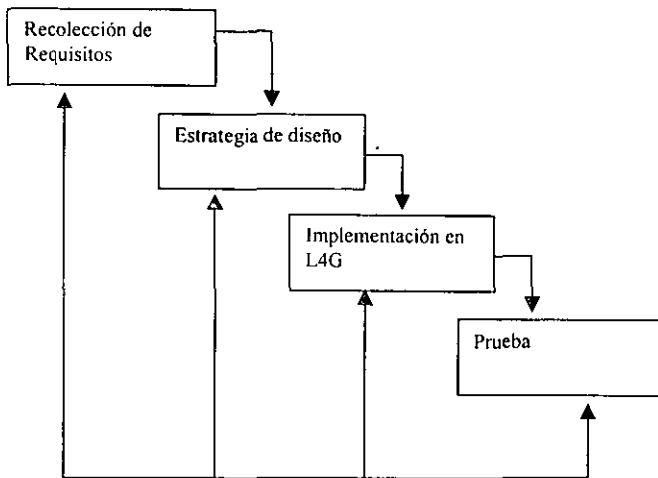
En la actualidad un entorno de desarrollo que soporte el paradigma de T4G puede incluir todas o algunas de las siguientes herramientas:

- Lenguajes procedimentales para consulta a bases de datos
- Generación de informes

- Manipulación de datos
- Interacción y definición de pantallas
- Generación de código
- Facilidades gráficas de alto nivel
- Facilidades de hoja de cálculo

El siguiente diagrama describe el paradigma T4G. Al igual que los paradigmas anteriores, se comienza con la recolección de los requisitos. Lo mejor es que el cliente sea el que los especifique. Se supone que dichos requisitos se pasan directamente a un prototipo funcional, pero en la realidad esto no puede ser posible debido a que muchas veces los requisitos no están claros o simplemente no pueden pasarse dichos requisitos de manera que sean aceptados por una herramienta T4G. Por eso es indispensable la comunicación entre el cliente y el desarrollador, al igual que en los paradigmas anteriores.

## Diagrama Técnicas de cuarta generación



Fuente: Libro de Ingeniería de Software; PRESSMAN (Pág. 33 )

Actualmente existen personas a favor y en contra del paradigma de T4G, los defensores explican que su utilización reduce considerablemente el tiempo de desarrollo de los programas y una mayor productividad en el personal que lo desarrolla. Por otro lado los que están en contra especifican que las herramientas de hoy en día de T4G no son fáciles de utilizar y el código obtenido es muchas veces ineficiente, y desde luego que tampoco los mantenimientos son fáciles de realizar.

Debido a que existen pocos estudios controlados acerca de las T4G es difícil establecer su eficiencia. El estado actual se puede resumir en los siguientes puntos:

1. Las T4G están limitadas hoy en día a sistemas de análisis de información y obtención de informes de bases de datos.

2. Datos preliminares de compañías que utilizan T4G indican que el tiempo necesario para desarrollar aplicaciones pequeñas y medianas se reduce, al igual que la cantidad de análisis y diseño.
3. el uso de T4G para grandes trabajos de desarrollo de software exigen el mismo tiempo o más de análisis, diseño y prueba, perdiéndose el tiempo ahorrado en la codificación.

### 5.3 Fases de la ingeniería de software

Los paradigmas de la ingeniería de software difieren entre si en diferentes aspectos pero los cuatro contemplan tres fases:

Fase de definición: Comienza con la planificación del software e incluye:

- Una descripción de las necesidades.
- Análisis de riesgos.
- Definición de los recursos necesarios.
- Estimación de tiempo y costo

Fase de desarrollo: Traduce el conjunto de requisitos al programa en si. Las etapas que contempla son:

- Diseño.
- Codificación

Fase de verificación, lanzamiento y mantenimiento: Es la prueba del programa para encontrar el mayor número de errores posibles antes de que sea puesto en ejecución, la verificación lo prepara para su lanzamiento, el mantenimiento asegura la vida útil del sistema a pesar de los cambios en las necesidades y el medio ambiente.

- Actividades de verificación
- Actividades de validación

(Ibid.:150).

#### **5.4 Análisis del sistema**

El análisis de los sistemas se realiza para lograr los siguientes objetivos:

- Identificar las necesidades del cliente
- Evaluar la viabilidad del sistema
- Realizar un análisis técnico y económico
- Asignar funciones al programa, al hardware, a la gente, a la base de datos y demás componentes del sistema.
- Establecer restricciones de costo y de tiempo
- Crear una definición del sistema para poder aplicar ingeniería en él.

Es muy difícil determinar cuanto esfuerzo se debe aplicar en el análisis y definición de un sistema, debido a que siempre será posible mejorar lo obtenido. Pero a manera general se pueden tomar en cuenta el tamaño del sistema, su complejidad, área de aplicación y el uso que se le dará. Una regla propuesta por Pressman dicta que se debe emplear un 10% al análisis y

un 10 % a la ingeniería. Que son aproximadamente el tiempo que dedique en el proyecto de esta tesis.

A la hora de realizar el análisis se debe tener contacto con el personal técnico y administrativo.

#### **5.4.1 Pasos para el análisis de los sistemas**

1. **Identificación de las necesidades:** Esto se logra mediante la entrevista con el cliente o su representante, es el punto de partida del desarrollo del sistema y su objetivo es hacer una distinción entre lo que el cliente quiere y lo que necesita realmente. Deben identificarse todos los objetivos y cuestionarse si se cuenta con los recursos necesarios para desarrollar el sistema, tanto económicos, tecnológicos como de personal. Si el producto será desarrollado para la venta es necesario también hacer un estudio de mercado para asegurarse que no será un fracaso una vez terminado. Esta información se recopila en un documento de conceptos del sistema.
2. **Estudio de viabilidad:** Este análisis se realiza para conocer si será posible desarrollar el sistema con los recursos disponibles. El análisis de viabilidad debe contemplar también un análisis de riesgo. El análisis de viabilidad debe aplicarse lo más pronto posible, pues evita gastos y esfuerzo innecesarios. Es necesario establecer la viabilidad económica para determinar si el rendimiento final absorberá el costo de desarrollo del sistema. La viabilidad legal asegurará que el desarrollo no quebranta ninguna ley. Por último deben tomarse en cuenta las alternativas existentes para justificar la elección del desarrollo.

Pressman estipula que no es necesario llevar a cabo un estudio de viabilidad para sistemas en los que la justificación económica es obvia, el riesgo técnico es bajo, se esperan pocos problemas legales y no existe otra alternativa razonable (Ibíd.:157).

3. **Análisis económico:** es el análisis costo-beneficio que se aplica al sistema a desarrollar. Se deben comprar los costos de desarrollo del proyecto contra los beneficios tangibles (dinero) e intangibles (confiabilidad) del sistema.

Los beneficios siempre deben compararse con el sistema que será reemplazado. El análisis puede variar dependiendo de las características del sistema, pero generalmente se debe tomar en cuenta que los sistemas de gestión de datos desarrollados, como el proyecto de esta tesis, deben mejorar la calidad, cantidad, rapidez y organización de la información (Ibíd.:159).

## **5.5 Análisis Orientado a Objetos**

El análisis orientado (AOO) se basa en conceptos que todos aprendimos cuando niños: Objetos y atributos, clases y miembros.

Para identificar los objetos puede uno basarse en la descripción del problema, llevando a cabo un análisis del texto. Se extraen cada nombre o cláusula nominal y se ponen en una tabla. Se deben anotar todos los sinónimos.

Los objetos pueden ser:

1. Entidades externas: otros sistemas, dispositivos, usuarios, etc. Que producen o hacen uso de la información del sistema. En este caso pueden ser los usuarios a los que se entrega el reporte.
2. Cosas: forman parte de la información del sistema, como los reportes.

3. Papeles: Los papeles que toman las personas que interactúan con el sistema:.
4. Unidades organizativas: divisiones, grupos o equipos.
5. Lugares: Laboratorios, oficinas donde se aplica el sistema, etc.
6. Estructuras: Definen las clases de los objetos

(Ibid.:54-255).

Para poder aceptar cada uno de los objetos debe cumplir con todos o con la mayoría de los siguientes requisitos:

1. Información retenida: El objeto potencial será útil durante el análisis sólo si la información sobre el mismo debe recordarse para que pueda funcionar el sistema.
2. Servicios necesarios: El objeto potencial debe tener un conjunto de operaciones identificables que pueden cambiar el valor de sus atributos.
3. Múltiples atributos: Durante el análisis de requisitos se debe centrar la atención en la información principal, los objetos con un sólo atributo pueden ser declarados durante el análisis pero son mejor representados como atributos de otro objeto.
4. Atributos comunes: Se pueden definir conjuntos de atributos para los objetos potenciales y así aplicar esos atributos a todas las ocurrencias del objeto.
5. Operaciones comunes: Se pueden definir conjuntos de operaciones para los objetos potenciales y aplicar así esas operaciones a todas las ocurrencias del objeto.
6. Requisitos esenciales: En el modelo de requisitos casi siempre se definirán como objetos las entidades externas que aparecen en el espacio del problema y que producen o consumen información esencial para el funcionamiento de cualquier solución que se desarrolle para el sistema



### 5.5.1 Modelización de los datos:

La modelización de los datos se considera generalmente como un análisis complementario.

Se concentra en los datos y consiste en una representación de la red de datos del sistema. A diferencia del enfoque de análisis estructurado, la modelización de los datos considera los datos independientemente del procesamiento que transforma los datos.

La modelización de los datos no se enfoca en el procesamiento de los datos. Es utilizada ampliamente en aplicaciones de bases de datos, como el sistema del este proyecto de tesis,. Proporciona al analista y al diseñador de la base de datos una amplia visión de los datos y de las relaciones que gobiernan los datos (Ibíd.270).

Un objeto de datos no toma la misma forma que un objeto regular. Sólo encapsula los datos (dentro de un objeto de datos no hay referencia de las operaciones que se hacen a los datos).

Por lo tanto el objeto de datos puede ser representado como un atabla.

Los atributos de los objetos de datos los caracterizan tres formas diferentes.

1. Se utilizan para dar el nombre a una instancia del objeto de datos.
2. Describir la instancia.
3. Hacer referencia a otra instancia de otra tabla (Ibíd.:271).

También se puede definir uno o más atributos como identificador o clave que puede servir para identificar una instancia del objeto de datos. Para tener un adecuado control de las instancias debe contarse con al menos identificador único.

En la tabla de ejemplo del objeto de datos automóvil, tenemos los siguientes atributos:

- Identificador debe ser [placa] pues es un atributo que no se repetirá en dos instancias de ese objeto de datos
- [ID de propietario] hace referencia al atributo de otra instancia de otra tabla.
- [Marca]y [Modelo] describen la instancia

Tabla del objeto de datos automóvil

| Marca | Modelo | Placa   | Número de Propietario |
|-------|--------|---------|-----------------------|
| Ford  | 1999   | TFG7654 | 7762662222            |
| GMC   | 1989   | 56FGTR5 | 9178277722            |

Con el objeto de reducir al mínimo la redundancia (minimizar la cantidad de información necesaria para solucionar un problema) las tablas de los objetos de datos pueden ser formalizadas aplicando un conjunto de reglas de normalización que dan como resultado un modelo relacional de los datos.

Las reglas de normalización son:

1. Una instancia dada de un objeto de datos tiene un valor y Sólo uno para cada atributo: Esta regla requiere que en la tabla del automóvil (Tabla superior) Sólo exista un propietario por cada instancia de automóvil. Si el GMC tuviera dos propietarios se tendría que rediseñar el objeto de datos. Adicionalmente se asume que la tabla tiene una entrada por cada atributo. No debe haber “huecos” en la tabla. Esta regla ayuda a definir una relación.
2. Los atributos representan elementos de datos elementales: No contienen estructuras internas. Cada atributo debe ser indivisible, es decir, volviendo al

ejemplo de la tabla superior, si en el atributo de número de propietario se almacenara además el nombre, esta regla se rompería. Para resolver una situación en la que un atributo fuera un elemento de datos compuestos, se debe definir la tabla de forma que cada uno de los componentes este separado formando un atributo nuevo (Ibíd.;272).

3. Cuando se usa más de un atributo para identificar un objeto de datos, hay que asegurarse que los atributos descriptivos y referenciales representen una característica del objeto completo y no una característica de algo que pueda ser identificado Sólo por una parte del identificador. Un ejemplo es que, supongamos que añadimos un nuevo identificador a la tabla de los automóviles [venta]... y si continuamos modificando la tabla (descripción del objeto de datos) y añadimos lugar de venta como atributo descriptivo, se rompe la regla 3, debido a que el lugar de venta es un atributo de uno Sólo de los identificadores [venta]. No es un atributo del número de propietario, que sería el otro identificador.

Si se diseñan las tablas de acuerdo con estas tres reglas se dice que el modelo de datos resultante está en la segunda forma normal.

4. Todos Los atributos que no sean identificadores deben representar alguna característica de la instancia nombrada por el identificador del objeto de datos y describir algún otro atributo que no sea un identificador, por ejemplo si se añade un atributo [nombre de pintura] al objeto de datos de la tabla de

automóviles, se violaría la regla 4. [nombre de pintura] es una característica de color, no de la instancia del automóvil, identificada por el número de propietario (Ibid.273).

Como conclusión, podemos ver en este capítulo, el ciclo de vida de los sistemas no es cumplido siempre siguiendo el orden absoluto que expresa, y más bien se adapta su seguimiento conforme se desarrolla el sistema, en el caso particular de este proyecto de tesis, yo creo que la implementación de un prototipo fue una medida exitosa, a tal grado que se pudo comenzar a trabajar con el prototipo del sistema mientras se desarrollaban las mejoras, conforme su utilización surgieron nuevos requerimientos y necesidades que de otra forma no podrían haberse detectado sino hasta finalizado el sistema.

Al analizar la forma de construir sistemas que he implementado en estas últimas etapas de mi carrera descubro con sorpresa que siempre hay una inclinación a la creación de prototipos y que esto me ha funcionado muy bien a la hora de satisfacer necesidades y requerimientos surgidos durante el desarrollo.

Sin embargo es necesario recalcar que el ciclo de vida de los sistemas debe servir siempre como pauta a la hora de construir un sistema, ya que permite tomar en cuenta todos los aspectos necesarios para obtener un sistema completo y funcional.

En este proyecto utilizo el paradigma de prototipos. el paradigma de técnicas de cuarta generación me parece muy interesante y quizá podría encontrar su aplicación en este sistema de evaluación que se va a desarrollar. lamentablemente no cuento ni con los programas requeridos ni con los conocimientos necesarios para manejarlos en caso de conseguirlos, quedarán, pues, para futuros desarrollos en caso de poderlos conseguir.

También mencionaré que la fase de verificación fue una de las fases más importante en el desarrollo, porque permitió no solamente encontrar problemas en el programa sino que además sacó a la luz nuevos requerimientos no solamente internos (más espacio en ciertos campos) sino que además cambios provocados por la base sobre la que sería instalado (red) y acceso a los datos del sistema (de forma distribuida).

## Capítulo VI

---

### Estudio de caso del Sistema de evaluación del desempeño de los profesores

#### 6.1 Antecedentes de la Universidad Don Vasco A.C.

Fundada en 1963, la Universidad Don Vasco es la primera institución de educación superior en el estado, actualmente ofrece 9 licenciaturas, estudios de postgrado y un programa de bachillerato incorporados a la Universidad Nacional Autónoma de México, así como una secundaria (SEP)

Misión: “Ser una institución de educación forjadora de personas con una inquebrantable robustez de espíritu, formadora de jóvenes responsables, generosos, críticos y creativos; conscientes de su trascendencia en cuanto a su origen y destino y reconstructores de la sociedad desde una amplia y mayor perspectiva”

Filosofía: “La Universidad Don Vasco se sustenta en los principios de integración y superación. Entiende por integración la unidad de mentes y esfuerzos de todos los integrantes de la institución, para lograr una mejor formación. La superación implica la búsqueda constante de nuevos y mejores horizontes para lograr una vida más completa y digna”

Propósitos:

- Promover en la juventud un verdadero espíritu de estudio, investigación y profesionalismo.
- Proporcionar a los jóvenes una formación integral que los comprometa moralmente.
- Dar oportunidad a todas las clases sociales de obtener una formación a la altura de los mejores centros educativos de la nación.

- Responder a las necesidades de la región.

La Escuela Superior de Administración, Contaduría e Informática (E.A.C.I.) de la Universidad Don Vasco A.C. engloba a estas tres licenciaturas en las cuales se cuenta con 22 grupos de alumnos, se imparten 77 materias y laboran 52 profesores. Todos los profesores tienen licenciatura o ingeniería relacionada con las materias que imparten a los alumnos, de tal forma que tienen conocimientos bien cimentados en el tema. La E.A.C.I cuenta con aproximadamente 600 alumnos.

## **6.2 Metodología utilizada**

La metodología utilizada para la obtención de información fue la entrevista directa con los encargados de elaborar, aplicar y analizar las evaluaciones del desempeño de los profesores. Se hizo también una investigación documental para conocer las evaluaciones, el puntaje y la forma en que se aplicaban las evaluaciones. También se examinaron las técnicas de obtención de resultados y la valoración de las respuestas.

1. Se analizaron los problemas y oportunidades existentes en torno a la evaluación del desempeño de los profesores.
2. Se aplicó un cuestionario a los alumnos de la universidad para saber su opinión acerca de las evaluaciones actuales que se realizan a los profesores. El resultado de este cuestionario también serviría para saber la opinión respecto a aplicar las evaluaciones en las computadoras. Los resultados indicaron que todos estaban de acuerdo en que se aplicara en computadora.

3. Se aplicó un cuestionario a los profesores para saber su opinión a cerca de las evaluaciones que se les hacían. En general los resultados les ayudan a conocer sus defectos y mejorar en su desempeño docente.

### **6.3 Procedimientos actuales de evaluación**

Actualmente el desempeño de los profesores se califica mediante la aplicación de un cuestionario que es contestado por los alumnos del profesor a evaluar. Dicho cuestionario ha sido estructurado en dos partes principales, la primera parte tiene como fin evaluar al alumno con preguntas acerca de su desempeño en general, la segunda parte se enfoca a la parte del maestro y su desempeño profesional en clase.

Los procedimientos llevados a cabo para evaluar a los profesores son:

- **Diseño del cuestionario:** Se reúne un grupo de asesores pedagógicos de la E.A.C.I. y analizan y seleccionan todos los aspectos referentes a la evaluación y los datos que desean recabar. Luego arman el cuestionario con asesoría de personal de informática. El cuestionario consta de preguntas con opciones del 1 al 8, 1 es la mínima calificación, 8 es la más alta. Sólo se debe seleccionar una opción y cada una de las preguntas es obligatoria, no debe dejarse en blanco ninguna.
- **Inducción a los alumnos:** Antes de aplicar la evaluación se explica a los alumnos el objetivo y la importancia del cuestionario, el cual arrojará datos que pueden mejorar tanto el desempeño de los alumnos como del profesor. Se explica también la manera de contestar el cuestionario.



