

# UNIVERSIDAD DE PINAR DEL RÍO

## “HERMANOS SAÍZ MONTES DE OCA”



*Facultad de Informática y Telecomunicaciones*  
*En Convenio con la Universidad Técnica de Cotopaxi*

**PROYECTO DE DIPLOMA EN OBTENCIÓN AL TÍTULO DE  
INGENIERO EN INFORMÁTICA Y SISTEMAS COMPUTACIONALES**

**TEMA:**

“Sistema de Almacenamiento de Información Relevante de la  
Facultad de Informática y Telecomunicaciones”  
(SAIRFIT)

**AUTORES:**                    **ÁLVAREZ ZURITA CÉSAR FERNANDO**  
**ESCOBAR BANDA LUIS EDUARDO**

**TUTOR:**                        **DRA. MAGDALENA P. MAZÓN HERNÁNDEZ**

**Pinar del Río - Cuba**  
**2005**

# INDICE

INTRODUCCIÓN .....	1
<b>CAPITULO I. ESTADO DEL ARTE .....</b>	<b>4</b>
1.1. Fundamentación del Problema.....	4
1.2. Objeto de Estudio.....	4
1.2.1. Descripción del Objeto de Estudio.....	5
Misión.....	5
Objetivo Científico -Técnico General.....	6
1.2.2. Ubicación dentro de la Estructura Organizacional y Territorial.....	6
Organigrama de la Facultad de Informática y Telecomunicaciones.....	7
1.2.2.1. Estructura de los Departamentos con los que cuenta la Faculta...	8
1.2.3. Funciones principales que realiza la F.I.T.....	9
1.2.4. Estrategia Maestra Principal de la F.I.T.....	10
Enfoque Integral para la labor Educativa y Político-Ideológica .....	10
Estrategias Específicas.....	11
1.2.5. Niveles de Automatización.....	16
1.2.6. Volumen de Información de Entrada y Salida.....	16
1.2.7. Principales Deficiencias y Problemas.....	18
1.2.8. Definición del Objetivo del Sistema.....	19
1.2.8.1. Objetivo General.....	19
1.2.8.2. Objetivos Específicos.....	19
1.2.9. Requerimientos Técnicos.....	20
1.2.9.1. Recursos Humanos.....	20
1.2.9.2. Recursos Tecnológicos.....	20
1.2.10. Comparación del Sistema nuevo con otros similares.....	21

1.2.11. Análisis de Factibilidad.....	22
1.2.11.1. Económico.....	22
1.2.11.2. Técnico.....	27
1.2.11.3. Operativo.....	28
<b>CAPITULO II. ANÁLISIS Y DISEÑO DE SAIRFIT.....</b>	<b>29</b>
2.1. Metodología Utilizada.....	29
2.1.1. Ciclo de Vida de un proyecto en Met Visual E.....	30
2.2. Tabla de Eventos.....	32
2.3. Diagrama de Contexto.....	34
2.4. Diagrama de Flujo de Datos.....	34
Simbología Utilizada.....	35
2.5. Diagrama de Descomposición .....	35
2.6. Descripción Textual de los Flujos de Datos.....	38
2.7. Grafo Conversacional.....	43
2.8. Diagrama Entidad Relación.....	44
2.8.1. Entidad.....	44
2.8.2. Atributo.....	45
2.8.3. Relación.....	45
2.8.3.1. Relaciones 1-1.....	45
2.8.3.2. Relaciones 1-n.....	46
2.8.3.3. Relaciones n-n.....	46
2.9. Tendencias y Tecnologías actuales a considerar.....	48
2.9.1. Herramienta Case ER/Studio.....	48
2.9.1.1. Modelo Lógico.....	49
2.9.1.2. Modelo Físico .....	50
2.9.2. Sistemas Gestores de Base de Datos Relacionales (SGBDR).....	50

2.9.2.1. Objetivos del SGBD.....	51
Abstracción de la Información.....	51
Independencia.....	51
Redundancia Mínima.....	52
Consistencia.....	52
Seguridad.....	52
Integridad.....	53
Respaldo y recuperación.....	53
Control de la concurrencia.....	54
Tiempo de respuesta.....	54
2.9.3. Microsoft Access 2000.....	54
2.9.3.1. Características de Microsoft Access.....	54
2.9.4. Motor de Base de Datos Jet 4.....	55
2.9.4.1. DLL(Dynamic Linking Library).....	56
2.9.5. Normalización.....	57
2.9.5.1. Primera Forma Normal 1FN.....	59
2.9.5.2. Segunda Forma Normal 2FN.....	60
2.9.5.3. Tercera Forma Normal 3FN.....	61
2.9.6. Delphi 6.....	61
2.9.6.1. Componentes.....	63
2.9.6.2. Eventos.....	64
2.9.6.3. Desarrollo Visual.....	64
2.9.6.4. Depurador Integrado.....	65
2.9.6.5. Ambiente de Desarrollo Integrado (IDE).....	65
2.9.6.6. Acceso de Delphi a las Base de Datos.....	66
2.9.6.7. ADO/Delphi 6.....	67
2.9.6.8. Objetos ADO.....	68
2.9.6.9. Reportes QReport.....	69