- Aplicación del cuestionario. Se aplica el cuestionario por grupo bajo la supervisión de algún coordinador. El tiempo de aplicación de cada cuestionario (un cuestionario por maestro) varía de 5 a 8 minutos. Y se realiza en el salón de clase del grupo seleccionado.
- Captura de la información: La información recopilada en cuestionarios es pasada a tablas y listas para su clasificación. Para esto se utiliza la hoja de cálculo
- Procesamiento de la información: Una vez que se tiene la información capturada en tablas se procede a aplicarle ciertas fórmulas, principalmente estadísticas, para poder sacar las conclusiones. Se obtienen promedios generales de todos los cuestionarios por materia y promedios generales por reactivo. Después se hacen las gráficas de los promedios por materia y promedios por reactivo. Se muestran por separado los resultados de los reactivos referidos a maestros y alumnos.
- Generación de tablas y gráficas: Los datos son convertidos a gráficas para comparar el desempeño entre las materias que imparte un profesor y entre los mismos profesores. Se hacen gráficas de promedios por materia y gráficas de promedios por reactivo.
- Presentación de resultados impresos a los profesores: Se les dan los resultados a los profesores para que evalúen sus aspectos desfavorables y hagan hincapié en sus aspectos favorables. Los resultados de maestros y alumnos se muestran por separado. Algunas de las decisiones que se toman basándose en los resultados son:
  - a) Estrategias a implementar con cada maestro para mejorar su práctica docente.
  - b) Apoyos pedagógicos y didácticos que requieren los profesores como son recursos materiales, formación, etc.

- c) Decisiones de tipo administrativo que pueden ser:
  - a. Asignación de materias específicas en las que el profesor se desempeña de mejor manera.
  - b. Permanencia o despido de un profesor.

#### **6.4 Identificación de los requerimientos**

Es en esta etapa del desarrollo del sistema en que se deben encontrar los elementos necesarios para justificar la necesidad del sistema, de lo contrario su elaboración podría ser una pérdida de tiempo y de recursos.

En el anexo puede encontrarse un documento en el que se establece la propuesta de evaluación en del desempeño docente de los profesores en la E.A.C.I. (Escuela de Administración Contaduría e Informática)

Todo proyecto de sistemas inicia por varios motivos los cuales se presentan conforme evoluciona la organización y pueden ser la solución a un problema o una oportunidad de mejora. Después de haberse seleccionado un determinado proyecto a realizar debe procederse a probar su factibilidad.(KENDALL, 1997:47).

#### **6.5 Descripción del problema**

Desde hace tres semestres, la E.A.C.I. aplica un cuestionario a los alumnos con la finalidad de conocer la opinión que tienen del desempeño de los profesores y de los ellos

mismos, cada alumno contesta un cuestionario por cada materia impartida. La evaluación se divide en tres secciones, la primera sección está encaminada a evaluar el desempeño del alumno en 8 preguntas que incluyen aspectos como asistencia, tiempo de estudio, etc. La segunda sección consta de 14 preguntas dirigidas a evaluar las funciones del profesor como puntualidad, forma de calificar etc. La tercera y última parte del cuestionario son 2 preguntas abiertas en las cuales el alumno puede explicar de manera más abierta sus opiniones y consideraciones.

A excepción de las últimas 2 preguntas que son abiertas, las primeras 22 se contestan seleccionando una opción dada en un rango de 1 a 8, donde 1 es la puntuación más baja y 8 es la máxima puntuación. (Para más detalles acerca de las preguntas de la evaluación consulte el anexo 5b al final de los capítulos)

Algunas reglas a la hora de calificar los cuestionarios son:

- Ninguna pregunta puede tener más de una opción contestada, cuando una pregunta tiene dos opciones, se selecciona la opción con mayor calificación
- Todas las preguntas deben estar contestadas, si una pregunta no lo está se selecciona un valor medio, si hay más de 4 opciones sin contestar se descarta la evaluación.
- Las preguntas abiertas deben contener aspectos significativos y que no contengan lenguaje impropio.

La aplicación de la evaluación se lleva a cabo pasando a los grupos respectivos a cada semestre y de cada carrera.

Se entrega un número de cuestionarios a cada alumno igual al número de materias que cursa en ese semestre. Cada evaluación contiene además de las preguntas, el nombre del profesor y la materia que imparte en ese grupo.

En promedio los alumnos tardan de 7 a 10 minutos en contestar cada cuestionario, y dependiendo de las materias del grupo, la aplicación total de la evaluación puede durar hasta una hora.

Una vez aplicados los cuestionarios a todos los grupos, se procede a vaciarlos en una hoja de cálculo para su clasificación.

Cabe mencionar que es durante este vaciado cuando se verifica que los cuestionarios estén completamente contestados y el contenido de las respuestas abiertas. Es en esta etapa donde puede cometerse errores de captura que pueden viciar los resultados finales.

Una vez capturados todos los datos, se procede a cuantificarlos y los reportes resultantes son:

- Gráfica de Promedio de cada profesor, organizados por carrera. Impresa.
- Gráfica de Promedio de cada pregunta, organizados por carrera. Impresa.
- Gráfica de Promedio general de cada materia, organizadas por carrera. Impresa.
- Tabla de Promedios generales de cada materia impartida por cada profesor. (Este reporte es un compendio de los promedios generales). Impresa.
- Respuestas abiertas. Se capturan de nuevo en limpio. Impresas.

Objetivos que debe cumplir el sistema:

- Ejecución múltiple
- Almacenamiento de las evaluaciones a aplicar
- Almacenamiento de las respuestas a las evaluaciones
- Verificar que todas las respuestas sean contestadas
- Realizar los cálculos y procesos necesarios para la obtención de los resultados
- Presentar reportes de resultados
  - Promedio profesor por carrera
  - Promedio de pregunta por carrera
  - Promedio general de materias por carrera
  - Promedios generales de profesores
  - Promedio general de materias
  - Promedio general de preguntas
  - Tabla general
- Debe ser aplicable en las instalaciones de la Universidad Don Vasco

El límite de entrega del sistema es el 15 de Mayo de 2000, fecha en la cual se inicia el periodo de evaluaciones, esta fecha está considerada en la aplicación del anterior sistema manual de evaluación. Aplicándolo después de esta fecha no garantiza la obtención de los resultados antes de terminar el ciclo escolar.

## 6.6 Estudio de viabilidad

Desde luego que existe viabilidad en el desarrollo del sistema de evaluación del desempeño del profesor, el costo de desarrollo del sistema es nulo para la universidad debido a que es un proyecto de tesis, personalmente no tuve que realizar ninguna erogación en especial para poder desarrollar el sistema debido a que contaba con los recursos técnicos (computadora y programas) para realizar el proyecto. Se cuenta con la infraestructura necesaria para la ejecución del sistema, la cual está representada por los laboratorios de la Universidad Don Vasco, los cuales deben cumplir con los requisitos del sistema para poder ejecutarse de manera correcta. No se puede comprobar su total funcionalidad hasta que se realizan las pruebas de los prototipos, pero al menos cuentan ya con los requerimientos mínimos necesarios.

Actualmente en la universidad no se aplica ningún sistema informático para llevar a cabo esta tarea, y no tengo conocimiento de que exista en el mercado ningún sistema que satisfaga los requisitos particulares. Existe, no obstante, un programa desarrollado por el encargado de los laboratorios, que puede utilizarse para la aplicación de las evaluaciones. Pero hasta ahora ha sido utilizado para obtener los reportes generales de las evaluaciones, no puede ser utilizado para obtener datos de bases de datos externas.

Viabilidad legal: El sistema desarrollado no infringe ninguna ley, la universidad cuenta con una licencia del lenguaje de programación Visual Basic Ver.6 de Microsoft. Así como del manejador de la base de datos, el cual viene incluido con el lenguaje de programación. El

materia gráfico fue creado para este proyecto y el resto de los componentes del sistema son de dominio público (applets e iconos)

Problemas encontrados en el estudio de viabilidad:

- Tiempo de entrega demasiado corto.
- Capacidad de los laboratorios limitada, debido a que no puede aplicarse la evaluación a grupos numerosos en un mismo laboratorio.

### 6.7 Análisis económico

Hay un ahorro significativo en recursos mediante la utilización del sistema y sólo se mantienen los siguientes costos, los cuales se minimizan al ser necesario menos personal para aplicar la evaluación con el sistema:

- Costo del personal encargado de aplicar la evaluación (1 persona)
- Costo del personal encargado de obtener los reportes del sistema (1 persona)
- Costo de impresiones de reportes (Tinta y papel)

Se ha descuido, sin embargo incluir un listado de los elementos utilizados en el desarrollo, para dar un ejemplo de todo lo que se necesita para poder crear un sistema de este tipo y sirva como referencia para otros desarrolladores.

El costo de desarrollo del sistema:

| Componentes   | Costo aproximado* |
|---|-------------------|
| Sistema de cómputo para el desarrollo (Incluye memoria, procesador, teclado, mouse, monitor, disco duro, drives y demás componentes indispensables) | \$15,000          |

|   |                        |
|---|------------------------|
| Regulador                                   | \$750                  |
| Escáner cama plana                          | \$2,500                |
| Multicontacto                               | \$100                  |
| Impresora:                                  | \$1,000                |
| Cartucho de tinta:                          | \$350                  |
| Sistema Operativo:                          |                        |
| Windows 98                                  | \$1,500                |
| Software Adicional:                         |                        |
| Office 2000                                 | \$4,500                |
| Microsoft Visual Basic Ver. 6 profesional   | \$6,000                |
| Paint Shop Pro                              | \$800                  |
| FIRE Works                                  | \$1,800                |
| Horas de Trabajo                            |                        |
| Horas de programación:                      | \$28000                |
| 80 hrs.(350 * hora)                         |                        |
| Horas de Instalación(2 hr.)(100*hr.)        | \$200                  |
| Horas de Capacitación(10hr.)(100*hr.)       | \$1000                 |
| Horas de asesoría(10hr.)(100*hr.)           | \$1000                 |
| Instalación                                 |                        |
| Diskettes de instalación                    | \$200                  |
| CD de instalación                           | \$30                   |
| Grabación de CD de instalación              | \$200                  |
| <b>Papelería:</b>                           |                        |
| Hojas para impresiones                      | \$1000                 |
| Ampliaciones, reducciones, fotocopias, etc. | \$1000                 |
| Lápices                                     | \$10                   |
| Lapiceros                                   | \$10                   |
| Gomas / líquido corrector                   | \$15                   |
| Reglas                                      | \$10                   |
| Renta de la oficina ( 2 mes )               | \$5000                 |
| <b>Muebles</b>                              |                        |
| Escritorio                                  | \$1000                 |
| Silla                                       | \$500                  |
| Lámparas                                    | \$200                  |
| Enseres de limpieza                         | \$100                  |
| Utensilios de Cocina                        | \$500                  |
| Gas   | \$150                  |
| Agua  | \$500                  |
| Energía eléctrica                           | \$100                  |
| Teléfono                                    | \$50                   |
| Gasolina                                    | \$200                  |
| Total:                                      | \$75275 (7,527.5 Dlls) |

*\*Los precios están dados en moneda nacional.*

Equipo de cómputo recomendado para la ejecución:



Minimo:

- Computadora PC o compatible
- Disco Duro de 500 Mb.
- 20Mb. En Disco duro libres ( esto puede variar de acuerdo a la instalación )
- 32 Mb en RAM
- Procesador Pentium 200Mhz
- Drive de 3.5 ( disquetera )
- Teclado español
- Ratón
- Monitor VGA
- Resolución de 480 X 640
- Sistema Operativo Windows 9x /Windows NT

Costo aproximado: \$8,000 (800 Dlls)

Recomendado:

- Computadora PC o compatible
- Procesador Pentium II 450 Mhz
- Disco Duro de 2. Gb.
- 40Mb. En Disco duro libres ( esto puede variar de acuerdo a la instalación )
- 128 Mb en RAM
- CD ROM 32x

- Teclado español
- Ratón
- Monitor VGA
- Resolución de 800 X 600
- Sistema Operativo Windows 98 Segunda edición/ Windows NT

Costo Aproximado:\$10,000 (1000 Dlls)

Beneficios del sistema desarrollado:

#### Personal

- Reducción del personal necesario para aplicar y obtener los resultados de las evaluaciones (1 persona es suficiente)
- Cualquier persona con una mínima capacitación puede manejar el sistema.
- Facilidad de uso del sistema en la respuesta a la evaluación.
- Menor tiempo de trabajo de personal para la obtención de los resultados.

#### Cálculo e impresión

- Reducción en el costo de impresiones, no se tiene que imprimir las evaluaciones, ni es necesario el uso de estencil, etc.
- Mejora en la exactitud de las tareas de cálculo, eliminación de los errores.
- Cambio de las variables (preguntas) y los valores de cálculo.
- Visualización información de manera global o individual
- Cálculo automático de las operaciones.
- Menor tiempo de cálculo y manejo automático de errores en los datos de entrada.

- Reportes a la medida

#### Registros

- Entradas normalizadas y estandarizadas de datos.
- Almacenamiento de los registros en medios magnéticos que ocupan poco espacio físico
- Recuperación de la información desde bases de datos externas al sistema.
- Soporte y manejo de datos de entrada incompletos. se calculan los resultados por el número de respuestas y no por evaluaciones. (No afecta el calculo de los resultados si el sistema se cae durante una evaluación)
- Mantenimiento a los registros fácilmente.
- Aumento en la capacidad de registros y poco costo de almacenamiento.
- Seguridad en el almacenamiento de los registros mediante el uso de tres bases de datos separadas de acuerdo a la importancia, Profesores, respuestas y evaluaciones, la infracción en cada una de ellas no afecta el desempeño de las demás. Sólo se puede almacenar información con la disponibilidad de las tres.
- Formato de registros portable a los principales programas comerciales.
- Estandarización de los registros.
- Respaldos y recuperación de información de los respaldos
- Captura de datos en varios sistemas de cómputo sin red, o aplicar la evaluación en redes separadas.

#### Búsqueda de información

- Las respuestas abiertas son legibles.

- Obtención de información automática
- Reportes automáticos
- Modificación de registros principales, datos de profesores, materias, evaluaciones, etc.
- Extracción y consulta de datos desde fuentes externas al sistema, se pueden guardar registros en diferentes lugares y luego extraer la información a la base de datos principal.

#### Versatilidad

- Añadir, eliminar y consultar y modificar las evaluaciones
- Añadir, eliminar y consultar y modificar las preguntas de las evaluaciones
- Añadir, eliminar y consultar y modificar el valor de puntuación de las preguntas
- Utilización del sistema para cualquier tipo de evaluación. (Siempre que contenga preguntas de opciones)
- Aplicación de manera simultánea varias evaluaciones.
- Estructura de base de datos abierta a agregar datos del evaluador para calificarlo.(se utilizaría en evaluaciones aplicadas a los alumnos)
- Menor tiempo requerido para aplicar la evaluación (especialmente cuando se aplica a alumnos con experiencia en el manejo de las computadoras)

#### Encuestado

- Le permite contestar las respuestas en orden y evita que deje alguna sin contestar o marque dos respuestas a una misma
- No requiere de ningún material para contestarlo (lápiz o lapicero)

- Conserva su anonimato de manera más segura. Podría llegar a identificarse un alumno por las características de su escritura.
- Interfase visual para facilitar la utilización del sistema.
- Fácil y rápido aprendizaje para contestar las respuestas.

Desventajas del sistema desarrollado:

- Necesidad de instalaciones de cómputo para la aplicación del sistema (red)
- Necesidad de personal encargado del sistema
- Necesidad de características del equipo de cómputo para aplicar la evaluación (Hardware y software)
- Necesidad de almacenamiento magnético

Los siguientes costos son erogados en la aplicación de las evaluaciones utilizando el sistema manual

Costos del sistema actual por aplicación:

|   |                  |
|---|------------------|
| 3600 Hojas de papel (600 alumnos, 6 evaluaciones por alumno) (\$.1 c/u) | \$360            |
| 360 Hojas de papel de reportes (120 materias, 3 por materia) (\$.1c/u)  | \$36             |
| Esténcil  | \$30             |
| 120 Fotocopias (\$.5 c/u)   | <u>\$600</u>     |
|   | Subtotal: \$1026 |
| Número de personas que trabajan en la aplicación de las evaluaciones    | 2                |
| Tiempo en realizar las evaluaciones (1 hora por grupo)                  | 21               |
| Sueldo por hora del personal  | <u>\$40</u>      |

|  |                 |
|--|-----------------|
|  | Subtotal \$1680 |
| Número de personas encargadas de pasar la información a la computadora | 2               |
| Tiempo en realizar la captura de información                           | 20              |
| Sueldo por hora del personal   | <u>\$40</u>     |
|  | Subtotal \$1600 |
| Número de personas encargadas de obtener los reportes                  | 2               |
| Tiempo en obtención de los reportes                                    | 10              |
| Sueldo por hora del personal   | <u>\$40</u>     |
|  | Subtotal \$800  |
| Total de gastos en las evaluaciones                                    | \$5106          |

Beneficios del sistema actual:

- No se necesitan requerimientos especiales de instalaciones para la aplicación de la evaluación
- No es necesario
- Se mantienen las evaluaciones en papel para referencias futuras

Desventajas del sistema actual:

- Costo de personal para aplicar las evaluaciones y obtención de resultados
- Proceso manual
- Costo elevado de papelería
- Tiempo de clasificación y procesamiento de los datos.

- Errores en la clasificación de los datos
- Errores en la captura de los datos
- Espacio de almacenamiento físico
- Personal con conocimientos especializados para la obtención de los resultados

### **Comparación de los costos de ambos sistemas:**

El costo del sistema desarrollado no representa ninguna erogación para la universidad, los costos que se mantienen fijos son los siguientes:

- Hojas de papel de reportes (120 materias, 3 por materia) (\$.1c/u) : \$36
- Sueldo del personal por aplicar la evaluación: 1 persona, 21 horas, \$40 la hora) \$840
- Sueldo del personal por imprimir reportes: 1 persona, 10 horas, \$40 la hora) \$400

Total de costo de aplicación y obtención de resultados en el nuevo sistema: \$1276 (127.6 Dlls)

Total de costo de aplicación y obtención de resultados en el sistema anterior: \$5106 (510.6 Dlls)

Ahorro económico mediante la utilización del sistema: 3830 (383 Dlls)

Total de horas de trabajo personal con el sistema manual : 102 horas

Total de horas de trabajo personal con el nuevo sistema: 31

Ahorro en horas de personal : 71

## 6.8 Definiendo los límites del sistema

El sistema de evaluación es una aplicación de complejidad media, la cual debe ser capaz de seleccionar los datos de la evaluación, aplicarla y después obtener los resultados. Es importante recalcar que también debe ser capaz de manejar situaciones en las que falla el sistema, de tal forma que una caída en la red no impida su ejecución.

Parte del reto de desarrollar el sistema es analizar los procedimientos actuales con que se realiza la evaluación e introducir mejoras. En este aspecto no hubo oportunidad de mejoras significativas en cuanto a los procedimientos, los datos tomados en cuenta para la evaluación habían sido seleccionados cuidadosamente por el personal encargado de la evaluación para agilizar los procedimientos, los cuales se llevan a cabo de manera manual.

Inicialmente se pensó en realizar un sistema hecho a la medida que cumpliera exactamente con los requisitos propuestos, pero también se estudió la posibilidad de dar versatilidad en uso al sistema de tal forma que no solamente fuera aplicable a la evaluación del desempeño de los profesores, sino que pudiera ser fácilmente aplicada en otras evaluaciones, permitiendo agregar nuevas preguntas o modificar y eliminar las actuales, capacidad de crear nuevas evaluaciones y combinar preguntas abiertas con preguntas de opciones de tal forma que los resultados no se vieran afectados (Hay que tomar en cuenta que las preguntas abiertas no pueden ser calificadas por el sistema, solamente las de opciones basándose en el valor asignado de manera individual). Por lo anterior el sistema aumentó en complejidad pero al mismo tiempo en versatilidad y eficiencia, pues extendería su vida útil aún cuando la estructura, contenido y valor de la evaluación cambiara. También se pensó en dejar abierta la



posibilidad de obtener datos del alumno o persona que resolviera la evaluación para poder calificar sus respuestas.

El sistema también debe ser capaz de ejecutarse en un ambiente de red para poder aplicar la evaluación a grupos enteros de alumnos, pero debe incluir la posibilidad de ejecutar de manera individual cada evaluación y luego hacer posible la extracción de los datos a una base de datos primaria. Esto resulta muy útil cuando las comunicaciones de red tienen una caída o cuando se debe aplicar la evaluación en redes separadas.

El análisis del sistema plantea las siguientes actividades como las más importantes:

- Creación de evaluaciones y asignación de preguntas: Incluye asignación de opciones y el valor de cada una de éstas.
- Asignación de profesores y materias a cada grupo: Incluye agregar profesores, asignación de grupos a semestres y de semestres a carrera. Incluye asignación de materias a cada semestre de carrera y luego dichas materias a profesores. También se asigna cada profesor a cada grupo.
- Aplicación de la evaluación: Incluye módulo para contestar la evaluación
- Presentación de reportes de resultados: Incluye gráficas e impresión de tabla general.

El sistema es desarrollado teniendo en mente su aplicación en red, aunque bien puede utilizarse en monousuario, las capacidades multiusuario permiten que se apliquen las evaluaciones a grupos enteros, en este aspecto se depende de la infraestructura de la universidad, así como de la capacidad de los laboratorios.

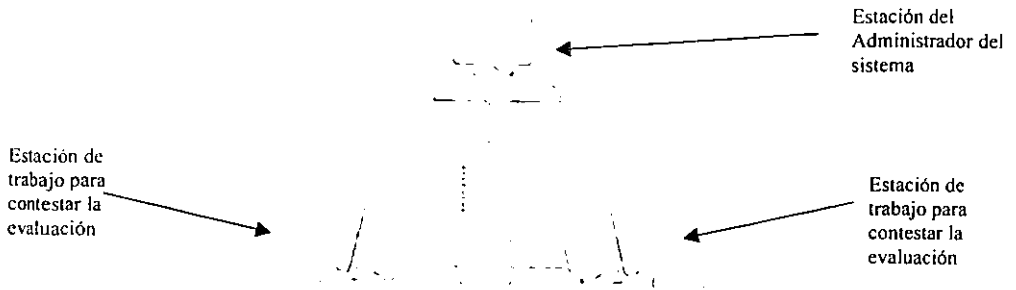


Fig. 5.1 Red para la evaluación del desempeño de los profesores

En la fig. 5.1 se puede ver que el sistema está compuesto por los siguientes elementos:

- Estación del administrador: en esta se ejecuta el módulo principal del sistema de evaluación del desempeño, es manejada por el encargado del sistema de evaluaciones, en este módulo del sistema, además de cargar las variables necesarias para la ejecución, se estipula la evaluación que se ejecutará. Es en este módulo donde se tienen acceso a los reportes de resultados y a las funciones principales del sistema como extracción de datos externos y respaldos.
- Estación de trabajo: en ella se instala solamente el módulo de evaluación, en el cual los alumnos contestan la evaluación asignada.
- Servidor: en el se guardan y recuperan los datos del sistema.

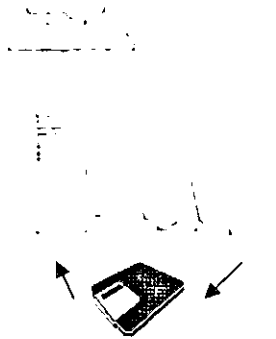


Fig. 5.2 Caída del sistema

En caso de una caída en las comunicaciones del sistema, las computadoras podrían trabajar de manera individual, almacenando los datos localmente, los cuales después podrían pasarse a la base de datos central mediante medios magnéticos o al reestablecerse las comunicaciones, de esta forma se garantiza la disponibilidad del sistema aún en condiciones adversas. El módulo principal es capaz de extraer datos externos, como los provistos en estas ocasiones.

## 6.9 Escenarios

Ahora que se ha delimitado el alcance del sistema se procede al estudio de los diferentes escenarios que se presentan en el sistema:

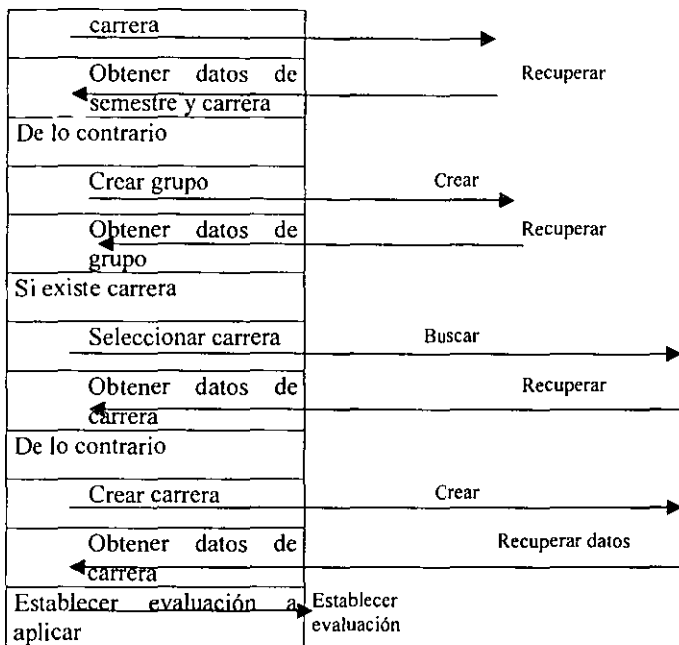
- El administrador establece una evaluación
- Se asignan grupos a semestres
- Se asignan grupos de semestres a carreras
- Se asignan materias a los grupos de semestres de carreras

- Se asignan profesores a las materias
- El alumno responde la evaluación
- Reporte de resultados totales

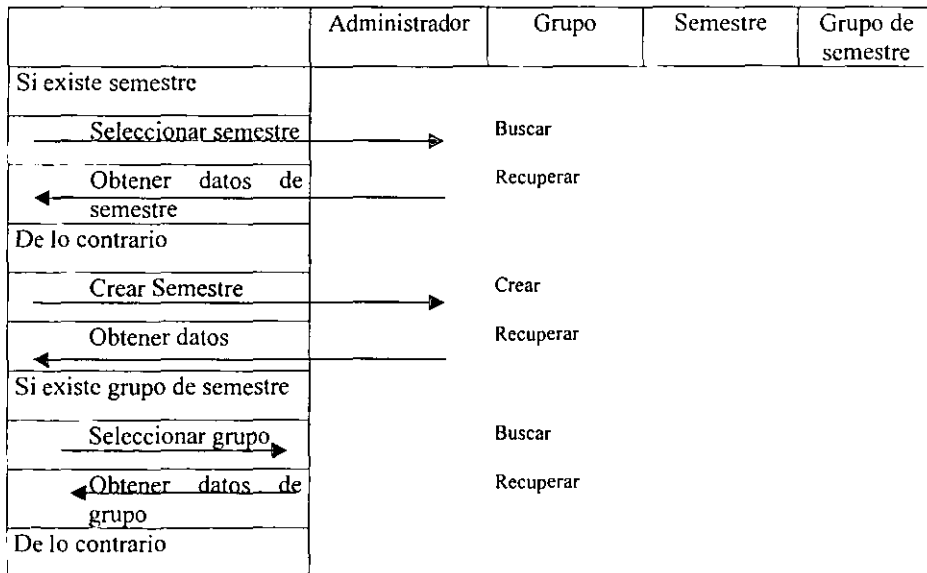
Es importante identificar las clases de personas, porque ellas representan los diferentes papeles de interacción con el sistema. Si se decide tener un seguimiento de quién, cuándo y porqué ciertos eventos se llevaron a cabo, entonces tenemos que formalizar dichos papeles. Por ejemplo, en el caso que se deseara conocer quién es el alumno que contesta la evaluación, se tendría que crear un registro para cada alumno. En este caso no se requiere dicha información (por cuestiones de anonimato en las respuestas) ni tampoco el saber quién es el administrador, por eso sólo se implica su papel para hacer más claro el análisis.

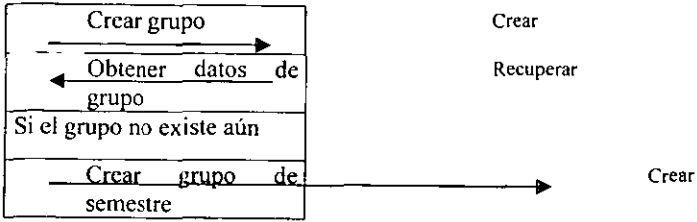
#### Escenario de establecimiento de evaluación

|   | Administrador | Evaluación | Grupo de semestre | Carrera |
|---|---------------|------------|-------------------|---------|
| Seleccionar evaluación de grupo de semestre y carrera |               |            |                   |         |
| Por cada evaluación a aplicar                         |               |            |                   |         |
| Si existe evaluación                                  |               |            |                   |         |
| → Seleccionar evaluación →                            |               | Buscar     |                   |         |
| ← Obtener datos de evaluación ←                       |               | Recuperar  |                   |         |
| De lo contrario                                       |               |            |                   |         |
| → Crear evaluación →                                  |               | Crear      |                   |         |
| ← Obtener datos de evaluación ←                       |               | Recuperar  |                   |         |
| Si existe grupo                                       |               |            |                   |         |
| Seleccionar grupo de semestre y                       |               |            | Buscar            |         |

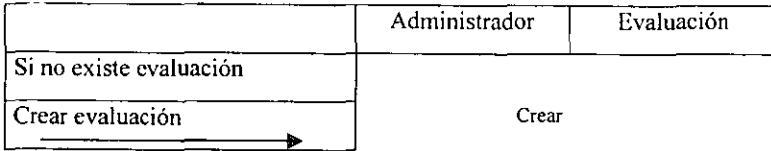


Escenario de crear grupo de semestre

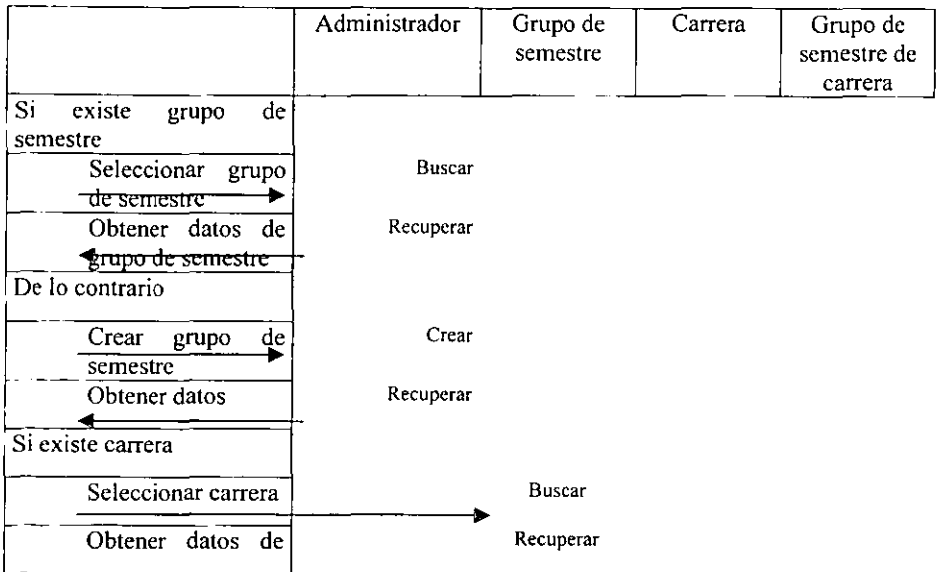


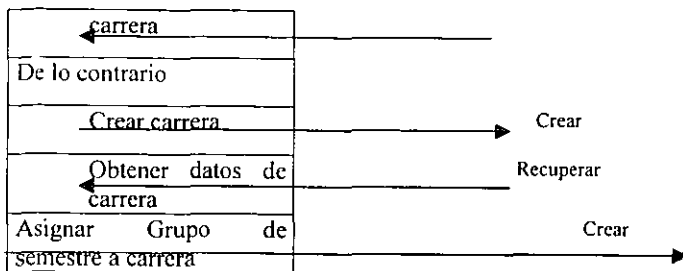


Escenario de crear evaluación



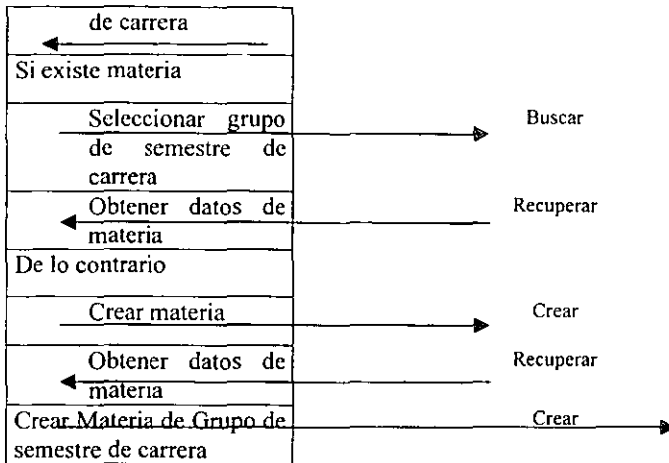
Escenario de creación de grupo de semestre de carreras



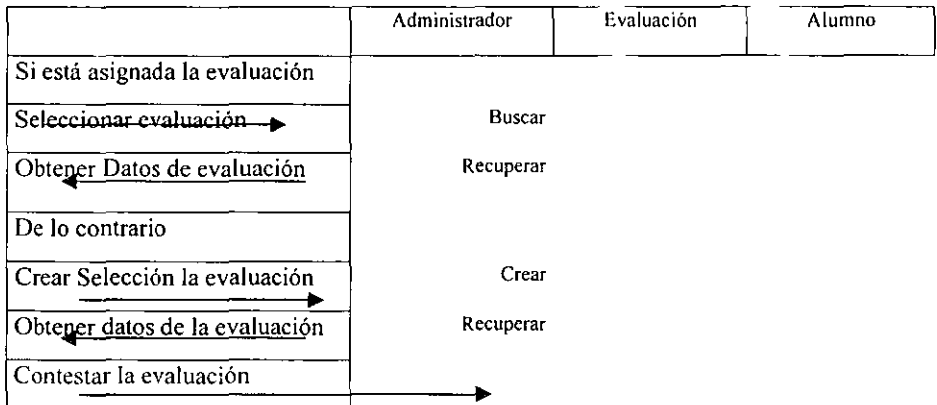


Escenario de Asignación de Materias a grupo semestre

|   | Administrador | Grupo de semestre de carrera | Materia | Materia de Grupo de semestre de carrera |
|---|---------------|------------------------------|---------|---|
| Si existe grupo de semestre de carrera        |               |                              |         |   |
| Seleccíonar grupo de semestre de carrera      | Buscar        |                              |         |   |
| Obtener datos de grupo de semestre de carrera | Recuperar     |                              |         |   |
| De lo contrario                               |               |                              |         |   |
| Crear grupo de semestre de carrera            | Crear         |                              |         |   |
| Obtener datos de grupo de semestre            | Recuperar     |                              |         |   |



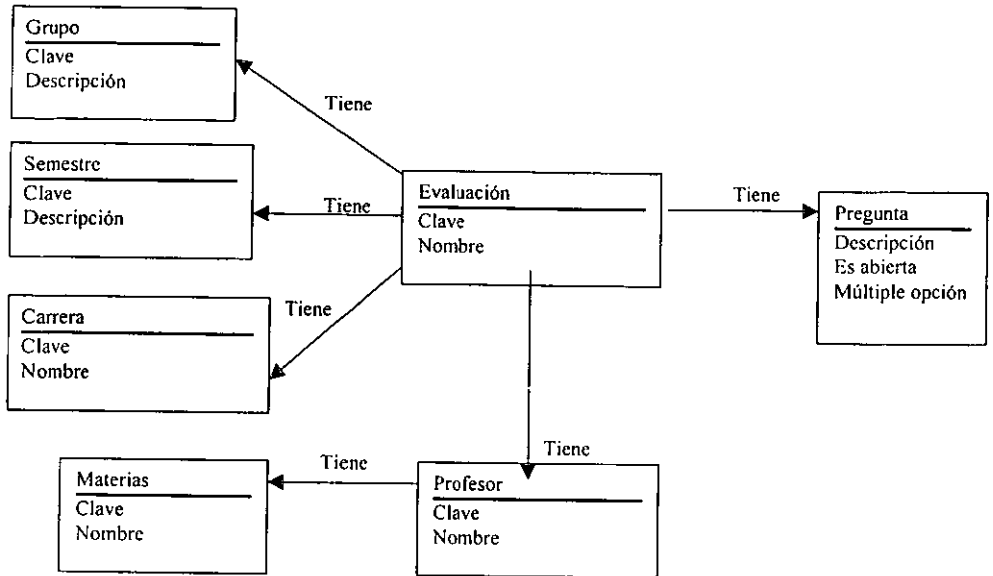
Escenario de aplicación de la evaluación



A continuación se presenta un diagrama con los objetos del sistema para dar un seguimiento gráfico a los escenarios



## Clases principales en el sistema



## 6.10 Diagramas de Gantt

### 6.10.1 Diagrama de Gantt Proyectado

En las siguientes páginas se presenta el diagrama de Gantt donde, de manera gráfica, se puede apreciar el cronograma planeado de las actividades requeridas para la creación del sistema de evaluación.