2.9.6.9.1. Componentes QReport .....	69
2.9.6.9.2. Componentes de Banda QReport.....	70
2.9.6.10. Ventajas de Delphi con Respecto a Visual Basic.....	71
<b>CAPITULO III. DESARROLLO DEL SISTEMA.....</b>	<b>73</b>
3.1. Panorámica General de SAIRFIT.....	73
3.2. Objetos Utilizados para crear la interfaz.....	74
3.2.1. Tablas.....	74
3.2.2. Formularios.....	75
3.2.3. Subformularios.....	76
3.2.4. Informes.....	77
3.2.5. Módulos.....	77
3.2.6. Los Controles.....	78
3.2.7. Propiedades del Formulario.....	84
3.2.8. Métodos de los Formularios.....	86
3.2.9. Eventos.....	87
3.3. Interfaz del Usuario.....	88
<b>CONCLUSIONES.....</b>	<b>105</b>
<b>RECOMENDACIONES .....</b>	<b>106</b>
<b>REFERENCIAS BIBLIOGRÁFICAS.....</b>	<b>107</b>
<b>BIBLIOGRAFÍAS .....</b>	<b>108</b>

## ANEXOS

## **RESUMEN**

**Actualmente la Facultad de Informática y Telecomunicaciones de la Universidad de Pinar del Río almacena la información que por caracterizar su desempeño académico, científico y extensionista considera relevante de forma manual, en documentos de papel o en ficheros de Word, teniendo redundancia e inconsistencia en los datos y no posee un sistema que organice toda esa información de manera coherente.**

**El sistema SAIRFIT permite almacenar la información relevante de una forma consistente y sin pérdida, manteniendo el control y organización de la misma.**

**SAIRFIT desarrollado sobre Access 2000 y Delphi posee una página principal con opciones tanto para Administrador como para Usuario, permitiendo al primero acceder y manipular dicha información, mientras que al segundo, el usuario, sólo poder acceder a la misma en forma de consultas o reportes.**

**Con el presente trabajo se logró obtener un software que brinde respuestas rápidas y eficientes en lo que se refiere a búsqueda, almacenamiento y**

**consolidación de la información, útil también para otras facultades dentro y fuera de la Universidad de Pinar del Río.**

## **SUMMARY**

**The faculty of Computer science and Telecommunications of the University of Pinar del Río, store information that is considered relevant due to their character academic, scientific and extensionist in a manual way. They stored paper documents or in Word files; having redundancy and inconsistency in the data and it does not have a organizes system all information in a coherent way.**



**The system SAIRFIT allows storing relevant information in a satisfactory way and without loss, maintaining its control and organization.**

**SAIRFIT developed on Access 2000 and Delphi have a main page with options for Administrator as well as for User, allowing at the first to entry and manipulate this information, while the second, the user only to be able to access the data in consultations form or reports.**

**In addition, with the present work, was possible to obtain software that offers quick and efficient answers in what refers to search, storage and consolidation of the information, useful for other**

**abilities inside of and outside of the Pinar del Río  
University.**

## **INTRODUCCIÓN**

**La información es un recurso primordial, tanto para una organización como para la sociedad en general.**

**La efectividad en el desempeño de un trabajo está determinada por diversos factores, pero para lograr un objetivo planteado mediante un proceso continuo e interactivo de toma de decisiones; sin lugar a dudas, uno de los más importantes factores es la calidad de la información que es capaz de brindar un Sistema Informativo.**

**Mediante la información, el desarrollo de los sistemas informáticos se ha convertido en un pilar fundamental para la automatización de todas las**

**operaciones dentro de las instituciones,  
simplificando las tareas manuales y proporcionando  
información ágil, dinámica y segura.**

**En la Facultad de Informática y Telecomunicaciones  
de la Universidad Pinar del Río continuamente se  
realizan actividades académicas, de investigación,  
deportivas, sociales, etc., emprendidas por el  
personal directivo, docente y estudiantil, que en su  
mayoría sólo quedan registradas en papel o en la  
memoria frágil de quienes las vivieron, conllevando  
a la pérdida, deterioro o manipulación impropia de la  
misma.**

**Por lo anteriormente expuesto nos hemos visto en la necesidad de proponer el desarrollo de un Sistema de Almacenamiento de Información Relevante de la Facultad de Informática y Telecomunicaciones de la Universidad Pinar del Río para que cuente con un registro exacto de las actividades que se han venido desarrollando, útil para la toma de decisiones inmediatas y para el resguardo de la memoria histórica de la Facultad, la Universidad y en sentido general de la provincia de Pinar del Río.**

Para el mejor entendimiento y comprensión del presente trabajo a continuación se detallarán algunos de los capítulos, según a la estructura a seguir:

## CAPÍTULO I

**En el presente capítulo se destacan los puntos importantes relacionados con el objeto de estudio, la caracterización y las funciones principales que cumple, así como los requerimientos técnicos para el nivel de automatización existentes, tratando los problemas y necesidades que tiene la Facultad de Informática y Telecomunicaciones para tratar de dar soluciones a una de sus necesidades como el almacenamiento de la información relevante.**

## CAPÍTULO II

En este capítulo se realizará la respectiva descripción del fundamento teórico de las herramientas utilizadas (Microsoft Access, Borland Delphi 6, Herramientas CASE E/R). Además se mostrarán los aspectos vinculados a la Ingeniería de Sistemas con el Diagrama de Contexto, Diagrama de Flujo de Datos con sus diferentes niveles de descomposición y la respectiva descripción de Flujo de Datos, Descripción Lógica, Modelo Entidad Relación y el diseño del Grafo Conversacional.

## CAPÍTULO III

En este capítulo detallaremos como está constituido el Software y sus aspectos relevantes y fundamentales para una mejor comprensión del mismo.

## **CAPÍTULO I**

### **DESCRIPCIÓN DEL ARTE**

#### **1.1. FUNDAMENTACIÓN DEL PROBLEMA.**

En el presente capítulo se realizará una caracterización del objeto de estudio, sus funciones principales, así como los requerimientos técnicos, niveles de automatización existentes, y algunos problemas y necesidades de la Facultad de Informática y Telecomunicaciones (F.I.T) de la Universidad de Pinar del Río.