Las actividades se dividen en:

- Análisis y diseño (No tienen ninguna leyenda)

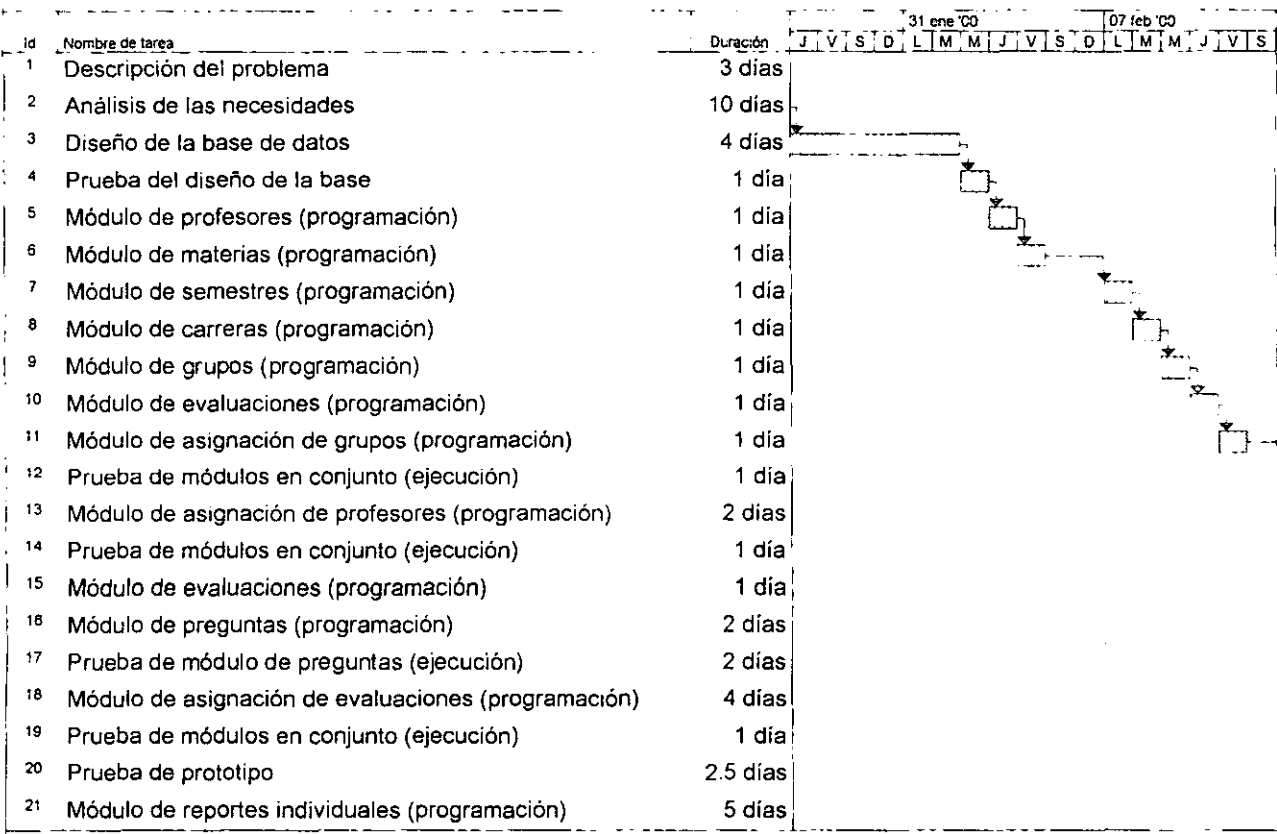
- Programación (tienen la leyenda de programación)
- Instalación (tienen leyenda de instalación)
- Ejecución (tienen leyenda de ejecución)

Las tareas de análisis y diseño incluyen la recopilación y el análisis de los datos necesarios para realizar el sistema, en las tareas de programación se desarrollaron los módulos del programa, en las de instalación y ejecución se instaló el programa para hacer las pruebas y posteriormente utilizarlo formalmente.

Adjunté en esta tesis un par de diagramas de Gantt, uno con los tiempos y actividades planeados al inicio del proyecto y otro con los tiempos y actividades reales. De esta manera puede hacerse una comparación del desarrollo total de las actividades y los tiempos necesarios.



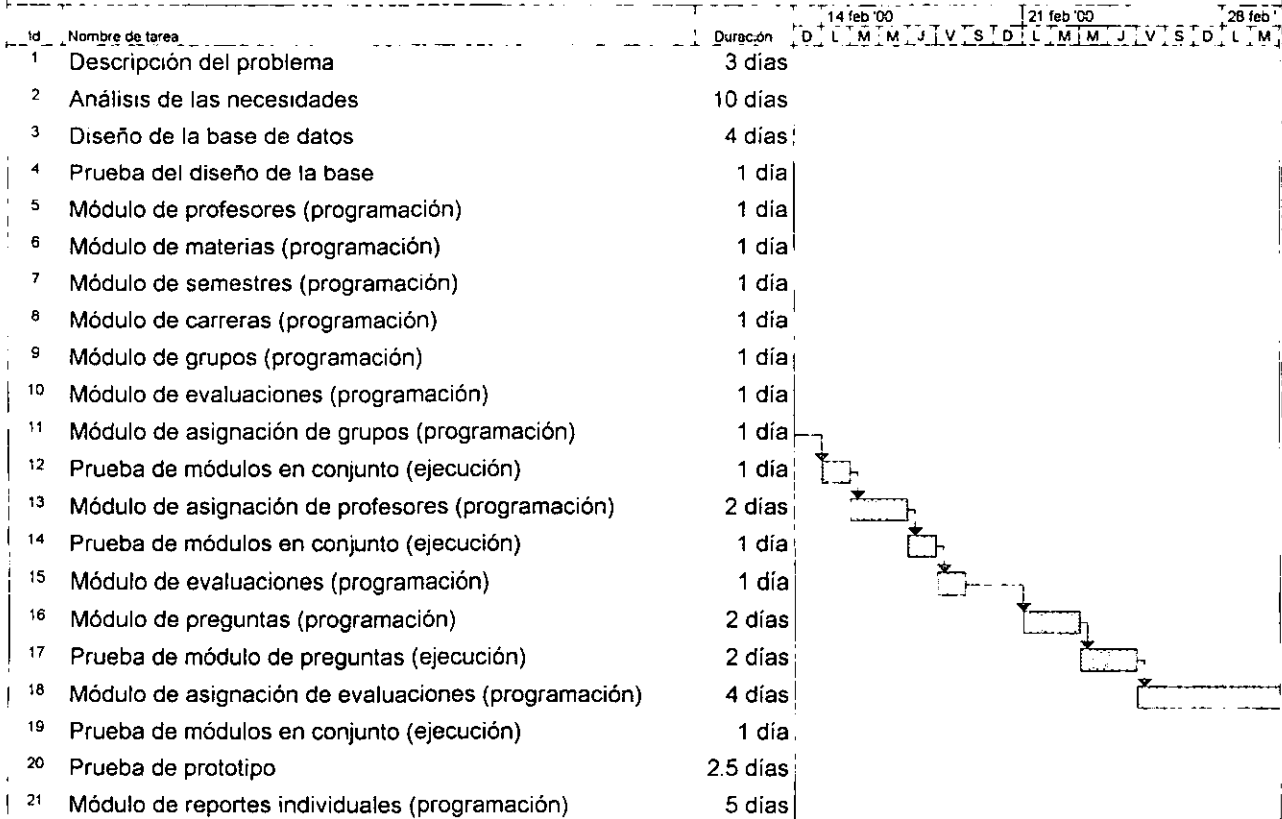
### Gráfica de Gantt Projectada



152

|  |          |  |                   |  |                      |  |
|--|----------|--|-------------------|--|----------------------|--|
| Proyecto Project1<br>Fecha vi 08/12/00 | Tarea    |  | Resumen           |  | Progreso resumido    |  |
|  | División |  | Tarea resumida    |  | Tareas externas      |  |
|  | Progreso |  | División resumida |  | Resumen del proyecto |  |
|  | Hito     |  | Hito resumido     |  |                      |  |
|  |          |  |                   |  |                      |  |

### Gráfica de Gantt Projectada



157

|                                       |   |   |   |
|---------------------------------------|---|---|---|
| Proyecto: Project1<br>Fecha: 08/12/00 | Tarea:<br>División:<br>Progreso:<br>Hito: | Resumen:<br>Tarea resumida:<br>División resumida:<br>Hito resumido: | Progreso resumido:<br>Tareas externas:<br>Resumen del proyecto: |
|---------------------------------------|---|---|---|



### Gráfica de Gantt Proyectada

| Id | Nombre de tarea                                     | Duración | Gantt                               |
|----|---|----------|-------------------------------------|
|    |   | 0        | 06 mar '00                          |
|    |   |          | 13 mar '00                          |
|    |   |          | M J V S D L M M J V S D L M M J V S |
| 1  | Descripción del problema                            | 3 días   |                                     |
| 2  | Análisis de las necesidades                         | 10 días  |                                     |
| 3  | Diseño de la base de datos                          | 4 días   |                                     |
| 4  | Prueba del diseño de la base                        | 1 día    |                                     |
| 5  | Módulo de profesores (programación)                 | 1 día    |                                     |
| 6  | Módulo de materias (programación)                   | 1 día    |                                     |
| 7  | Módulo de semestres (programación)                  | 1 día    |                                     |
| 8  | Módulo de carreras (programación)                   | 1 día    |                                     |
| 9  | Módulo de grupos (programación)                     | 1 día    |                                     |
| 10 | Módulo de evaluaciones (programación)               | 1 día    |                                     |
| 11 | Módulo de asignación de grupos (programación)       | 1 día    |                                     |
| 12 | Prueba de módulos en conjunto (ejecución)           | 1 día    |                                     |
| 13 | Módulo de asignación de profesores (programación)   | 2 días   |                                     |
| 14 | Prueba de módulos en conjunto (ejecución)           | 1 día    |                                     |
| 15 | Módulo de evaluaciones (programación)               | 1 día    |                                     |
| 16 | Módulo de preguntas (programación)                  | 2 días   |                                     |
| 17 | Prueba de módulo de preguntas (ejecución)           | 2 días   |                                     |
| 18 | Módulo de asignación de evaluaciones (programación) | 4 días   |                                     |
| 19 | Prueba de módulos en conjunto (ejecución)           | 1 día    |                                     |
| 20 | Prueba de prototipo                                 | 2.5 días |                                     |
| 21 | Módulo de reportes individuales (programación)      | 5 días   |                                     |

|                                      |           |                    |                       |
|--------------------------------------|-----------|--------------------|-----------------------|
| Proyecto: Proyec1<br>Fecha: 08/12/00 | Tarea:    | Resumen:           | Progreso resumido:    |
|                                      | División: | Tarea resumida:    | Tareas externas:      |
|                                      | Progreso: | División resumida: | Resumen del proyecto: |
|                                      | Hit:      | Hit resumido:      |                       |

### Gráfica de Gantt Proyectada

| Id | Nombre de tarea  | Duración | 08 mar '00 |   |   |   |   |   |   | 13 mar '00 |   |   |   |   |   |   |   |   |   |
|----|--|----------|------------|---|---|---|---|---|---|------------|---|---|---|---|---|---|---|---|---|
|    |  |          | M          | J | V | S | D | L | M | M          | J | V | S | D | L | M | M | J | V |
| 22 | Prueba de módulo de reportes (ejecución)                     | 3.5 días |            |   |   |   |   |   |   |            |   |   |   |   |   |   |   |   |   |
| 23 | Instalación de prototipo y pruebas (instalación y ejecución) | 5 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |   |   |   |
| 24 | Modificaciones al prototipo                                  | 5 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |   |   |   |
| 25 | Módulo de reportes generales (programación)                  | 10 días  |            |   |   |   |   |   |   |            |   |   |   |   |   |   |   |   |   |
| 26 | Prueba de módulo de reportes (ejecución)                     | 5 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |   |   |   |
| 27 | Módulo de respaldos (programación)                           | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |   |   |   |
| 28 | Prueba de módulo de respaldos (ejecución)                    | 4 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |   |   |   |
| 29 | Prueba piloto del sistema (ejecución)                        | 5 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |   |   |   |
| 30 | Implementación del sistema (instalación)                     | 2 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |   |   |   |
| 31 | Aplicación de las evaluaciones                               | 9 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |   |   |   |

Proyecto Project1  
Fecha vi 08/12/00

Tarea

División

Progreso

Hito

Resumen

Tarea resumida

División resumida

Hito resumido

Progreso resumido

Tareas externas

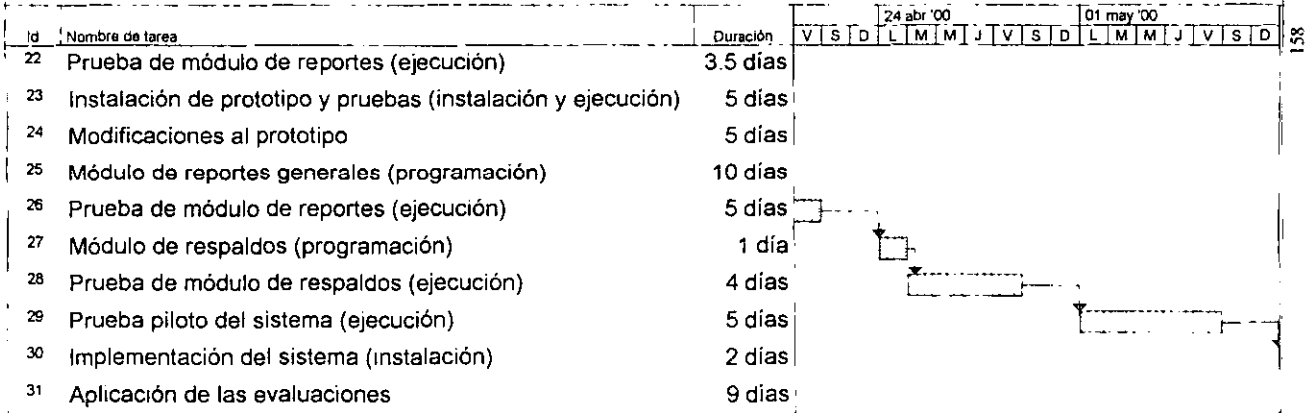
Resumen del proyecto







### Gráfica de Gantt Projectada



|                                      |          |  |                   |  |                      |  |
|--------------------------------------|----------|--|-------------------|--|----------------------|--|
| Proyecto Project1<br>Fecha: 08.12.00 | Tarea    |  | Resumen           |  | Progreso resumido    |  |
|                                      | División |  | Tarea resumida    |  | Tareas externas      |  |
|                                      | Progreso |  | División resumida |  | Resumen del proyecto |  |
|                                      | Hito     |  | Hito resumido     |  |                      |  |



### 6.10.2 Diagrama de Gantt Real

A continuación hago una breve explicación de cada una de las actividades y algunas de las diferencias, en los casos que aplica, de los tiempos programadas en relación a los reales.

Como puede apreciarse, la tarea que requirió más tiempo fue el análisis de las necesidades (10 días). En general este análisis fue encaminado en gran parte a identificar la información requerida que había motivado la aplicación de las evaluaciones. Básicamente se buscaba tener un promedio de la calificación del profesor en cada una de sus materia impartidas, y su posición en relación a la calificación obtenida de los demás docentes. También era indispensable conocer las opiniones de los alumnos a cerca del desempeño del profesor. Estos comentarios personales (respuestas abiertas) deberían concordar con los resultados numéricos (respuestas de opciones con valor) de la evaluación, de tal forma que si el profesor obtenía una calificación muy baja en el aspecto de asistencia, se espera que los alumnos hagan comentarios acerca de esto.

El diseño de la base de datos incluyo no sólo la definición de los campos necesarios y la normalización de la base, sino que además se estudiaron diferentes alternativas para llegar a un diseño más eficiente y que economizara recursos del sistema tanto al realizar operaciones como en el minimizar el espacio ocupado para guardar lo registros.

Otra de las actividades que se llevó mucho tiempo fue el módulo de reportes generales, debido a que involucraban muchas operaciones con los registros de la base de datos así como la validación de los resultados.

Una comparación entre las dos gráficas de Gantt, la real y la proyectada, permite afirmar que, hasta la instalación del prototipo, los tiempos concuerdan, pero luego existen diferencias.

En principio se agregaron actividades para el módulo de respaldo, pruebas del módulo de respaldos. También hubo una revisión al diseño y optimización del código (por ejemplo crear funciones para procedimientos repetidos y reducción y aumento de algunos campos).

Además se asignó una actividad para evaluar la facilidad de uso (con fines de mejorar futuros desarrollos y realizar algunos cambios en el programa para que resultara más amigable).

Las últimas actividades agregadas fueron las relacionadas con el manejo de la información distribuida, dicha necesidad fue debida a fallas en la conectividad de la red las cuales imposibilitaron aplicar las evaluaciones de manera simultánea en los tres laboratorios de la Universidad.

Debido a la creación de módulos individuales que resolvían cálculos simples, la creación de los reportes se simplificó en gran medida, ya que mediante la combinación de estos módulos se resolvieron procesos de cálculos más complejos.

Las actividades referentes a datos distribuidos fueron actividades no planeadas que surgieron por necesidad de última hora, fue una suerte haber ahorrado tiempo en algunas actividades (cómo en los reportes) para que no se retrasara el proyecto.

A continuación presento la gráfica de Gantt proyectada para el desarrollo del sistema de evaluación del desempeño de los profesores.

Como puede observarse a continuación, la actividad con mayor cantidad de días asignados es el análisis, el cual está encaminado a determinar las necesidades actuales que motivan el desarrollo del sistema. También se planea un lapso de 10 días para el desarrollo de los reportes, los cuales se obtendrán de las evaluaciones contestadas por los alumnos, tal cantidad de días se contempla debido a la importancia de crear módulos de procesamiento de datos que funcionen de manera individual calculando datos simples, pero que uniéndolos puedan resolver cálculos complejos que involucran gran cantidad de datos.