#### **1.2. OBJETO DE ESTUDIO.**

**El Objeto de estudio de este trabajo de investigación es la Facultad de Informática y Telecomunicaciones del que se plantea realizar la automatización de la Información Relevante, pues en muchas ocasiones**



**se lo realiza en forma manual en documentos de papel o en hojas de Word teniendo dispersa dicha información, lo cual hace que sea inconsistente, redundante y sin garantías de seguridad.**

### 1.2.1. DESCRIPCIÓN DEL OBJETO DE ESTUDIO.

**La Facultad de Informática y Telecomunicaciones creada en septiembre de 2001 asume el encargo social de formar profesionales integrales capaces de impulsar el desarrollo del territorio con un uso eficiente de las tecnologías de la información y las comunicaciones en medio de las condiciones particulares de la economía cubana.**

**Su fortaleza radica en un colectivo joven, competente, con elevado sentido de pertenencia y comprometido con la responsabilidad política, académica y científica que la sociedad ha puesto en sus manos y que enfrenta con dinamismo y cohesión.**

- **MISIÓN**

**Contribuir al desarrollo y aplicación de las tecnologías de la información y las comunicaciones en el territorio mediante la formación integral de profesionales de la Informática, las Telecomunicaciones y la Electrónica, así como su superación postgraduada y al avance de la ciencia y la técnica en correspondencia con las necesidades y expectativas de la Sociedad Pinareña y nuestro Estado Revolucionario.**

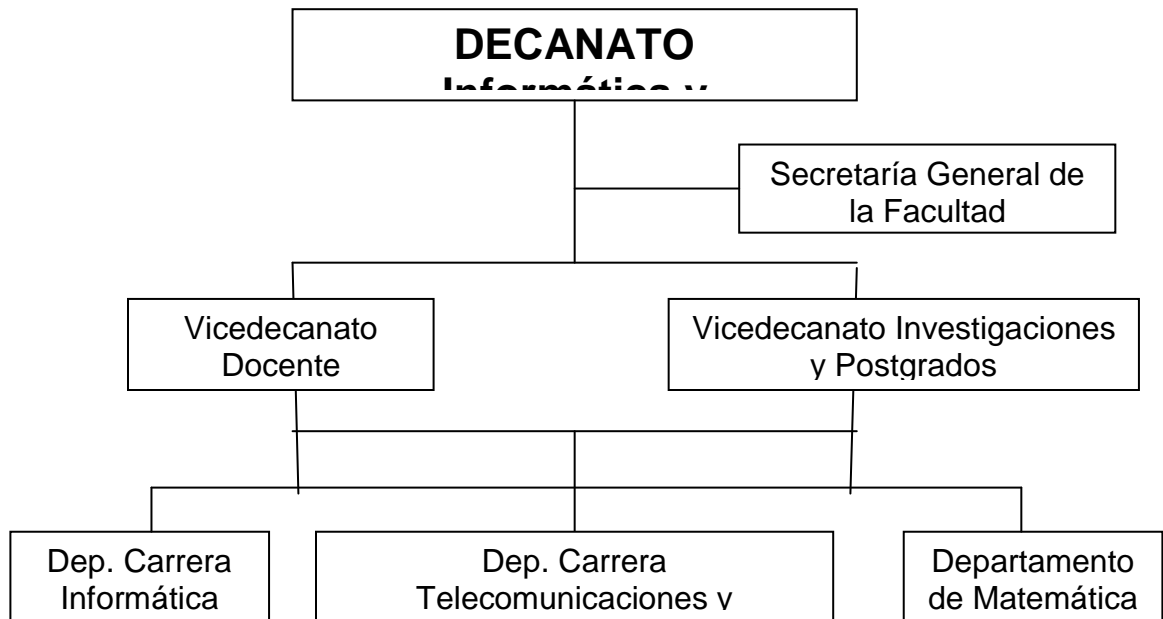
- **OBJETIVO CIENTÍFICO-TÉCNICO GENERAL**

Obtener y transferir resultados de alto impacto y reconocimiento territorial y nacional, a partir de una elevada integración tanto interna como con el ISPJAE, la UH, CIMEX, el MIC en general, priorizando la consolidación del trabajo por proyectos, con especial énfasis en la identificación de clientes de nuestros resultados, elevando la profundidad de la gestión económica de nuestros productos científicos y aplicando una política sistemática para la protección intelectual de éstos.

### 1.2.2. UBICACIÓN DENTRO DE LA ESTRUCTURA ORGANIZACIONAL TERRITORIAL.

**La Facultad de Informática y Telecomunicaciones de la Universidad Pinar del Río se localiza en el Municipio de Pinar, Provincia de Pinar del Río, País Cuba, ubicada a 145 Km. al oeste de la capital del país, Ciudad Habana, a la cual se enlaza por la**

**autopista nacional como principal vía de  
comunicación a estructura organizativa de la F.I.T. es  
la siguiente (Fig.: I.1).**



***Figura I.1. Estructura orgánica de la F.I.T.***

**La Facultad de Informática y Telecomunicaciones  
cuenta con cuatro laboratorios de computación,  
cerca de un centenar de computadoras personales,**

**también posee laboratorios de Electrónica  
Analógica, Antenas, Mediciones y Electrónica Digital.**

**La estructura organizacional de la Facultad de  
Informática y Telecomunicaciones está dada de la  
siguiente manera:**