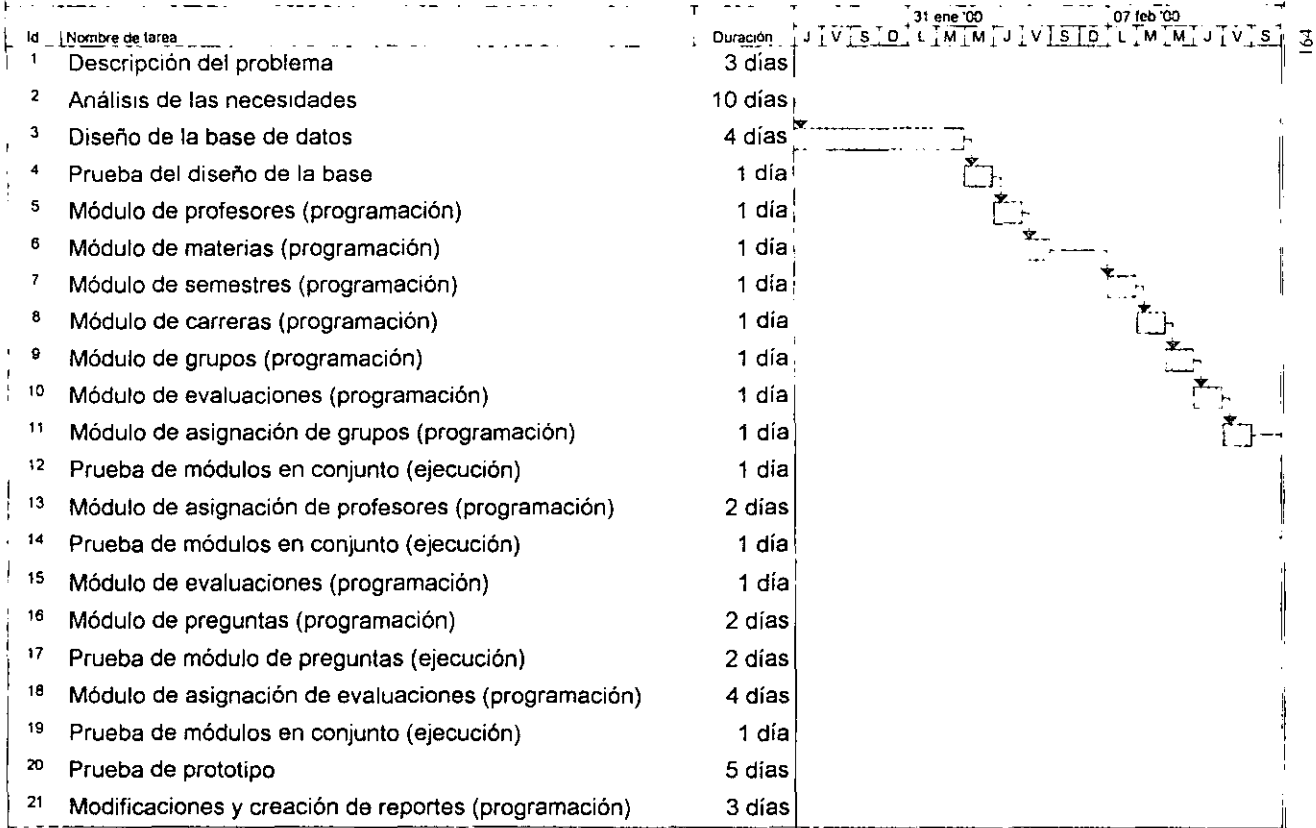
Por ejemplo, un módulo pueda calcular los resultados promediados de una materia impartida por un profesor, y la sumatoria de los resultados de ese mismo módulo, al ser utilizado para calcular todos los promedios de las materia impartidas de un profesor, nos puede dar el promedio total del profesor.

Al final se planea la creación y prueba de los módulos de respaldo, los cuales asegurarán la integridad de los datos aún cuando se presenten problemas en el almacenamiento primario.





### Gráfica de Gantt con Tiempos Reales



|  |          |  |                   |  |                      |  |
|--|----------|--|-------------------|--|----------------------|--|
| Proyecto: Project1<br>Fecha: vi 08/12/00 | Tarea    |  | Resumen           |  | Progreso resumido    |  |
|  | División |  | Tarea resumida    |  | Tareas externas      |  |
|  | Progreso |  | División resumida |  | Resumen del proyecto |  |
|  | Hito     |  | Hito resumido     |  |                      |  |
|  |          |  |                   |  |                      |  |

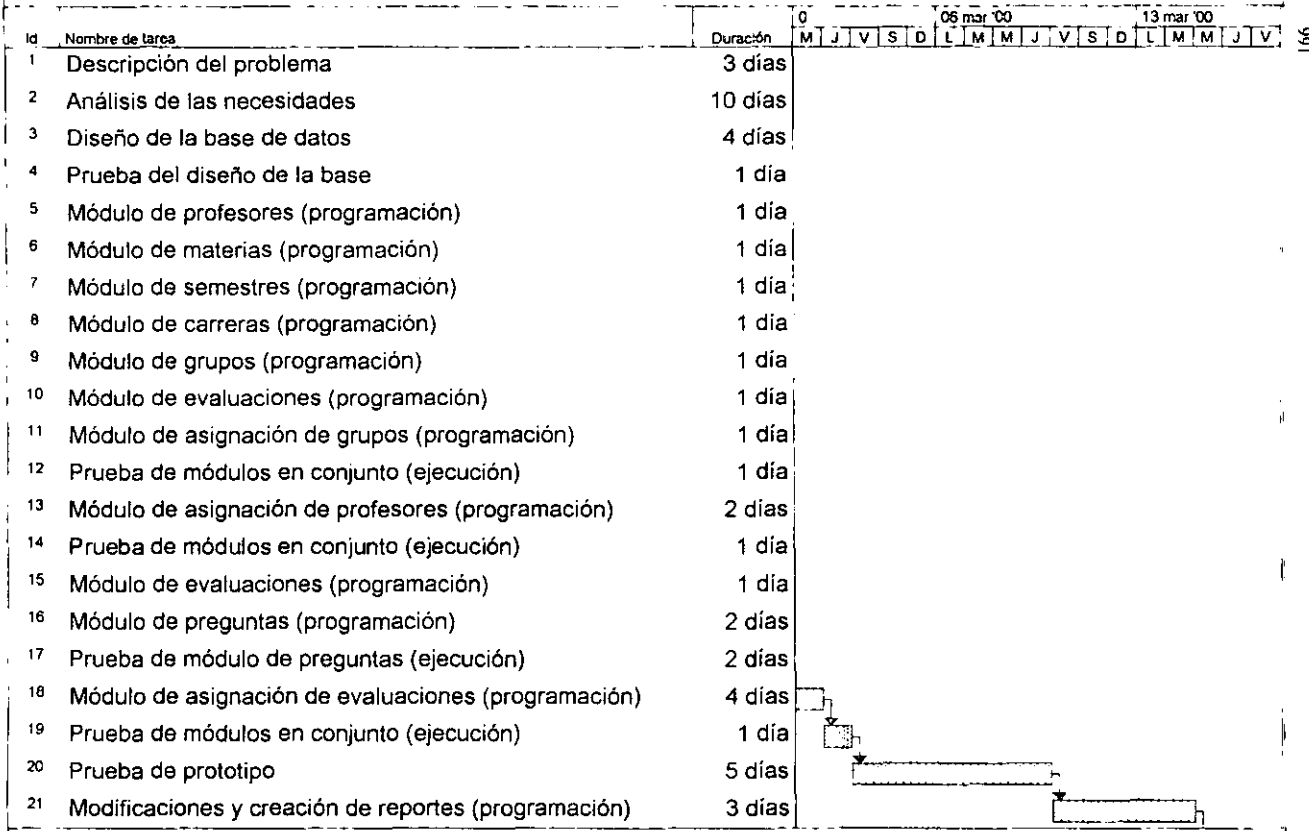
### Gráfica de Gantt con Tiempos Reales

| Id | Nombre de tarea                                      | Duración | 14 feb '00 |   |   |   |   |   |   | 21 feb '00 |   |   |   |   |   |   | 28 feb '00 |   |   |  |
|----|--|----------|------------|---|---|---|---|---|---|------------|---|---|---|---|---|---|------------|---|---|--|
|    |  |          | D          | L | M | M | J | V | S | D          | L | M | M | J | V | S | D          | L | M |  |
| 1  | Descripción del problema                             | 3 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 2  | Análisis de las necesidades                          | 10 días  |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 3  | Diseño de la base de datos                           | 4 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 4  | Prueba del diseño de la base                         | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 5  | Módulo de profesores (programación)                  | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 6  | Módulo de materias (programación)                    | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 7  | Módulo de semestres (programación)                   | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 8  | Módulo de carreras (programación)                    | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 9  | Módulo de grupos (programación)                      | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 10 | Módulo de evaluaciones (programación)                | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 11 | Módulo de asignación de grupos (programación)        | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 12 | Prueba de módulos en conjunto (ejecución)            | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 13 | Módulo de asignación de profesores (programación)    | 2 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 14 | Prueba de módulos en conjunto (ejecución)            | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 15 | Módulo de evaluaciones (programación)                | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 16 | Módulo de preguntas (programación)                   | 2 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 17 | Prueba de módulo de preguntas (ejecución)            | 2 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 18 | Módulo de asignación de evaluaciones (programación)  | 4 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 19 | Prueba de módulos en conjunto (ejecución)            | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 20 | Prueba de prototipo                                  | 5 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |
| 21 | Modificaciones y creación de reportes (programación) | 3 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |  |

165

|  |          |                      |                   |  |
|--|----------|----------------------|-------------------|--|
| Proyecto: Proyec11<br>Fecha: vi 08/12/00 | Tarea    |                      | Resumen           |  |
|  | División |                      | Tarea resumida    |  |
|  | Progreso |                      | División resumida |  |
|  | Hito     |                      | Hito resumido     |  |
|  |          |                      | Progreso resumido |  |
|  |          | Tareas externas      |                   |  |
|  |          | Resumen del proyecto |                   |  |

### Gráfica de Gantt con Tiempos Reales



|  |          |  |                   |  |                      |  |
|--|----------|--|-------------------|--|----------------------|--|
| Proyecto Project1<br>Fecha vi 08/12/00 | Tarea    |  | Resumen           |  | Progreso resumido    |  |
|  | División |  | Tarea resumida    |  | Tareas externas      |  |
|  | Progreso |  | División resumida |  | Resumen del proyecto |  |
|  | Hito     |  | Hito resumido     |  |                      |  |
|  |          |  |                   |  |                      |  |

### Gráfica de Gantt con Tiempos Reales

| Id | Nombre de tarea  | Duración | 0 |   |   |   |   |   |   | 06 mar '00 |   |   |   |   |   |   | 13 mar '00 |   |   |   |   |  |  |
|----|--|----------|---|---|---|---|---|---|---|------------|---|---|---|---|---|---|------------|---|---|---|---|--|--|
|    |  |          | M | J | V | S | D | L | M | M          | J | V | S | D | L | M | M          | J | V | S | D |  |  |
| 22 | Prueba de módulo de reportes (ejecución)                     | 3,5 días |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |
| 23 | Instalación de prototipo y pruebas (instalación y ejecución) | 5 días   |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |
| 24 | Ultimas Modificaciones al prototipo                          | 5 días   |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |
| 25 | Modulo de respaldo y extracción de datos (programación)      | 3 días   |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |
| 26 | Prueba de respaldos y extracción (ejecución)                 | 2 días   |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |
| 27 | Prueba en entorno multiusuario (ejecución)                   | 1 día    |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |
| 28 | Verificación de reportes (ejecución)                         | 1 día    |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |
| 29 | Implementación del sistema (instalación)                     | 1 día    |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |
| 30 | Verificar concurrencia en la base de datos                   | 1 día    |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |
| 31 | Revisión al diseño para mejoras                              | 1 día    |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |
| 32 | Revisión de codigo para optimización                         | 2 días   |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |
| 33 | Programación de nuevo código                                 | 5 días   |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |
| 34 | Prueba de las modificaciones                                 | 2 días   |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |
| 35 | Búsqueda de errores no detectados                            | 1 día    |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |
| 36 | Prueba de Facilidad de uso                                   | 1 día    |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |
| 37 | Diseño de ejecución con datos distribuidos                   | 1 día    |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |
| 38 | Diseño de módulo de extracción de datos distribuidos         | 0,5 días |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |
| 39 | Modulo de datos distribuidos (Programación)                  | 0,5 días |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |
| 40 | Prueba y ejecución en múltiples laboratorios (Ejecución)     | 3 días   |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |
| 41 | Aplicación de las evaluaciones                               | 9 días   |   |   |   |   |   |   |   |            |   |   |   |   |   |   |            |   |   |   |   |  |  |

167

|  |          |  |                   |  |                      |  |
|--|----------|--|-------------------|--|----------------------|--|
| Proyecto: Project1<br>Fecha: vi 08/12/00 | Tarea    |  | Resumen           |  | Progreso resumido    |  |
|  | División |  | Tarea resumida    |  | Tareas externas      |  |
|  | Progreso |  | División resumida |  | Resumen del proyecto |  |
|  | Hito     |  | Hito resumido     |  |                      |  |
|  |          |  |                   |  |                      |  |

### Gráfica de Gantt con Tiempos Reales

| Id | Nombre de tarea  | Duración | 20 mar '00 |   |   |   |   |   |   | 27 mar '00 |   |   |   |   |   |   | 03 |   |   |
|----|--|----------|------------|---|---|---|---|---|---|------------|---|---|---|---|---|---|----|---|---|
|    |  |          | S          | D | L | M | M | J | V | S          | D | L | M | M | J | V |    | S | D |
| 22 | Prueba de módulo de reportes (ejecución)                     | 3.5 días |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |
| 23 | Instalación de prototipo y pruebas (instalación y ejecución) | 5 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |
| 24 | Ultimas Modificaciones al prototipo                          | 5 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |
| 25 | Modulo de respaldo y extracción de datos (programación)      | 3 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |
| 26 | Prueba de respaldos y extracción (ejecución)                 | 2 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |
| 27 | Prueba en entorno multiusuario (ejecución)                   | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |
| 28 | Verificación de reportes (ejecución)                         | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |
| 29 | Implementación del sistema (instalación)                     | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |
| 30 | Verificar concurrencia en la base de datos                   | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |
| 31 | Revisión al diseño para mejoras                              | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |
| 32 | Revisión de codigo para optimización                         | 2 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |
| 33 | Programación de nuevo código                                 | 5 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |
| 34 | Prueba de las modificaciones                                 | 2 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |
| 35 | Búsqueda de errores no detectados                            | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |
| 36 | Prueba de Facilidad de uso                                   | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |
| 37 | Diseño de ejecución con datos distribuidos                   | 1 día    |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |
| 38 | Diseño de módulo de extracción de datos distribuidos         | 0.5 días |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |
| 39 | Modulo de datos distribuidos (Programación)                  | 0.5 días |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |
| 40 | Prueba y ejecución en múltiples laboratorios (Ejecución)     | 3 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |
| 41 | Aplicación de las evaluaciones                               | 9 días   |            |   |   |   |   |   |   |            |   |   |   |   |   |   |    |   |   |

168

Proyecto Project1  
Fecha vi 08/12/00

Tarea

División

Progreso

Hito

Resumen

Tarea resumida

División resumida

Hito resumido

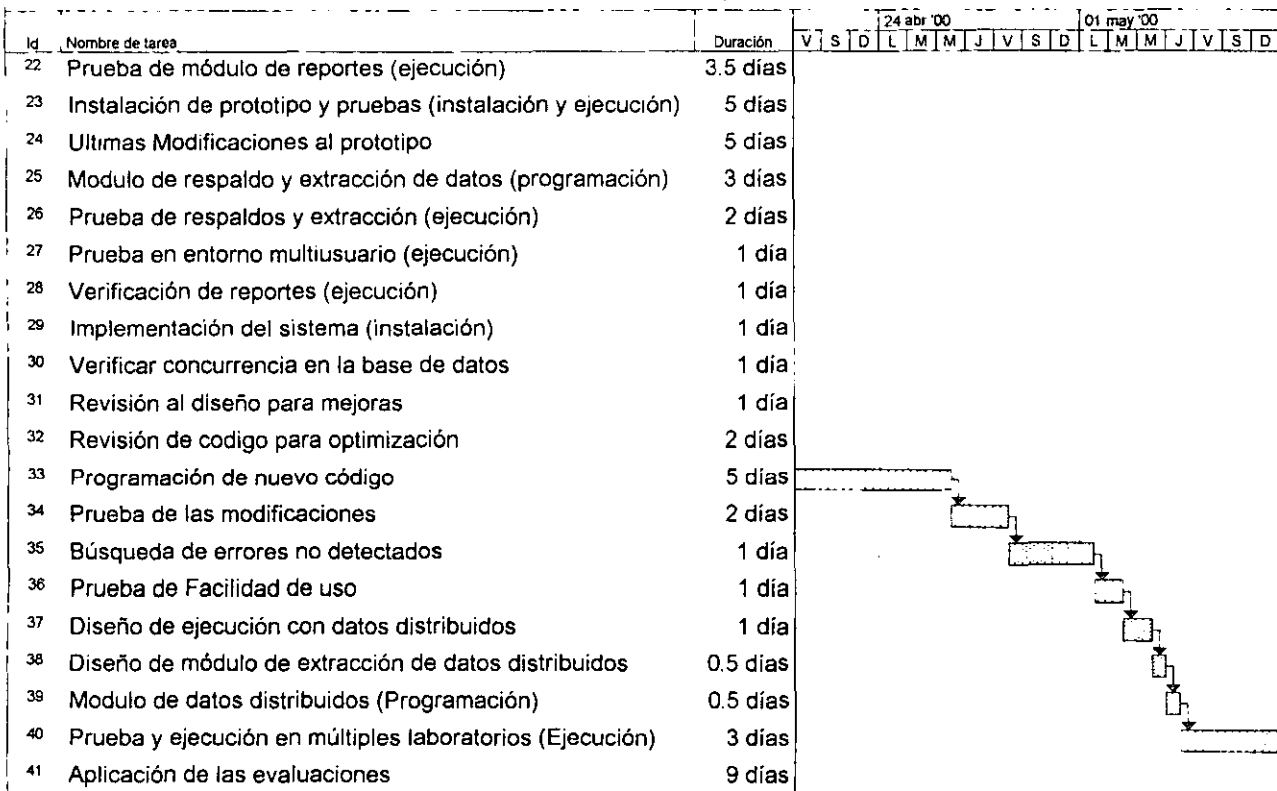
Progreso resumido

Tareas externas

Resumen del proyecto



### Gráfica de Gantt con Tiempos Reales



170

|  |          |  |                   |  |                      |  |
|--|----------|--|-------------------|--|----------------------|--|
| Proyecto Project1<br>Fecha vi 08/12/00 | Tarea    |  | Resumen           |  | Progreso resumido    |  |
|  | División |  | Tarea resumida    |  | Tareas externas      |  |
|  | Progreso |  | División resumida |  | Resumen del proyecto |  |
|  | Hito     |  | Hito resumido     |  |                      |  |





## 6.11 Prueba de Sistemas

Todo sistema que involucre información importante debe ser puesto a prueba para poder determinar si su aplicación puede cubrir los requisitos necesarios de funcionalidad, exactitud, efectividad y seguridad. Los pasos para probar un sistema propuestos en el sistema TEST RX de la compañía CSST Technologies son los siguientes: (<http://www.csst-technologies.com/>, Junio 2000)

a) Formar el equipo de pruebas: El equipo de prueba debe estar organizado de manera concurrente con el equipo de desarrollo. El objetivo del equipo de prueba es realizar la verificación y validación hasta la que se realice la implementación. Para un proyecto en específico, el objetivo de este equipo es:

- Realizar verificación y validación de las entregas de desarrollo
- Actuar como consultantes para el equipo de desarrollo durante la unidad de prueba.

En el proyecto de esta tesis, el desarrollo y todos los demás pasos fueron llevados a cabo solamente por una persona, por lo cual se descarta la formación del equipo de prueba.

b) Realizar un análisis de riesgo: EL objetivo es identificar los componentes de alto riesgo de la aplicación que deben ser probados más a fondo, e identificar los componentes más propensos a error dentro de la aplicación para probarlos más rigurosamente. Los resultados del análisis pueden ser utilizados para determinar los objetivos de las pruebas.

## Métodos de análisis de riesgos:

- Juicio e instinto: Es el método más común, es informal y se basa en los conocimientos y experiencia del evaluador de la aplicación. Una desventaja es que cuando las pruebas se hacen en equipo, no todos tienen la misma experiencia ni conocimientos
- La estimación de dinero: Este método cuantifica el riesgo en base al dinero, es difícil de usar porque los resultados están basados en estimados de frecuencia de ocurrencia y de la pérdida por ocurrencia. Por lo tanto no es muy preciso.
- Identificación y cuantificación de los atributos de riesgo: En este método se identifican los atributos que producen el riesgo. El evaluador utiliza una serie de valores de una tabla para determinar cuáles son las áreas de riesgo. La tabla determina cuales componentes de la aplicación deben ser probados más cuidadosamente que otros, y el total de la puntuación puede ser utilizada para comparar el sistema con otros.
- Paquete de estimación de riesgos del programa: este es un método automatizado que involucra la compra de un paquete de estimación. La ventaja es facilidad de uso y la habilidad de identificar si hay errores en el análisis. Existen paquetes de programas para la estimación automática de riesgos que utilizan los métodos de estimación de dinero y la identificación y cuantificación de los atributos de riesgo, no obstante, los evaluadores pueden crear sus propios cuestionarios de estimación de error.

En el desarrollo del sistema de evaluación del desempeño de los profesores, se utilizó principalmente el método de juicio e instinto, apoyándose en la ventaja de que sólo una persona era la encargada de realizar las pruebas. Sin embargo en módulos específicos como el de respaldo y captura de respuestas se utilizó el método de identificación y cuantificación de los atributos de riesgo para establecer mejor los controles y minimizar la pérdida de información en situaciones adversas como fallos en los sistemas o capturas erróneas.

- c) Establecer los objetivos de las pruebas: Es una parte crítica en la planeación del proceso de pruebas. Definir los objetivos de evaluación es una de las actividades más difíciles porque las personas frecuentemente no saben claramente lo que quieren hacer hasta que lo comienzan a hacer. Esto significa que hasta el mejor plan de evaluación sufre cambios durante el proceso de ejecución de las pruebas.

Un objetivo es una meta en la evaluación. Es el establecimiento de lo que el evaluador quiere lograr cuando se implante una actividad de evaluación específica. Cada actividad puede tener varios objetivos. Existen dos niveles de especificación de objetivos que un plan de evaluación debe contener: Los objetivos generales de alto nivel, y los objetivos comprobables de bajo nivel de cada evaluación implementada.

- d) Creación de los planes de pruebas: El propósito de los planes de las pruebas es especificar quién, qué, cuándo y porqué de las pruebas.

- e) **Diseño y construcción de las opciones de prueba:** Se analizan las opciones para aplicar las pruebas, debe tomarse en cuenta la factibilidad económica y las necesidades. En esta parte es posible aplicar técnicas de diseño asistido por computadora (CASE)
- f) **Pruebas de ejecución individual y de integración:** El objetivo de las pruebas individuales es probar que las funciones del programa funcionan como unidades individuales en las que cada una hace lo que tiene que hacer. El desarrollador del módulo de programa debe ser el que conduzca la prueba individual.
- El propósito de las pruebas de integración es probar que las unidades de la aplicación trabajan juntas de manera apropiada y realizan sus funciones. También se prueba que las unidades no ejecuten tareas que no les corresponden. El equipo de evaluación debe realizar la prueba de integración.
- g) **Prueba de ejecución del sistema.** El propósito es utilizar el sistema en un ambiente controlado, simulando como lo utilizará el usuario. Las pruebas del sistema deben probar que el sistema total realizará lo que se supone que debe hacer y que no realizará cosas que no debe hacer.
- h) **Análisis y reporte de las pruebas:** Indica cuales funciones del sistema trabajan y cuáles no. Este reporte debe describir el nivel en el cual el sistema cumple los objetivos de la prueba. Este reporte es el principal y cubre todo acerca de las actividades que han sido completadas.
- i) **Ejecutar pruebas de regresión:** Su objetivo principal es verificar que las mejoras al sistema, reajustes de rutinas y mantenimiento no afectan la funcionalidad original del

sistema. El segundo objetivo es probar que los cambios en mejoras y mantenimiento realizan su cometido y no otras cosas diferentes.

Las pruebas de regresión se efectúan cada vez que hay cambios en el sistema. Esto significa que las pruebas de regresión no deben ser descartadas después de cada prueba. Las pruebas regresivas son la forma de prueba más tratable para automatizar. En este paso pueden utilizar Robots SQA para diseñar y construir scripts de prueba. Los scripts pueden ser mejorados y reconstruidos para cada prueba regresiva subsecuente.

- j) Reportes de las pruebas regresivas: Indican cuáles funciones del sistema funcionan y cuales no. El reporte debe describir el nivel en el cual el sistema cumplió con los objetivos de la prueba. Debe incluir recomendaciones basadas en el éxito o fracaso del alcance de objetivos.

Debido a que las pruebas de regresión son un proceso cíclico que continuará con cada cambio en el sistema, este reporte debe ser archivado y actualizado en cada prueba del sistema. ([http://www.csst-technologies.com/genericStandards\\_Overview.html](http://www.csst-technologies.com/genericStandards_Overview.html), Junio 2000)

## Prueba #1

**Objetivo:** Instalación del sistema en un ambiente multiusuario y determinar la funcionalidad del sistema, verificar el acceso, almacenamiento y extracción de los datos así como la exactitud de los reportes.

**Fecha:** Marzo 2000

**Descripción:** Se utilizaron 3 computadoras del laboratorio morado para realizar una prueba simultánea del sistema.

El programa se instaló en una computadora de la red (La computadora M1) y se accedió a ella desde las computadoras M2 y M3 de esa misma red. Esta acción se conoce como mapeo y para realizarlo se ejecuto el comando "NET USE X: \\M1\RAFA". Con esto las computadoras creaban una unidad de red X:\M1\RAFA en la cual podían acceder a los archivos del examen. Para probar el sistema se iba a requerir de tres personas, pero por problemas de horarios de el personal a cargo, las pruebas fueron realizadas solamente por dos personas.

Se utilizaron los exámenes escritos de un grupo para capturar las respuestas y luego comparar los resultados del programa con los ya obtenidos de la evaluación escrita.

Se capturaron 15 exámenes de manera simultánea en las dos computadoras (8 en M1 y 7 en M2) y no se registró ningún problema, los resultados fueron los esperados y sólo había un detalle en la presentación de los resultados: no aparecían los decimales en los resúmenes totales, cosa totalmente atribuible a la configuración regional del sistema operativo de las computadoras. Se pidió a los encargados del laboratorio que cambiaran dicha configuración para que estuviera lista para las futuras pruebas.

**Resultados:** La prueba fue un éxito.

Se encontraron problemas en la presentación de los decimales y algunas faltas de ortografía y palabras cortadas en las preguntas de la evaluación y en algunos mensajes del sistema.

**Opciones correctivas:**

Errores en los mensajes y faltas de ortografía en las preguntas:

- Captura correcta de las preguntas.
- Corrección de los mensajes de sistema

Error en la presentación de decimales:

- Cambio de la configuración regional del sistema operativo de las computadoras en las que se aplicará el examen.
- Cambio en el código del sistema para que imprima las cantidades como texto.

**Procedimiento correctivo:**

Se capturaron las respuestas de manera correcta y se revisó la ortografía de los mensajes de sistema.

Debido a que no se cambió la configuración regional se optó por modificar el código del programa para que aparecieran los decimales.

**Evaluación de las correcciones:** Las correcciones fueron acertadas.

Las respuestas fueron corregidas y las cantidades presentaban los decimales esperados en los reportes.

## Prueba #2

**Objetivo:** Determinar la funcionalidad del sistema en ejecución múltiple, verificar el acceso, almacenamiento y extracción de los datos así como verificar exactitud de los reportes.

**Fecha:** Miércoles 12 de abril de 2000

**Descripción:** La prueba del sistema fue planeada para aplicarse en el laboratorio amarillo, que cuenta con las computadoras de mejores características de hardware y software, pero debido a que no era posible aplicar la evaluación en horarios convenientes porque estaba ocupado la mayor parte del tiempo, entonces se decidió aplicarlo en el laboratorio rojo. Para la aplicación se pidió la colaboración de los alumnos de 10 semestre de informática y del profesor con quien tenían clase. Lamentablemente no fue posible aplicar la evaluación debido a que las computadoras del laboratorio rojo no contaban con el software necesario para la ejecución del sistema, el cual necesitaba de controladores que no estaban disponibles. La instalación de los controladores necesarios tomaría demasiado tiempo como para esperar para ejecutar el sistema, por lo tanto se canceló la prueba y se pidió al personal a cargo de los laboratorios que realizaran las instalaciones necesarias.

**Resultados:** La prueba fue un éxito.

La prueba no se pudo realizar. Primero se encontró con el dilema de que no se podía instalar el programa en cada computadora utilizando un CD de instalación y en caso de utilizar disquetes de 3 ½ planteaba el problema de que la cantidad de disquetes de instalación (13 disquetes) hacía imposible una rápida y total instalación.



**Opciones correctivas:**

- Instalar el programa en cada computadora
- Instalar en cada computadora los controladores necesarios

**Procedimiento correctivo:**

Instalar en cada computadora los controladores necesarios para la ejecución del sistema

**Evaluación de las correcciones:** Las correcciones no fueron realizadas. Solamente el personal a cargo del departamento de informática puede instalar los controladores en las computadoras de los laboratorios. El fracaso de las correcciones no es imputable al sistema.

### Prueba #3

**Objetivo:** Determinar la funcionalidad del sistema en ejecución múltiple, verificar el acceso, almacenamiento y extracción de los datos así como la exactitud de los reportes.

**Fecha:** Jueves 13 de abril de 2000

**Descripción:** Para esta prueba se seleccionó el laboratorio amarillo, el grupo de 4º semestre de informática tenía clase en dicho laboratorio. El programa se instaló en una computadora de la red (La computadora A3) y se accedió a ella desde cada una de las otra computadoras de esa misma red. Esta acción se conoce como mapeo y para realizarlo se ejecuto el comando "NET USE X: \\A3\EXAMEN". Con esto las computadoras creaban una unidad de red X:A3\EXAMEN en la cual podían acceder a los archivos del examen.

Del total de las computadoras del laboratorio amarillo sólo 10 pudieron realizar el mapeo debido a un límite que existe para que una computadora puede ser accesada por otras. Se llevo a cabo la primera evaluación con 10 computadoras simultáneamente, y no se registró ningún problema. Al ejecutar la segunda evaluación el programa registró los siguientes:

1. Error al bloquear el archivo del programa ejecutable
2. Error de sistema al ejecutar el archivo del programa.
3. Error al leer la base de datos de respuestas.

Evaluando los detalles y la situación con el asesor de informática se llegó a la conclusión que los problemas eran una consecuencia de la forma en que se conectaban las computadoras con los archivos del sistema, y no por errores en el programa. Se eliminó el programa de la

máquina A3 y se volvió a instalar, luego se procedió a ejecutar de nuevo la evaluación, pero esta vez utilizando solamente 8 computadoras. Se registraron los errores en la aplicación del examen por octava ocasión, pero se pudo volver a continuar con su ejecución, luego se ejecuto en las 10 computadoras y al cabo de tres ejecuciones del examen se presentaron los mismos errores la base de datos resultó dañada. La prueba se detuvo y se convino en buscar otra manera de acceder de manera concurrente al examen, lamentablemente no se pudo hacer nada más ese día porque el encargado del laboratorio había salido.

**Resultados:** La prueba fue un éxito.

El almacenamiento, acceso y extracción de los datos fueron correctos, al menos hasta que se daño la base de datos debido al error de concurrencia en la red.

Se encontraron errores en el acceso concurrente a la base de datos y la base de datos sufrió daños como consecuencia de los problemas de acceso, por lo anterior no fue posible obtener ningún reporte.

**Opciones correctivas:**

Daño a la base de datos:

- Respaldar la base de datos antes de cualquier aplicación para minimizar el riesgo de pérdida total de los resultados.

Error en el acceso concurrente:

- Ejecutar el programa pero ahora instalado en el servidor para que todas las computadoras tuvieran acceso directo a él y no utilizando un mapeo.

- Instalar en cada una de las computadoras el programa y luego hacer un módulo que recogiera los resultados de cada computadora y los concentrara en una sola base de datos para obtener los resultados.
- Instalar en cada una de las computadoras el programa pero que se enviaran los datos a la base de datos en el servidor.
- Revisar el código para optimizar el acceso.

Se comentó con el encargado del laboratorio los problemas que había tenido, él explicó que pudiera ser una causa el límite en el número de computadoras que pueden acceder a una mediante el mapeo.

Después de evaluar los pros y las contras de las opciones se decidió por la opción de instalar en cada una de las computadoras el programa pero que se enviaran los datos a la base de datos en el servidor debido a que el encargado del laboratorio comentó que había realizado pruebas de sistemas similares y había utilizado esas características con buenos resultados.

#### **Procedimiento correctivo:**

Revisé el código y mejoré el acceso a la base de datos, esencialmente modifiqué el código para acceder a la base de datos sólo cuando se iban a leer las preguntas o guardar los registros.

Instalar el programa en cada una de las computadoras y que accedieran y almacenaran los datos en una unidad de red común.

**Evaluación de las correcciones:** Las correcciones realizadas a la base de datos fueron acertadas. La corrección en el acceso a la base de datos para que accasaran a una misma base de datos desde cada computadora fue satisfactoria.

#### Prueba #4

**Objetivo:** Determinar la funcionalidad del sistema en ejecución múltiple con concurrencia en la base de datos, verificar el acceso, almacenamiento y extracción de los datos, así como la exactitud de los reportes.

**Fecha:** Viernes 14 de abril de 2000

**Descripción:** Para la prueba de este día se utilizó el laboratorio morado, se instaló el programa en el servidor y se instaló el programa en cada una de las máquinas. Fue necesario compilar el programa para que hiciera referencia a la unidad del servidor y de esta manera todos los resultados fueran enviados a la misma base de datos sin importar donde estuviera instalado. Para la evaluación se pidió la colaboración de 10 alumnos de 2 semestre de informática que tenían clase en otro de los laboratorios.

Los alumnos contestaron 10 exámenes en las computadoras del laboratorio morado y no se registró ningún problema.

Al finalizar las evaluaciones se comentaron los resultados con el asesor de informática y se acordó que el sistema de evaluación debería tener la posibilidad de ejecutarse con las características que utilizamos ese día, pero también con la opción de que se guardaran los resultados en una base de datos de cada computadora, y luego pasarlas a una base de datos central.

**Resultados:** La prueba fue un éxito, el sistema tuvo que ser compilado para poder manejar la característica de acceder a la base de datos en distintas rutas.

**Opciones correctivas:**

Modificar el código del programa para que pudiera especificarse la ruta en la que debería buscar la base de datos

**Procedimiento correctivo:**

Se actualizó el programa implementándole la característica de que leyera desde un archivo de texto (dir.cfg) la ruta de acceso a las bases de datos, de esta manera podría ejecutarse desde cualquier computadora y se pudiera obtener y guardar información de la base de datos en la red. En caso de que se moviera la base simplemente se cambiaba la ruta en el archivo de texto. Si el programa no encontraba el archivo de texto "dir.cfg" con la ruta, él mismo creaba el archivo con la ruta actual, en caso de que en la ruta contenida en el archivo no estuvieran las bases de datos, el sistema pedía confirmación para establecer la ruta actual como predeterminada para el programa.

**Evaluación de las correcciones:** Las correcciones implementadas al sistema fueron satisfactorias, el sistema pudo ejecutarse en cada una de las computadoras y hacer referencia a una base de datos en una unidad de red. Los datos fueron recuperados y almacenados de manera correcta y los reportes fueron exactos.

NOTA: Del 15 de abril al 2 de mayo hubo vacaciones con motivo de semana santa.

El 2 de mayo, después de regresar de las vacaciones de semana santa, se acordó con el asesor de pedagogía continuar con las pruebas del sistema durante toda la semana porque las evaluaciones oficiales empezarán a partir del próximo lunes 8 de mayo.

#### Prueba #5

**Objetivo:** Búsqueda de errores no detectados en el sistema (de acceso a la base de datos y concurrencia de ejecución) y prueba de facilidad de operación del sistema.

**Fecha:** 3 de Mayo del 2000

**Descripción:** Ya los resultados del programa serían oficiales a partir de ahora, las pruebas anteriores demostraban que la información era obtenida correctamente, aún cuando no estaba al 100% operacional (faltaban el módulo de reportes de preguntas abiertas) ya el sistema era capaz de almacenar y administrar la información.

Se instalaron las bases de datos en un directorio del servidor que era accesado por cada una de las estaciones de trabajo del laboratorio Amarillo, las bases del programa y el módulo de administración en dicho directorio. Desde las estaciones de trabajo, ese directorio aparecía como una unidad de red O:\ para poder tener acceso a ella deberían entrar los alumnos con el login normal (usuario: A + número exterior del equipo).

El asesor de pedagogía dio una breve introducción a cerca de la evaluación, valor de los puntos y los objetivos que se pretendía alcanzar con ella antes de iniciarla.

Se aplicó la evaluación a los alumnos de 8° semestre de informática, los problemas que se encontraron fueron los siguientes:

- Algunas computadoras no podían acceder al directorio de las bases de datos, al revisar el problema se descubrió que los alumnos habían entrado con su login personal, al entrar con el login normal, pudieron acceder a la base de datos.
- Cuando entraron con el login personal el sistema no detectó las bases de datos y puso la ruta actual del sistema como predeterminada, en este caso fue necesario manualmente editar el archivo “dir.cfg” para que hiciera referencia a la unidad O:\
- Solamente podían acceder al directorio de las bases de datos 21 computadoras al mismo tiempo, es decir que la red únicamente soportaba 21 computadoras accedendo a los mismos recursos al mismo tiempo.

Más tarde, al finalizar la evaluación se comentó con el encargado del laboratorio amarillo el problema del acceso que nos restringía a tan sólo 21 entradas concurrentes al directorio compartido del sistema, el encargado explicó que el problema podría radicar en las licencias de la red.

Una de las soluciones para la restricción del número de computadoras con acceso simultáneo fue la de dividir los grupos para que sólo entraran en grupos de 21 personas al laboratorio amarillo. Esto planteó la necesidad de utilizar todos los laboratorios de manera simultánea cuando los grupos fueran numerosos.

**Resultados:** La prueba fue un éxito.

Se encontraron errores en la ruta de acceso a la base de datos debido a que los alumnos entraron con su login personal. Solamente 21 computadoras pudieron ejecutar de manera



simultanea la evaluación. Se planteó la necesidad de que se pudieran conectar los tres laboratorios para aplicar la evaluación en más de 21 computadoras.

### **Opciones correctivas:**

Error en la ruta por login personal:

- Dar instrucciones a los alumnos de cómo iniciar las computadoras.