- **Decana:**

Dra. Magdalena P. Mazón Hernández, malena@info.upr.edu.cu

- **Vicedecano de Investigaciones y Postgrados:**

Dr. Osvaldo Fosado Téllez, fosado@mat.upr.edu.cu

- **Vicedecano Docente:**

MSc. Alberto Serrano Gómez, serrano@info.upr.edu.cu

- **Jefe del Departamento Carrera de Telecomunicaciones y Electrónica:**

Dr. Wilfredo Falcón Urquiaga, falcon@tele.upr.edu.cu

- **Jefe del Departamento Carrera de Informática:**

MSc. Caridad Salazar Alea csalazar@info.upr.edu.cu

- **Jefe del Departamento de Matemática:**

MSc. Pablo Raúl Chávez H. [chavez@mat.upr.edu.cu](mailto:chavez@mat.upr.edu.cu)

- **Secretaria general de la Facultad:**

Lic. Elyrram Triana Carmona, [ely@tele.upr.edu.cu](mailto:ely@tele.upr.edu.cu)

## ESTRUCTURA DE LOS DEPARTAMENTOS CON LOS QUE CUENTA LA FACULTAD:

### **Estructura del departamento de Telecomunicaciones y electrónica**

- Profesores: 18 (2 en el extranjero y 1 en Programa de Imagenología)
- Adiestrados: 5 (2 Programa de Imagenología)
- Reservas Científicas: 0
- Doctores: 3 (16.7%)
- Master: 6 (33.3%)



## **Estructura del departamento de Informática**

- Profesores: 12 (3 Extranjero y 1 UCI)
- Adiestrados: 2 (1 Programa de Imagenología)
- Reservas Científicas: 0
- Doctores: 2 (16.7%)
- Master: 6 (50%)

## **Estructura del departamento de Matemática**

- Profesores: 20 (4 Extranjero y 4 UCI)
- Adiestrados: 0
- Reservas Científicas: 0
- Doctores: 4 (20.0%)
- Master: 11 (55.0%)

### **1.2.3. FUNCIONES PRINCIPALES QUE REALIZA LA FIT.**

La Facultad de Informática y Telecomunicaciones asume el encargo social de formar profesionales integrales capaces de impulsar el desarrollo del territorio con uso eficiente de las tecnologías de la información y las comunicaciones en medio de las condiciones particulares de la economía cubana.

**Para la formación de los futuros Ingenieros en Informática y en Telecomunicaciones y Electrónica se cuenta con una matrícula como promedio anual de 300 estudiantes en el Curso Regular Diurno y 150 en el Curso regular para Trabajadores en ambas carreras.**

1.2.4. ESTRATEGIA MAESTRA PRINCIPAL DE  
FACULTAD DE INFORMÁTICA Y  
TELECOMUNICACIONES

## **ENFOQUE INTEGRAL PARA LA LABOR EDUCATIVA Y POLÍTICO- IDEOLÓGICA**

*Trabajar en la formación político-ideológica de estudiantes,  
trabajadores y comunidad del entorno, con un enfoque integral,  
sistémico y abarcador utilizando como vía esencial las Tecnologías de  
la Información y las Comunicaciones para contribuir al proyecto  
socialista cubano a partir de la experiencia acumulada de los factores  
que conforman la F.I.T.*

## ESTRATEGIAS ESPECÍFICAS.

1-Perfeccionar las estrategias de los departamentos y carreras en función de los programas priorizados de la Revolución, para garantizar el logro de los objetivos generales educativos de cada programa.

2-Continuar profundizando en la incorporación de la Facultad como ente político esencial del territorio a la Batalla de Ideas que libra el pueblo cubano a través de productos multimedia sobre:

- La batalla por la libertad de los Héroes Cubanos prisioneros en Estados Unidos por luchar contra el terrorismo
- La batalla por todos los objetivos del Juramento de Baraguá.
- La batalla contra las consecuencias de la crisis económica que azota a la humanidad.
- La batalla por la educación y la cultura general integral del pueblo de Cuba.
- La batalla por la paz.
- La lucha contra la droga.

3-Desarrollar proyectos educativos en las brigadas caracterizados por el protagonismo estudiantil a partir del diagnóstico realizado y los fundamentos del modelo del profesional con énfasis en la dimensión curricular, soportado en el uso de las NTIC.

4-Desarrollar los proyectos sociales intra y extrauniversitarios como principal forma de trabajo en la dimensión extensionista de los proyectos educativos, a través de actividades asociadas a la instalación de redes e informatización de los procesos universitarios y la sociedad pinareña en general.

5-Continuar consolidando el movimiento político para elevar la calidad en la formación de profesionales revolucionarios con énfasis en la dedicación al estudio, la elevación de los indicadores cualitativos de la promoción, las estrategias curriculares, la investigación extracurricular y el impacto de la actividad investigativo–laboral y político-social de los estudiantes.

6-Dirigir el trabajo metodológico a la integración de las dimensiones de la vida universitaria y sus procesos, priorizando la educación a través de la instrucción, el perfeccionamiento de los planes de estudio con una concepción inter y multidisciplinaria. Garantizar la preparación metodológica de los profesores adjuntos, tutores y consultantes de los programas de la universalización, para desarrollar el proceso de educación mediante la instrucción en todos los tipos de cursos.

7-Desarrollar la dirección por objetivos en la Facultad a partir de los valores compartidos de la UPR y la FIT, incluyendo a todos los profesores, estudiantes y trabajadores de nuestro radio de acción.

8-Ejecutar las acciones dirigidas a la educación, prevención, detección y combate frontal contra el fraude, las ilegalidades, el delito, la corrupción y el uso indebido de alcohol y drogas en la vida de la universidad. Facilitar la capacitación de la comunidad universitaria en los conocimientos fundamentales acerca del trabajo de promoción de hábitos saludables de vida como método fundamental para esta lucha.

9-Consolidar la política científica de la Facultad mediante la obtención y transferencia de resultados de alto impacto y con reconocimiento territorial y nacional, a partir de una elevada integración tanto interna como con otros CES y OACE vinculados al perfil de la FIT, priorizando la consolidación del trabajo por proyectos, con especial énfasis en la identificación de los clientes de nuestros resultados, elevando la profundidad de la gestión económica de nuestros productos científicos y aplicando una política sistemática para la protección intelectual de éstos.

10-Desarrollar con resultados superiores los proyectos aprobados, sistematizando e introduciendo los resultados de las investigaciones relacionados con el uso de las TIC, la Electrónica, la Modelación y la Pedagogía, en función del impacto social.

11-Insertar la formación de postgrado en el Enfoque Integral de la Labor Educativa y Político-Ideológica, poniendo especial énfasis en las acciones vinculadas a la Universalización y las referidas a la Educación a Distancia.

12-Contribuir a fomentar un clima político-moral favorable en la Facultad mejorando de modo continuo la calidad en las condiciones de estudio, trabajo y vida de los estudiantes y trabajadores, así como un ambiente universitario que estimule el sentido de pertenencia y unidad de acción.

**13-Ejecutar una estrategia en Residencia Estudiantil que se integre al Enfoque Integral y que contribuya a mejorar los resultados obtenidos hasta la fecha, incrementando la atención por parte de la dirección de la Facultad y los colectivos de año, elevando la proporción de cuartos declarados universitarios.**

14-Aplicar el enfoque integral para la labor educativa y político-ideológica a la formación de profesionales en los CRPT atendiendo sus individualidades.

15-Mantener el liderazgo en la introducción de la Estrategia Maestra Informatización y continuar consolidando la introducción de las estrategias curriculares, con énfasis en la de Idioma Inglés, Historia de Cuba, Lengua Materna, y la Dimensión Ambiental en todos los procesos formativos universitarios.

16-Incorporar el enfoque integral para la labor educativa y político-ideológica al proceso de extensión universitaria, apoyados en las posibilidades que nos brindan las NTIC.

17-Incrementar el diálogo constante como método esencial del trabajo político-ideológico con los estudiantes y la comunidad universitaria en general. Continuar con una sistemática labor en la Facultad, orientada al desarrollo de valores éticos y revolucionarios en toda la comunidad universitaria en la que predomine el ejemplo personal de todos los cuadros de dirección y del claustro de profesores. .

18-Continuar priorizando la labor del profesor como educador y activista de la política del PCC.

19-Elevar constantemente la formación humanista de los estudiantes , trabajadores y profesores con énfasis en las competencias comunicativas en la lengua materna,



Historia de Cuba, historia de la profesión, cultura ambiental, aspectos legales de la profesión, ética y estética entre otros según lo planificado en los proyectos educativos y los planes metodológicos de carrera, año y disciplina.

20-Contribuir a la capacitación del claustro en aspectos que favorezcan la labor educativa, en particular la didáctica para la formación de valores, la psicología educativa, y el empleo de las NTIC como vehículo de la Estrategia Maestra Principal.

#### 1.2.5. NIVELES DE AUTOMATIZACIÓN.

**La información relevante que actualmente tiene la Facultad es como ya hemos mencionado almacenada manualmente, los datos se encuentran dispersos en la secretaría de la facultad, la secretaría del decanato y en los vicedecanatos, recogidos fundamentales en los Informes semestrales de evaluación de Áreas de Resultados Claves, Informes**

**Anuales de la Actividad de Investigación, Informes periódicos de las Secciones Sindicales , Balances del Forum de Ciencia y Técnica así como Tablas y reportes finales de Promoción por semestres.**

**Por lo que surge la necesidad de crear un Sistema de Almacenamiento Información Relevante de la Facultad de Informática y Telecomunicaciones, lo que permitirá optimizar los recursos y obtener un flujo de información de una forma ágil, dinámica, íntegra, confiable y en tiempo oportuno según los requerimientos de la Facultad.**

**1.2.6. VOLUMEN DE INFORMACIÓN DE ENTRADA Y SALIDA.**

**Describiremos los datos de entrada y salida de la información relevante tanto de la F.I.T. como de los departamentos, profesores y estudiantes que la conforman.**

➤ **Información de Entrada:**

- **Profesores.**

- **Datos personales.**
- **Cargos.**
- **Carrera.**
- **Departamento.**
- **Fotografía.**
- **Departamentos**

- **Estudiantes**

- **Datos personales.**
- **Año académico.**
- **Carrera.**
- **Deportes.**
- **Premios**

- **Fotografía.**

- **Publicaciones.**
- **Investigaciones y Proyectos**
- **Premios**
- **Registro de software**

➤ **Información de salida**

- **Reporte de Estudiantes por carrera.**
- **Reporte de Profesores por departamento.**
- **Reporte de Personas (Estudiantes/Profesores)**
- **Reporte de Publicaciones.**
- **Reportes de Investigaciones y Proyectos**
- **Reporte de Premios**
- **Reporte de Registro de software**
- **Reporte de Visitas a departamentos**

1.2.7. **PRINCIPALES DEFICIENCIAS Y PROBLEMAS.**

**No se cuenta en la actualidad con un sistema automatizado para el tratamiento de la información de estudiantes en la secretaría de la facultad,**

**tampoco a nivel de organización se posee un formato único para determinar qué información es realmente relevante y necesario preservar de manera homogénea para caracterizar de forma general la evolución de la actividad desde el punto de vista cualitativo; solamente se evalúa el desempeño por el grado de consecución de los criterios de medida vinculados a cada una de las área de resultado clave.**