Error en el acceso a la base de datos compartida por más de 21 computadoras:

- Comprar más licencias para las computadoras del sistema operativo.
- Instalar el programa y las bases de datos en cada computadora y después rescatar los resultados de cada una de las computadoras.
- Dividir los grupos y sólo aplicar 21 evaluaciones en el laboratorio amarillo.

### **Procedimiento correctivo:**

Solución al error en la ruta de la base de datos:

Se dieron instrucciones a los alumnos de cómo deben iniciar su sesión en la red.

Se planeó para que sólo entraran grupos de 21 personas al laboratorio amarillo.

**Evaluación de las correcciones:** Las correcciones fueron efectivas, una vez que se explica la manera de acceder a la red no se presenta ningún error en la ruta de acceso.

La solución de dividir los grupos no sería muy eficaz y se optó por utilizar dos laboratorios a la vez cuando el número de alumnos fuera numeroso.

## Prueba #6

**Objetivo:** Evaluar el sistema en ejecución en varios laboratorios a la vez.

**Fecha:** 4 - 6 de Mayo del 2000

**Descripción:** 4 de mayo. Esta vez aplicamos la evaluación al grupo de 4º semestre, contrario a nuestras expectativas de que sólo podríamos contar con la aplicación de 21 evaluaciones simultáneas en el laboratorio amarillo, en un momento determinado estaban 24 alumnos aplicando la evaluación al mismo tiempo.

Se pidió esa tarde al encargado del laboratorio que dispusiera lo que fuera necesario para que la evaluación pudiera aplicarse en los tres laboratorios (amarillo, rojo y morado) simultáneamente para así aprovechar y aplicar la evaluación a cualquier grupo que tuviera clases en los laboratorios.

6 de Mayo de 2000, Sábado

Este día no se aplicarán evaluaciones pero se ofreció ayuda al encargado de los laboratorios para realizar la conexión de los tres laboratorios de tal forma que se pudiera ejecutar la evaluación desde cualquiera de ellos.

El encargado de los laboratorios reportó que había problemas con el *Gateway*, los laboratorios comparten el uso de Internet mediante el uso de dicho dispositivo y es mediante este mismo

que pueden compartir información entre ellos, su malfuncionamiento descartó la idea de que pudiera aplicarse simultáneamente.

El encargado de los laboratorios comentó que el problema podría ser debido a una falla en la comunicación en el cableado, explicó que ya había llamado para que fueran a revisar pero que no sabía exactamente cuando estaría resuelto ese problema.

**Resultados:** La prueba fue un éxito se encontraron problemas en la aplicación de la evaluación de manera simultánea en los tres laboratorios de cómputo en la universidad.

#### **Opciones correctivas:**

Viendo la incierta probabilidad de que el *Gateway* funcionara para la próxima semana, se llegó a la conclusión de que únicamente había dos posibilidades para aplicar la evaluación a grupos que sobrepasaran el cupo del laboratorio amarillo:

- Dividir el grupo en dos o más grupos de veinticuatro personas y que pasaran a contestarlo en diferentes horas.
- Instalar una copia del programa en diferentes laboratorios y después pasar los resultados a una base de datos primaria.

#### **Procedimiento correctivo:**

La división de los grupos resultaría en una pérdida de tiempo tanto para la persona encargada de aplicar los exámenes así como para los alumnos, que tendrían que esperar su turno para contestar las evaluaciones por ello se seleccionó la opción de ejecutar diferentes copias del

programa en cada laboratorio y luego unir los resultados. Lamentablemente sólo se podían aplicar evaluaciones en los laboratorios amarillo y morado por que en el laboratorio rojo aún no habían sido instalados los controladores necesarios.

A el sistema de evaluación se le tuvo que crear un nuevo módulo de extracción de datos de fuentes externas, esto incrementó el tiempo de desarrollo, y creó la necesidad de probar dicho módulo.

**Evaluación de las correcciones:** Las correcciones fueron satisfactorias, se logró aplicar la evaluación en los laboratorios morado y amarillo, en el laboratorio rojo no fue posible debido a que aún no se instalaban los controladores necesarios para la ejecución de la evaluación.

## Prueba #7

**Objetivo:** Evaluar la funcionalidad del sistema ejecutado simultáneamente en dos laboratorios, verificar el acceso, almacenamiento y extracción de los datos, así como la exactitud de los reportes en estas nuevas condiciones. También se evaluará el nuevo módulo de extracción de datos externos y la facilidad de uso del sistema.

**Fecha:** 8 y 9 de Mayo del 2000

**Descripción:** 8 de Mayo de 2000. Se aplicó la evaluación a los alumnos de 10° semestre de informática, es un grupo de 12 personas y no fue necesario aplicarlo aún utilizando dos laboratorios a la vez.

Al finalizar la evaluación se revisaron los resultados con el asesor de pedagogía y quedó establecido que las evaluaciones del día siguiente que se aplicaran a grupos numerosos tendrían que realizarse en dos laboratorios separando la base de datos y haciendo uso del nuevo módulo del sistema para unirlos al final de las evaluaciones.

9 de Mayo de 2000

Se instalaron los módulos actualizados del sistema de evaluación en el laboratorio amarillo, en general la modificación consistía en una mejora en el manejo del archivo de ruta de acceso "dir.cfg" y la posibilidad de cancelar la ejecución de la evaluación al teclear una contraseña, esta característica es muy útil cuando se prueba el sistema y cuando por error se ejecuta el programa de evaluación, por ejemplo dos veces en una misma máquina sin querer. Este día se

terminó de capturar los nombres del maestro, las materias que imparten y los grupos a los que dan clases. Se aplicarían evaluaciones a tres grupos, uno de ellos utilizaría dos laboratorios.

Sería más simple utilizar el laboratorio amarillo y el rojo, por estar adyacentes uno de otro, para las evaluaciones que tuvieran que hacerse en dos laboratorios, pero como las computadoras del laboratorio rojo no tenían instalados los controladores necesarios para ejecutar el programa, se descartó la idea y se utilizó el laboratorio morado.

En el laboratorio morado solamente 11 de las computadoras tenían los controladores necesarios para ejecutar el programa.

La primera Evaluación del día fue a las 7 de la noche y se aplicó a los alumnos de 8º semestre de administración. Para aplicar la evaluación en los dos laboratorios, primero se hizo una copia del programa y de las bases de datos. La copia de la base de datos de respuestas que se utilizaría en el laboratorio morado estaba vacía.

Debido a que el número de alumnos superó la capacidad del laboratorio amarillo, se trasladó a los alumnos que no tenían computadora al laboratorio morado, en total solamente fueron tres alumnos los que contestaron la evaluación en este laboratorio.

En esta prueba por vez primera se aplicó a alumnos de carreras distintas a la de informática, el asesor de pedagogía comentó que hubo que explicar de manera explícita la forma en que deberían de acceder a la red y de cómo entrar al directorio y ejecutar el programa, esto planteó la necesidad de no sólo incluir la breve plática de acerca de la evaluación, sino incluir otra

donde se explique además como hacer uso de la computadora como entrar a la red y ejecutar un programa.

También se descubrió un problema en el orden de presentación de las evaluaciones:

Como algunos grupos tienen un mismo profesor para dos materias, a la hora de contestar la evaluación, aparecían primero las preguntas orientadas al alumno de las dos materias asignadas al profesor, luego las dos series de preguntas orientadas al profesor y al final las preguntas abiertas, en vez de que aparecieran separadas unas de otras según la materia.

Otro problema que se presentó fue que, de manera equivocada, se iniciaron las evaluaciones en dos computadoras antes de que fuera seleccionada la evaluación de ese grupo, por lo tanto se inició la evaluación con los profesores del grupo anterior... el sistema estaba hecho para que los alumnos no puedan cerrar el programa sin terminar la evaluación, así que fue necesario cancelar la ejecución desde el sistema operativo. Esta equivocación plantó la necesidad de tener opción de cancelar la ejecución, pero que únicamente pudiera realizarla personal autorizado con una contraseña.

Este día se solucionaron los problemas del *Gateway* y las redes pudieron acceder a Internet de nuevo, al finalizar las evaluaciones le pedimos al encargado de los laboratorios que tratara de conectar los tres laboratorios para evitar el tener que copiar la base de datos en cada laboratorio, lo cual aún cuando el programa lo puede manejar, es impráctico y resulta en pérdida de tiempo a final de cuentas porque hay que extraer los datos de cada laboratorio..

Pero no sería sino hasta el jueves 11 que se podría probar de nuevo la conexión de los tres laboratorios, mañana es 10 de mayo.

**Resultados:** La prueba fue un éxito.

Se detectó la necesidad de explicar a los alumnos de manera detallada como iniciar la sesión de red y además como entrar al programa, esto es sobre todo necesario en grupos que no pertenecen a la carrera de informática, sobre todo aquellos que son de 5 semestre para arriba, porque también nos dimos cuenta que las nuevas generaciones, aún cuando no son de la carrera de informática, tienen mayor experiencia con las computadoras.

Hubo un problema al tratar de cancelar ejecución errónea de las evaluaciones.

Por otro lado el nuevo módulo de recuperación creado para extraer los datos de fuentes externas funciona a la perfección.

#### **Opciones correctivas:**

Problemas con el inicio y ejecución de las evaluaciones por parte de los alumnos:

- Iniciar la evaluación en cada computadora antes de que lleguen los alumnos, para que los alumnos solamente tengan que llegar al laboratorio y contestar.
- Explicar inicialmente paso a paso como deben acceder a la red, entrar su login, contraseña. Después una vez que ya están conectados,, debe explicarse como han de ejecutar el programa. Y Finalmente como utilizar el programa y que deben de hacer cuando este finalice.

Problema de cancelación de evaluaciones iniciadas:

- Modificar el código para que acepte cancelación después de introducir una contraseña.



**Procedimiento correctivo:**

Se dio una breve introducción de cómo debían ejecutar el sistema y la forma de utilizarlo.

Se modificó el código del sistema para que aceptara la cancelación de la ejecución de las evaluaciones después de introducir una clave de acceso, de tal manera que sólo el personal autorizado podría cancelarlas.

**Evaluación de las correcciones:** Las correcciones fueron satisfactorias, los alumnos recibieron instrucciones explícitas de cómo entrar al programa, y la modificación al código del sistema para que se pudieran cancelar las evaluaciones fue acertada.

NOTA: Del 9 al 19 de Mayo se continuó con la utilización del sistema y no se presentó ningún problema en la ejecución.

## Prueba #8

**Objetivo:** Determinar la exactitud de los reportes modificados.

**Fecha:** 19 de Mayo de 2000

**Descripción:** Ya Sólo faltan cuatro grupos a los cuales aplicar la evaluación.

Aún no se han podido conectar los tres laboratorios para utilizar una sola base de datos en todas las evaluaciones, por consiguiente se ha instalado el sistema por separado en cada uno de los laboratorios, se aplica la evaluación en cada uno y después paso las bases de datos de las evaluaciones a la base de datos principal.

La mayoría de las computadoras del laboratorio morado pueden ya ejecutar el sistema.

En el laboratorio rojo ya se ha instalado el programa en 13 computadoras, pero faltan unas 10.

Como resultado de las evaluaciones canceladas por errores en la ejecución, algunas evaluaciones (5 en total) contenían Sólo parte de la evaluación contestada en su totalidad, por lo anterior, los promedios (que eran obtenidos tomando en cuenta el total de evaluaciones iniciadas) tenían errores en el cálculo. Como sería imposible determinar exactamente cuales evaluaciones son las inconclusas para así eliminarlas, entonces tuvo que modificarse el módulo de procesos para que calculara los promedios basándose en el número de usuarios que contestaban cada una de las preguntas y no en el número total de evaluaciones iniciadas.

**Resultados:** La prueba tuvo éxito. Se encontró un problema con el promedio obtenido en los reportes debido a que algunas evaluaciones no eran terminadas en su totalidad por problemas en la ejecución.

**Opciones correctivas:**

- Modificar el código del módulo de reportes para que realizara los cálculos en base a cada respuesta contestada y no al total de evaluaciones iniciadas.
- Iniciar de nuevo la evaluación del grupo en que se aplicaba el sistema y borrar los datos de las evaluaciones hasta entonces terminadas.
- Alimentar al sistema con la clave del alumno que ejecutará la evaluación.

**Procedimiento correctivo:**

Iniciar de nuevo la evaluación implicaba que todos reiniciaran la evaluación, lo cual implicaría mayor tiempo en la evaluación y el descontento de los alumnos, por ello esta opción queda descartada.

Establecer el nombre del alumno para que entrara con su clave a contestar la evaluación acabaría con el anonimato y las respuestas pudieran ser afectadas por este factor. Esta opción queda descartada.

Se modificó el código del módulo de reportes para que realizara los cálculos en base a cada respuesta contestada y no al total de evaluaciones iniciadas.

**Evaluación de las correcciones:** Las correcciones realizadas fueron satisfactorias, los reportes muestran información exacta.

Seguridad de la información durante las pruebas del sistema

El sistema fue instalado en estaciones de trabajo conectadas a las redes de computadoras de la Universidad, los archivos del programa estaba en directorios públicos para que pudieran ser accedidos.

Los laboratorios de computación de la universidad están abiertos a los estudiantes de manera libre, el requisito para entrar a trabajar con las estaciones de trabajo es presentar una credencial que lo acredite como alumno inscrito y que existan estaciones de trabajo disponibles.

En caso de que algún usuario causara un daño al sistema de evaluación de manera premeditada y en horas en que no estuvieran los encargados de la evaluación no se sabría exactamente quien fue el culpable.

En varias ocasiones de manera malintencionada o por descuido los usuarios eliminaron el módulo de ejecución de la evaluación de los directorios de las estaciones de trabajo.

Las bases de datos y el módulo principal estaban en un directorio compartido, en el cual se podía tener acceso a los datos para modificarlos pero no se podían eliminar los archivos. Para seguridad de la información se respaldaban los datos al finalizar cada evaluación y en cada nueva aplicación se utilizaba una nueva copia del sistema para evitar problemas en caso de que los datos hubiesen sido modificados.

Esta situación pudiera omitirse si se crea una cuenta especial para la aplicación de las evaluaciones.

### **6.11.1 Conclusiones**

Antes que nada quiero aclarar la diferencia que existe entre las pruebas del sistema completo y en ambiente real a las pruebas de sistema en desarrollo.

#### **Las pruebas del sistema en desarrollo**

Son aplicadas al programa mientras se está desarrollando, es decir, mientras se programa. Estas pruebas se llevan a cabo en cada módulo del sistema y luego entre dichos módulos. Estas pruebas sirven para verificar la funcionalidad del software.

#### **Las pruebas del sistema completo**

Las pruebas del sistema utilizaron un gran porcentaje del tiempo de implantación del sistema, pero arrojaron varios resultados muy importantes para la mejora del sistema. Hay una significativa diferencia entre implantar un sistema después de realizar pruebas e implantarlo sin efectuarlas.

A continuación expongo, de acuerdo a la experiencia recopilada en este proyecto de tesis, algunas de las ventajas y desventajas de realizar pruebas al sistema antes de implantarlo:

#### **VENTAJAS**

- Se verifica la funcionalidad de cada

#### **DESVENTAJAS**

- Se incrementa el tiempo de término

- |  |   |
|--|---|
| <p>módulo</p> <ul style="list-style-type: none"> <li>• Se verifica la operación de los módulos interrelacionados</li> <li>• Se revelan necesidades no identificadas en el análisis.</li> <li>• Se encuentran errores improbables</li> <li>• Se mejoran la ejecución de los módulos</li> <li>• Se evitan retrasos por problemas en el sistema una vez implantado</li> <li>• Se toman en cuenta variables ambientales imposibles de detectar en ambientes controlados</li> </ul> | <p>del sistema</p> <ul style="list-style-type: none"> <li>• Se incrementa el costo del sistema</li> <li>• Se da pie a nuevas modificaciones no establecidas por el usuario al inicio del desarrollo.</li> </ul> |
|--|---|

A pesar de algunas desventajas de la aplicación de las pruebas del sistema, su realización permite mejorar el sistema y afinar detalles que permiten su mejor utilización. El hecho de haber aplicado las pruebas del sistema permitió mejorar la eficiencia de algunos módulos, no es lo mismo probar el sistema con uno o dos usuarios de manera concurrente que con grupos de más de 25 personas.

## NOTAS:

- Seguramente se cuestionará el porqué no se dejaron las respuestas en lugar de respaldarlas y eliminarlas en cada evaluación. La justificación se encuentra en que alguien podría ejecutar la evaluación mientras no estamos en el laboratorio, y eso alteraría los resultados, por otro lado no podemos quitar el módulo de evaluación de las computadoras porque tendríamos que copiarlo en cada computadora antes de iniciar la evaluación.
- Cuando las computadoras del laboratorio amarillo no se reinician antes de ejecutar la evaluación, es muy probable que dos computadoras (de las 26 que hay en total) se traben durante la ejecución de la evaluación. Las computadoras que se traban no son siempre las mismas y varía el momento en que se traban. Haciendo pruebas llegamos a la conclusión de que solamente cuando las computadoras eran reiniciadas antes de la
- evaluación no presentaban problemas en la ejecución.

En este capítulo se trató todo lo referente al sistema correspondiente a esta tesis, como se puede observar, una de las cuestiones más importantes al planear un sistema es corroborar su factibilidad, tanto técnica como económica. Es común que sistemas terminen en fracaso debido precisamente a la falta de factibilidad o validez real, ya sea por que no funcionan en los ambientes reales o porque su costo se eleva demasiado.

Si tuviera que definir cual es la parte más importante de esta tesis, tendría, sin lugar a dudas, que mencionar al diagrama de la base de datos (Pág. 87) por que es en base a este que el sistema está construido, y en él se aprecian claramente muchos de los aspectos necesarios para tener un buen diseño.

El diagrama de clases muestra las relaciones entre los objetos de la base de datos y el pseudo código nos da una explicación de las operaciones internas de los módulos que conforman el sistema. Una vez más queda demostrado que los grandes sistemas (me atrevo a clasificar mi sistema como grande, basándome en la apreciación de mi joven experiencia como programador) pueden llevarse a cabo dividiendo el todo en pequeñas partes, lo más simple posibles, lo cual desde luego forma parte del paradigma orientado a objetos.



# Conclusiones

---

Para aquellos que mejorarán este sistema o incluso crearán otro:

*"El único día fácil para un hombre exitoso es el día de ayer" (NAVY S.E.A.L.S Canal TLC)*

Yo creo que el sistema cumple varios objetivos y de cada uno de ellos se pueden derivar conclusiones, por ello he decidido separarlas en:

## **Conclusiones personales:**

El Sistema de evaluación del desempeño de los profesores me ha dado no solamente la satisfacción de haber aplicado mis conocimientos adquiridos durante la carrera, sino además cumplir con mi función como Licenciado de Informática: ser un facilitador de información, en este caso en beneficio de la E.A.C.I.

Durante toda mi carrera tuve la oportunidad de realizar diferentes programas, al terminar cada uno de ellos descubría con asombro dos cosas, que cada vez programaba mejor y que los programas anteriores no eran tan buenos como yo creía.