**No existe de manera organizada un registro de las personas que de manera individual son acreedoras de estímulos y reconocimientos por actividades académicas, científicas, culturales o deportivas que tributan al prestigio de la Facultad y la Universidad.**

## 1.2.8. DEFINICIÓN DEL OBJETIVO DEL SISTEMA

### 1.2.8.1.OBJETIVO GENERAL.

**Automatizar los procesos de almacenamiento de Información Relevante de la Facultad de Ingeniería y Telecomunicaciones de la Universidad Pinar del Río “Hermanos Saíz Montes de Oca”.**

### 1.2.8.2.OBJETIVOS ESPECÍFICOS.

- **Identificar qué información resulta RELEVANTE para la Dirección de la FIT.**
- **Realizar el estudio preliminar del almacenamiento de la información relevante de la F.I.T.**



- **Diseñar e implementar una Base de Datos que cumpla con los requerimientos de almacenamiento de los datos de la F.I.T.**
- **Diseñar e implementar un Software que gestione la Base de Datos, en un ambiente multiusuario y que sea capaz de brindar la seguridad necesaria a la información.**
- **Utilizar las herramientas necesarias y adecuadas para la elaboración del Software de acuerdo a las más actualizadas tendencias que a nivel mundial se reconocen.**

#### 1.2.9. REQUERIMIENTOS TÉCNICOS.

**Para Automatizar los procesos de almacenamiento de información relevante de la F.I.T., se requieren de los siguientes recursos:**

**1.2.9.1. RECURSOS HUMANOS:**

**Se emplearán tres personas para el análisis, diseño y desarrollo del sistema de Almacenamiento de información relevante de la F.I.T**

**Tutor de Tesis**

**Dra. Magdalena**

**Mazón Hernández**

**Grupo de Investigadores**

**Egdo. César**

**Fernando Álvarez Zurita**

**Egdo. Luis Eduardo Escobar**

**Banda**

**1.2.9.2. RECURSOS TECNOLÓGICOS:**

**- HARDWARE:**

**Una computadora PIII con un disco duro de 20 GB,  
un Procesador Intel 851 MHz., 256 Memoria en RAM.**

**- SOFTWARE:**

**Sistema Operativo Windows.**

**Microsoft Access.**

**Borland Delphi 6.**

## 1.2.10. COMPARACIÓN DEL SISTEMA NUEVO CON OTROS SIMILARES.

La importancia y actualidad del tema está dada precisamente por la necesidad que tiene la Facultad de contar con una herramienta de apoyo, que permitirá utilizar los recursos de manera óptima y a la vez obtener resultados confiables y en un tiempo oportuno.

Actualmente no existen referencias de un sistema automatizado para la recolección de datos históricos de otras facultades. Lo cual hace que el sistema a implementarse en la facultad sea nuevo en este ámbito. Con la implementación del sistema se logrará una estructura de los datos que evite redundancia, deterioro y pérdida de la información.

La base de datos permitirá:

- Disminución del tiempo de espera.
- Aumento en la rapidez de la información.
- Generar información de calidad y confiable, de fácil transmisión en un tiempo óptimo.
- Mejora el manejo, almacenamiento y utilización de la información.

### 1.2.11. ANÁLISIS DE LA FACTIBILIDAD.

**Aquí analizaremos la factibilidad para el desarrollo del software a implementarse, así como evaluar el costo total del proyecto, tomando en cuenta los recursos económicos, técnicos y el factor tiempo para establecer los beneficios que prestará esta aplicación.**

#### 1.2.11.1. ECONÓMICO.

**La comparación del costo y esfuerzo del desarrollo del software son inmedibles. Hay que tomar en cuenta muchos aspectos que la pueden afectar. Por**

**esta razón las técnicas de comparación incluyen riesgos.**

Se utilizó para este análisis la metodología ADESA y dentro de ella el método COCOMO (Constructive Cost Model) donde se identificó el salario devengado considerando los costos de salario por según lo devengado por cualquier Recién Graduado de la Educación Superior.

Costos Directos:  $CD = CFT + CMT + CMAT + OG$

Donde:

**CFT** : Costo de la fuerza de trabajo.

**CMT** : Costo de los medios técnicos.

**CMAT** : Costo de los Materiales.

**OG** : Otros Gastos.

**Costos de la fuerza de trabajo:** Fue calculado en base al tiempo de trabajo invertido por el autor en la confección de este sistema, de acuerdo a la siguiente expresión:

$CFT = SB + SC + SS$

Donde:

**SB** = Salario Básico

**SC** = Salario Complementario

**SS** = Seguridad Social.

**SB** =  $A_i \times B_i \times C$

Donde:

**A<sub>i</sub>** = Horas promedio mensuales = 240

**C** = Cantidad de meses = 4

**B<sub>i</sub>** = Costo por hora del especialista.

$B_i$  = Salario mensual/ Horas promedio mensuales.

$B_i = 295/240h/m = 1.22\$/h$

$SB = 240h/m \times 1.22\$/h \times 4m$

**SB = \$1171.20/**

**SC = SB x 0.0909.**

$SC = \$1171.20 \times 0.0909$

**SC = \$106.46/**

**SS = 0.12 x (SC + SB)**

$SS = 0.12 \times (\$106.46 + \$1171.20)$

**SS = 153.32/**

$$\text{CFT} = 1171.20 + 106.46 + 153.32$$

$$\text{CFT} = \$1430.98//$$

**Costo de los medios técnicos.**

$$\text{CMT} = \text{Cdep} + \text{CE} + \text{CMTO}$$

Donde:

**Cdep** = Costo por depreciación

**CMTO** = Costo de mantenimiento de equipos.

**CE** = Costo por concepto de Energía.

$$\text{CE} = \text{HTM} \times \text{CEN} \times \text{CKW}$$

Donde:

**HTM** = Horas totales de tiempo de máquina

**CEN** = Consumo total de Energía.

**CKW** = Costo por Kwats/horas (\$0.09 hasta 100 Kws \$0.20 de 101 a 300 Kws y \$0.30 más de 300 Kws)



$$\mathbf{HTM} = (T_{dd} \times K_{dd} + T_{ip} \times K_{ip}) \times 152$$

Donde:

**T<sub>dd</sub>** = Tiempo promedio utilizado para el diseño y desarrollo (4 meses)

**K<sub>dd</sub>** = Coeficiente que indica el promedio de tiempo de diseño y desarrollo

que se utilizó en la máquina (.25)

**T<sub>ip</sub>** = Tiempo promedio utilizado para las pruebas implantación. (4 horas)

**K<sub>ip</sub>** = Coeficiente que indica el % de tiempo e implantación que se utilizó en la máquina (0.8)

$$HTM = (4 \times 0.25 + 4 \times 0.8) \times 152$$

$$HTM = (1 + 3.2) \times 152$$

$$\mathbf{HTM = 638.4 H}$$

$$\mathbf{CEN = 0.608 Kw/h// (ESTIMADO)}$$

$$\mathbf{KW} = \text{HTM} \times \text{CEN}$$

$$\mathbf{KW} = 638.4 \times 0.608$$

$$\mathbf{KW} = \mathbf{388.14}$$

$$\mathbf{CKW} = (100 \times 0.09) + (200 \times 0.20) + (88.14 \times 0.30)$$

$$\mathbf{CE} = \mathbf{\$75.44//}$$

*Costo Material Técnico. El costo de utilización de los medio técnicos:*

$$\mathbf{CMT} = \$12.00 + \$75.44 + \$15.00$$

$$\mathbf{CMT} = \mathbf{102.44//}$$

**Costo de Materiales.** El costo de materiales se calculó por la siguiente fórmula:

$$\mathbf{CMAT} = 0.05 \times \text{CMT}$$

Donde:

CMT = Costo de los Medios Técnicos

$$\mathbf{CMAT} = 0.05 \times 102.44$$

$$\mathbf{CMAT} = \mathbf{5.12//}$$

**OG: Se estima en \$1050//**

$$\mathbf{CD = CFT + CMT + CMAT + OG}$$

$$\mathbf{CD = 1430.97 + 102.44 + 5.12 + 1050}$$

$$\mathbf{CD = 2588.53//}$$

## **COSTO TOTAL:**

Para calcular el Costo Total del proyecto se utilizó la siguiente expresión:

$$\mathbf{CT = CD + 0.1 \times SB}$$

$$\mathbf{CT = 2588.53 + 0.1 \times 1171.20}$$

$$\mathbf{\underline{CT = 2705.65//}}$$

Aunque el Costo Total fue calculado en \$2705.65 los efectos periódicos identificados radican fundamentalmente en el incremento de la efectividad, eficiencia y rapidez del procesamiento y obtención de la información, sobre los que no existen bases de comparación. Ahorros

por concepto de salarios no son aplicables porque en realidad no se propone eliminar ninguna actividad de las entidades asociadas a la actividad de registro de la información relevante de la F.I.T con la inclusión de este sistema. En resumen el costo del proyecto está evaluado en \$2705.65.

#### 1.2.11.2. TÉCNICO.

- **Dispone de una computadora en el decanato de la Facultad de las siguientes características:**
  - **Computadora Pentium III**
  - **Procesador de 1.8 MHz**
  - **256 MB en RAM.**
  - **Disco Duro de 20 GB.**
  - **Tarjeta de Red de 100/10**
  - **Impresora.**

### **1.2.11.3. OPERATIVO.**

Se cuenta con el apoyo de la Dra. Magdalena Mazón, del Dr. Osvaldo Fosado Téllez y del MSc. Alberto Serrano Gómez, quienes nos han dado a conocer los procesos de recolección de datos relevantes de la Facultad de Informática y Telecomunicaciones. También se consultan los registros de la Secretaría de la Facultad, y los tres departamentos docentes.

# **CAPÍTULO II**

## **ANÁLISIS Y DISEÑO DE SAIRFIT**

En este capítulo se realizará la respectiva descripción del fundamento teórico de las herramientas utilizadas (Microsoft Access, Borland Delphi 6. Herramientas CASE E/R). Además se mostrarán los aspectos vinculados a la Ingeniería de Sistemas con el Diagrama de Contexto, Diagrama de Flujo de Datos con sus diferentes niveles de descomposición y la respectiva descripción de Flujo de Datos, Descripción Lógica, Modelo Entidad Relación y el diseño del Grafo Conversacional.

### ***2.1. METODOLOGÍA UTILIZADA***

La metodología ha sido concebida para aplicaciones a desarrollar en medios ambientes de programación tales como Microsoft Visual Basic, Microsoft Access y otros similares que se basan en técnicas estructuradas.

En estos medios de programación se posibilita la confección de un prototipo del sistema [Boar 84] <sup>1</sup> e inclusive evolucionar este prototipo en el software final, por esta razón MetVisual E se basa en la filosofía de trabajo que hace uso de prototipos [Connell 89] <sup>2</sup>. Con el uso del prototipo se facilita y mejora la comunicación con el usuario desde etapas tempranas del análisis, no sustituye la relación del análisis ni elimina la necesidad de documentar las aplicaciones pero logra un aumento de la calidad del producto final.

La MetVisual E utiliza el enfoque de desarrollo de prototipos, el cual consiste en “pensar un poco, construir un poco, probar un poco y entonces hacerlo de nuevo”. Ya que esa forma aprovecha la potencialidad para esos ambientes visuales y puede lograr un producto final que logre la satisfacción del usuario.