En este programa de mi tesis pude realizar mi tareas con mejores herramientas que las que dispuse al principio, mejores técnicas y métodos de programación y sin embargo, al igual que

siempre, ahora que lo he terminado y lo veo como algo realizado sé que aún cuando el sistema opera correctamente bien puede mejorarse.

Este proyecto tiene un conjunto de conclusiones, las cuales aun cuando indican el éxito del desarrollo bien pueden dividirse en las siguientes áreas:

### **Conclusiones de efectividad y eficiencia del sistema**

El sistema desarrollado cumple perfectamente con las expectativas con las que fue desarrollado: facilitar la obtención de los resultados de las evaluaciones, con mayor eficiencia y efectividad. No solamente se consiguen los reportes de resultados, sino que se tiene la plena seguridad de que los datos son calculados de manera exacta y sin lugar a errores.

- El sistema actual utiliza solamente el 24.9 % de los recursos necesarios en el anterior sistema manual.
- Considerando los recursos materiales se puede llegar también a la conclusión de que no sólo ahorra tiempo y dinero, sino que además implica un beneficio ambiental al evitar la utilización de 3600 hojas de papel.
- Se reduce en 50% el personal necesario para la aplicación de las evaluaciones. Se reduce en un 70% el tiempo requerido de trabajo personal para obtener las evaluaciones.

El tiempo promedio para responder la evaluación es de 10 minutos por persona en grupos no experimentados con el uso de la computadora. En grupos de informática se reduce hasta 7 minutos.

Se tienen más o menos tiempos aproximadamente iguales en responder la evaluación de manera manual que en el equipo de cómputo, lo que indica que el sistema no tiene un efecto adverso a la hora de contestar ni provoca incertidumbre en la forma de contestarlo.

### **Conclusiones económicas del sistema**

El aplicar la evaluación de desempeño de los profesores utilizando sistema manual tenía un costo neto de \$5106 (510.6 Dlls) pesos y el nuevo sistema tiene un costo de \$1276 (127.6 Dlls) hay un ahorro de 75% en los costos. El sistema prueba totalmente su factibilidad económica y ahorra recursos en la aplicación.

### **Conclusiones del desarrollo**

1. El tiempo programado para la entrega jamás será suficiente. Siempre hay que calcular con holgura todas las entregas y pruebas del sistema, cabe mencionar que esto se torna crítico para los estudiantes que no sólo tenemos que preocuparnos por la Tesis, sino además por las demás asignaturas de la carrera.
2. Al crear un sistema en el que se manejara información confidencial, hay que separar los orígenes de los datos, de tal forma que aquellos críticos se encuentren separados de las tablas que les den dicho sentido, en el caso particular del sistema de mi tesis separé las tablas de los profesores y las evaluaciones de las tablas de las respuestas, de tal forma que sólo teniendo todo el conjunto se podría tener información servible porque individualmente, o teniendo incluso dos de ellas, no aportarían ningún dato significativo.

3. Según las reglas de formalización de Bases de datos. Esto ayuda muchísimo a la hora de separar en módulos el sistema y también facilita el mantenimiento y modificaciones
4. Solamente hay que abrir la base de datos cuando se vaya a acceder a ella. Es una verdadera molestia estar abriendo y cerrando la base de datos en cada módulo del sistema, pero créanme, esa molestia se compensa cuando se evitan errores de acceso a la base de datos, sobre todo en ambientes multiusuario, ni hablar del ahorro de recursos del sistema.
5. A la hora de programar háganlo mediante módulos lo más indivisibles posibles, aún cuando se generen muchos módulos pequeños. Luego pueden formar módulos más complejos uniéndolos. Esto es especialmente útil en consultas compuestas. O cuando un resultado particular está formado por un conjunto de operaciones simples.
6. De ser posible el manual de usuario debe realizarse conforme se desarrollan los distintos módulos del sistema, de esta manera es más fácil hacer hincapié en los detalles pequeños, que son fácil de olvidar después de un tiempo.
7. Las pruebas del sistema deben incluir varios cambios en la ejecución y en diferentes ambientes, también debe probarse al máximo de la capacidad a la que se planea poner el sistema. En el caso particular de esta Tesis, el sistema se comportó muy bien con menos de 20 usuarios, pero cuando metimos la carga completa del laboratorio, es decir 26, presentó ciertas anomalías que se pudieron corregir a tiempo.
8. Mi opinión particular acerca del diseño y análisis es el siguiente: Aconsejo firmemente el paradigma de ingeniería de creación de prototipos, primero porque siempre he estado acostumbrado a acompañar el trabajo de papel (diseño y análisis) con un poco de aspectos prácticos (programación). Muchos dirán que es preferible sentarse y

planear bien todo antes de empezar cualquier sistema, pero al menos en los sistemas pequeños y medianos, me ha resultado muy bien comenzar con la programación después de un breve análisis, claro que no debe comenzarse con los módulos complejos, se comienza con los módulos simples y se comienzan los módulos más complejos conforme se avanza en el análisis. La creación de prototipos muestra a los clientes el sistema antes de que sea demasiado tarde para hacer modificaciones, y también plantea los problemas reales antes de terminar el sistema. Este como pueden ver es un arma de doble filo, pues yo mismo he tenido que implementar módulos cuando el sistema está casi terminado por realizar módulos complejos sin tener el debido análisis realizado. Un consejo muy importante para evitar problemas de análisis y diseño utilizando prototipos es tener a la mano toda la información que debe dar como salida el sistema, llámese reporte impreso, gráfico, etc.. entonces debe verificarse que el diseño de la base de datos pueda originar todas esas salidas.

9. No hay un sistema perfecto a prueba de fallas. Todo sistema puede mejorarse hasta infinito.. lo que quiere decir que siempre habrá aspectos que puedan mejorarse, diseño, análisis, técnicas de programación, lenguaje elegido para su desarrollo... todo en pocas palabras. Nosotros como programadores debemos evaluar que módulos deben ser analizados minuciosamente y cuales no, esto se decide dependiendo de su importancia, pues probar un módulo de manera demasiado exhaustiva puede consumir mucho tiempo y siempre quedarán posibilidades de combinación de situaciones que puedan presentar error.
10. Por último sólo me queda recomendar que antes de iniciar un proyecto hay que cerciorarse personalmente y probar la funcionalidad de la plataforma en la que se

instalará el sistema, en mi tesis tuvimos algunos retrasos debido a que no se calculó desde el principio las capacidades reales de los laboratorios y en caso concreto de la comunicación existente entre ellos.

### **Conclusiones de aprendizaje**

Este desarrollo me ha permitido poner en práctica muchos de los conocimientos aprendidos durante la carrera, no solamente involucré, como puede pensarse, los elementos computacionales, a continuación pongo un listado de las materias vistas en la carrera y los conocimientos utilizados de ellas en esta tesis, lo cual permite darse cuenta que ninguna materia del plan de estudios sale sobrando:

- Informática: Teoría de sistemas, análisis, desarrollo, diseño, implementación
- Ingeniería de software: Paradigmas de ingeniería, modelización, análisis y diseño.
- Administración: Optimización de recursos, dirección y control
- Recursos humanos / personal: Tratar con las personas involucradas en el proyecto.
- Operaciones: Gráficas de desarrollo, cronología del proyecto.
- Taller de Excel: Obtención de reportes.
- Taller de visual Basic: Lenguaje utilizado para desarrollar este sistema
- Taller de lenguaje C++ : programación orientada a Objetos
- Curso de bases de datos: diseño de las bases de datos.
- Arquitectura de computadoras: Utilización, optimización y selección de recursos informáticos. Identificación de causas de problemas.

- Taller de J++: Acceso a datos de Microsoft
- Estadística: Comprensión de procesamiento y utilidad de los reportes de resultados.
- Comunicaciones: Manejo de programas en red.
- Metodología de la investigación: Bases para el desarrollo de la tesis.
- Contaduría: Análisis de costo beneficio.
- Derecho: Legalidad del sistema
- Sistemas operativos multiusuarios: Instalación y ejecución del sistema en el sistema operativo de red.

Y por último la más importante

### **Conclusiones de la hipótesis**

Queda totalmente demostrada la hipótesis y se afirma positivamente que:

La implementación de un sistema informático para la aplicación de las evaluaciones del desempeño de los profesores produce resultados de manera más eficaz, eficiente y exacta que el sistema manual.

# Bibliografía

---

**Ceballos Sierra, Fco. Javier**

“Enciclopedia de Microsoft Visual Basic 4”

Editorial Alfaomega

México 1992

**Diccionario Dyson de Computación**

ISBN: 980-346-010-2

©Enigma Editorial C.A.

**Diccionario Léxico Hispano**

W.M. Jackson Inc. Editores

8va. Edición 1980

**Enciclopedia Didáctica de Computación**

Editorial Océano

España 1999

**Enciclopedia Encarta 2000**

Microsoft Corp.

1999 Español



**Freedman, Alan**

“Diccionario de Computación”

The Computer Language Company Inc.

1981-1993

Ver 5.2

**Greg Perry**

“Aprendiendo Visual Basic 6”

Editorial Prentice Hall

México 1999

**Introducción Windows95**

Microsoft Corp.

México 1995

**June Jamrich Parson y Dan Oja**

“Conceptos de Computación (Edición Estándar)”

Thomson Editores

Edición en Español

México, 1997

**Kendall y Kendall**

“Análisis y Diseño de Sistemas”

Prentice Hall

3ª Edición

México 1998

**Lemay, Laura y Perkins, Charles L.**

“Aprendiendo JAVA 1.1”

Editorial Prentice Hall

México 1997

**Martin Kenley y Cols.**

“Aprendiendo Windows NT 4”

Prentice Hall

**Pressman, Roger S.**

Ingeniería de software Un enfoque práctico

McGraw Hill

3era Edición

España, Madrid 1993

**Robert Cecil Martín**

Designing Object Oriented C++ Applications

Prentice Hall

Primera Edición

Estados Unidos , 1995

**Schildt, Herbert**

“C++ Manual de Referencia”

Editorial Mc Graw Hill

México 1996.

**Silberschatz, A**

“Sistemas Operativos “

Tercera Edición

Editorial Addison Wesley

Estados Unidos 1994

**Strostrup, Bjarne**

“El lenguaje de programación C++”

Segunda edición.

Editorial Addison Woesley

Estados Unidos 1993

**Wilson, Jim**

“Mecánica Popular (Revista)”

“Fuego amistoso”

Octubre

México 1996

# Hemerografía

---

## **CDWare (Revista)**

Editorial LARPRES

España 1996

## **PcComputing En Español (Revista)**

Editorial Televisa

1998

## **PcMagazine En Español (Revista)**

Editorial Televisa S.A. de C.V.

1998

## **PcMedia (Revista)**

Editorial NESS

1998

## **Personal Computing (Revista)**

Publicaciones Sayrols

México 1998

# Anexos

---

## ANEXO 5a

### PROPUESTA DE EVALUACIÓN DE LA PRÁCTICA DOCENTE

#### Introducción.

En el presente documento se expone el proyecto y el funcionamiento del proceso de evaluación de la práctica docente. Lo que se pretende es establecer claramente los lineamientos que nos permitan reconocer el ¿por qué?, el ¿para qué? y el ¿cómo? Evaluar la práctica del docente y de esta forma proceder con un amplio conocimiento de causa a realizar esta actividad.

#### La evaluación del docente: una necesidad.

En los últimos años ha cobrado mayor importancia la actividad de la evaluación en la educación, siendo ampliada la dimensión de su intervención y reconocida la utilidad de esta práctica.

Unos de los rubros donde se ha hecho más patente la necesidad de la evaluación ha sido la docencia y especialmente en el nivel superior. Cada vez son más las universidades que buscan mecanismos para darse cuenta de si están alcanzando sus objetivos y de qué resultados están obteniendo en relación a la formación profesional que proporcionan.

Es a partir del interés puesto en este tema que se ha podido pasar de la función de control administrativo, atribuida tradicionalmente a la evaluación de la docencia, a la de mejoramiento de la actividad de enseñanza, que es posible cuando el profesor recibe información referente al impacto de sus estrategias instruccionales.

En este sentido, Rueda y Rodríguez afirman que la evaluación puede ser considerada como “un conjunto de acciones sistemáticas que tienden a lograr un conocimiento sobre la realización de una actividad, en este caso la docencia, para su comparación con un conjunto de criterios previamente determinados y cuyo propósito principal es su mejoramiento. La evaluación así pensada se concibe como formativa”(Rueda, Nieto comps.,1996:13).

El camino que se ha recorrido hasta el momento y que ha permitido esta ampliación de la visión de la evaluación docente ha sido largo y todavía falta mucho por recorrer pues si bien, ahora se entiende que los procesos de evaluación de fenómenos educativos como la docencia tienen como función principal lograr una calidad mayor de la enseñanza, todavía a decir de Beltrán y Segura, no existe consenso en lo concerniente a qué aspectos constituyen la que significa ser un “buen profesor” así como en los procesos e instrumentos que permiten valorar esta actividad.

Por otro lado, para obtener una idea clara del funcionamiento docente no es suficiente con mediciones de orden cuantitativo sino que es necesario considerar como proponen Rueda y Rodríguez, aspectos cualitativos referidos a las características del contexto de la evaluación que aportarían elementos de análisis orientados a la definición de criterios acordes con el

profesor en su contexto y no al “debe ser”. Esto significa que la actividad del profesor debe ser evaluada a la luz de las condiciones en las cuales aquélla se lleva a cabo, lo cual representa una dificultad más que se añade al problema en cuestión y que nos deja claro cuán complejo es un proceso de este tipo.

Ahora bien, concebida la evaluación docente desde esta perspectiva, las finalidades que hacen necesaria su práctica debe ser: motivar para mejorar, corregir o perfeccionar; como medida que oriente la promoción y desarrollo de los profesores y sobre todo como retroalimentación de la práctica docente, obviamente que en estas finalidades está incluida la decisión de permanencia de sus funciones académicas, sin embargo es un a más sin otorgarle mayor peso que a las ya mencionadas.

Generalmente se ha recurrido a la opinión del alumno para obtener la información de la práctica docente pero existen más mecanismos para conseguir esta información, como la opinión del coordinador, la del director y la auto evaluación.

Basándonos en las prescripciones anteriores y tomando como modelo el Sistema de Evaluación del Desempeño Docente empleado actualmente en el ITESO (Instituto tecnológico de estudios superiores de occidente) proponemos el siguiente procedimiento:

1. Dar a conocer a los profesores, desde el inicio del semestre, el sistema de evaluación de su desempeño, su finalidad y los instrumentos a utilizar.



2. Cuando se acerque el final del semestre (unas cuatro semanas antes), aplicar el instrumento de apreciación de la practica docente que ya existe y que contestan los alumnos. Previamente se les explicará con qué finalidad se hace esto y bajo qué condiciones debe realizarse.
3. Procesamiento de esta información, tratamiento estadístico.
4. Proporcionar a los profesores el instrumento de auto evaluación de la práctica para que lo contesten.
5. Convocar a una reunión general o por carreras para entregar los resultados del instrumento aplicado a los alumnos.
6. Tener una entrevista con cada maestro(el director, subdirector o coordinador del área pedagógica de apoyo) para analizar juntos los resultados de las evaluaciones para, en los casos necesarios, elaborar un plan de mejoramiento de la práctica a aplicar el siguiente semestre.
7. Someter a crítica y mejoramiento el procedimiento empleado para la evaluación de la práctica docente así como los instrumentos. Esta revisión debe incluir la opinión de los profesores así como de los coordinadores y directivos de la escuela

(Sergio Quintero Vizcarra documento interno EACI-UDV)

## ANEXO 5b

### **Procedimiento para la puesta a punto del sistema en cada laboratorio:**

#### **Laboratorio morado:**

- Reiniciar la computadora
- Entrar con login normal (M + número de equipo)
- Ejecutar alguna aplicación del Office 2000 para que sea reconocida la unidad F, en la cual se encuentran las bases de datos.
- Cerrar la aplicación del Microsoft Office.
- Entrar al directorio de la unidad F:
- Si no está instalado el sistema, hay que instalarlo, copiando aquí el programa principal y las bases de datos
- Ejecutar el módulo principal del sistema de evaluación.
- Respalda la base de datos de respuestas
- Eliminar todas las respuestas
- Seleccionar el grupo al que se le va a aplicar la evaluación.
- Si aún no está instalado el módulo de evaluación del sistema, entonces tiene que copiarse el archivo evaluación.exe en todas las computadoras en el directorio c:\Temp. Y establecer en el archivo de configuración "dir.cfg" que la ruta será F:
- Ahora todos los que vayan a resolver la evaluación pueden ejecutarla en el directorio c:\Temp.\evaluación.exe
- Una vez terminada la evaluación, se respalda la base de datos y se asigna el grupo al que se le aplicará la siguiente evaluación.

- Al final de todas las evaluaciones se agregan todos los respaldos de las evaluaciones aplicadas a la base de datos principal.

#### **Laboratorio rojo:**

- Reiniciar la computadora
- Entrar al directorio de la unidad G:
- Si no está instalado el sistema, hay que instalarlo, copiando aquí el programa principal y las bases de datos
- Ejecutar el módulo principal del sistema de evaluación.
- Respalda la base de datos de respuestas
- Eliminar todas las respuestas
- Seleccionar el grupo al que se le va a aplicar la evaluación.
- Si aún no está instalado el módulo de evaluación del sistema, entonces tiene que copiarse el archivo evaluación.exe en todas las computadoras en el directorio c:\Temp. Y establecer en el archivo de configuración "dir.cgf" que la ruta será G:
- Ahora todos los que vayan a resolver la evaluación pueden ejecutarla en el directorio c:\Temp.\evaluación.exe
- Una vez terminada la evaluación, se respalda la base de datos y se asigna el grupo al que se le aplicará la siguiente evaluación.
- Al final de todas las evaluaciones se agregan todos los respaldos de las evaluaciones aplicadas a la base de datos principal.

### **Laboratorio Amarillo:**

- Reiniciar la computadora
- Entrar con login normal (A + número de equipo)
- Entrar al directorio de la unidad O:
- Si no está instalado el sistema, hay que instalarlo, copiando aquí el programa principal y las bases de datos
- Ejecutar el módulo principal del sistema de evaluación.
- Respaldar la base de datos de respuestas
- Eliminar todas las respuestas
- Seleccionar el grupo al que se le va a aplicar la evaluación.
- Si aún no está instalado el módulo de evaluación del sistema, entonces tiene que copiarse el archivo evaluación.exe en todas las computadoras en el directorio c:\Temp. Y establecer en el archivo de configuración “dir.cfg” que la ruta será O:
- Ahora todos los que vayan a resolver la evaluación pueden ejecutarla en el directorio c:\Temp.\evaluación.exe
- Una vez terminada la evaluación, se respalda la base de datos y se asigna el grupo al que se le aplicará la siguiente evaluación.
- Al final de todas las evaluaciones se agregan todos los respaldos de las evaluaciones aplicadas a la base de datos principal.