### **2.1.1. Ciclo de Vida**

- **Estudio Preliminar**

---

<sup>1</sup> Boar, B.H. “Application Prototyping : a requerments definition strategy for the 80s”

<sup>2</sup> Connell, J.L y Brice, L. “Structured Rapid Prototyping” Prentice Hall, 1989

- a.) Identificador de las necesidades del usuario
- b.) Valoración del uso de MetVisual E
- c.) Estudio de factibilidad
- d.) Elaboración del plan de desarrollo del sistema

- **Generación y especificación del prototipo**

- a.) Análisis preliminar de los requisitos.
- b.) Desarrollo de la especificación básica.
- c.) Desarrollo del prototipo inicial

- **Evolución del prototipo y especificaciones del software**

- a.) Prueba del prototipo.
- b.) Refinamiento del prototipo y de la especificación del software

- **Desarrollo**

- a.) Definición rigurosa de los componentes.
- b.) Completamiento de la programación.
- c.) Prueba de validación del software

- **Prueba de Implantación**

- a.) Prueba del sistema.



b.) Preparación de las condiciones

- **Implantación del sistema**

a.) Prueba del sistema.

b.) Preparación de condiciones.

c.) Implantación del sistema

La diferencia con el enfoque tradicional de desarrollo radica en que el ciclo de vida de un proyecto antes era estructurado en etapas o fases de estricto cumplimiento, donde se mantenía al usuario al margen, desde que firma las especificaciones hasta que se le entrega el sistema y se relega la etapa de prueba a una sola etapa del proyecto, en cambio el ciclo de vida de un proyecto en MetVisual E, al utilizar el enfoque de desarrollo por prototipos propone una estructura iterativa, con gran concurrencia entre los procesos que se desarrollan en las diferentes etapas o fases [Álvarez, Sofía 1999]<sup>3</sup> [1]

## **2.2. TABLA DE EVENTOS**

Aquí identificamos los documentos fuentes, los usuarios finales y los eventos a los que el sistema debe responder, así como los destinos, que pueden ser usuarios, ficheros o bases de datos [2].

---

<sup>3</sup> MetVisualE, ALVAREZ, Sofía, 1999

La siguiente tabla de eventos muestra los eventos identificados en SAIRFIT, con entrada de datos mediante las fuentes detalladas y salida de información hacia un destino establecido, es decir el Decanato de la F.I.T.

<b>Nro.</b>	<b>Evento</b>	<b>Entrada</b>	<b>Salida</b>	<b>Fuente</b>	<b>Destino</b>
1	Actualizar Estudiantes	Datos del Estudiante		Secretaría Docente	
2	Actualizar Profesores	Datos del Profesor		Departamento docente	
3	Registrar información relevante de Investigaciones y Proyectos	Datos Investigaciones y Proyectos		Departamento docente.	
4	Registrar Software	Datos Relevantes del Registro de Software		Departamento docente.	
5	Registrar Premios	Datos de los Premios		Vicedecanato de Investigaciones	
6	Registrar Publicaciones	Datos de Publicaciones		Departamento docente.	
7	Emitir reporte		Reporte de Datos		Decanato de la F.I.T

			relevantes de la FIT		
--	--	--	-------------------------	--	--

***Figura II.1. Tabla de Eventos de SAIRFIT***

### **2.3. DIAGRAMA DE CONTEXTO**

El diagrama de Contexto está presentado por un solo proceso, que identifica cual es la función principal del sistema, mostrando además, los flujos de información que lo relacionan con otros sistemas: las entidades externas. El diagrama de contexto tiene una gran importancia puesto que resume el requisito principal del sistema de recibir ciertas entradas, procesarlas de acuerdo con determinada función y generar ciertas salidas. A partir del diagrama de contexto se puede ir construyendo nuevos diagramas que vayan definiendo con mayor nivel de detalle lo flujos de datos y procesos de transformación que ocurren en el sistema, de forma que al final obtenemos una jerarquía de diagramas. [3] (Ver Anexo A)

### **2.4. DIAGRAMA FLUJO DE DATOS**

El diagrama de flujo de datos, es una representación gráfica que permite al analista definir entradas, procedimientos y salidas de la información en la organización bajo estudio, permitiendo así comprender los procedimientos existentes con la finalidad de optimizarlos, reflejándolos en el sistema propuesto.

**En resumen un DFD representa a un sistema o parte de este como una red de procesos funcionales conectados mediante “conductos” y “almacenamientos”. [4]. (Ver Anexo B)**

### **Simbología Utilizada**

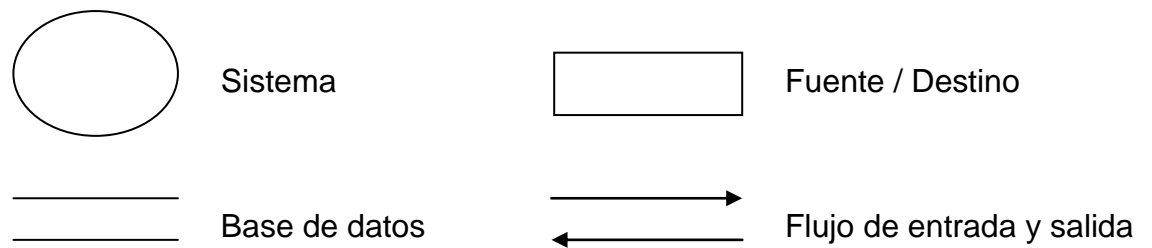


Figura II.2. Simbología utilizada para los diagramas de flujo de datos.

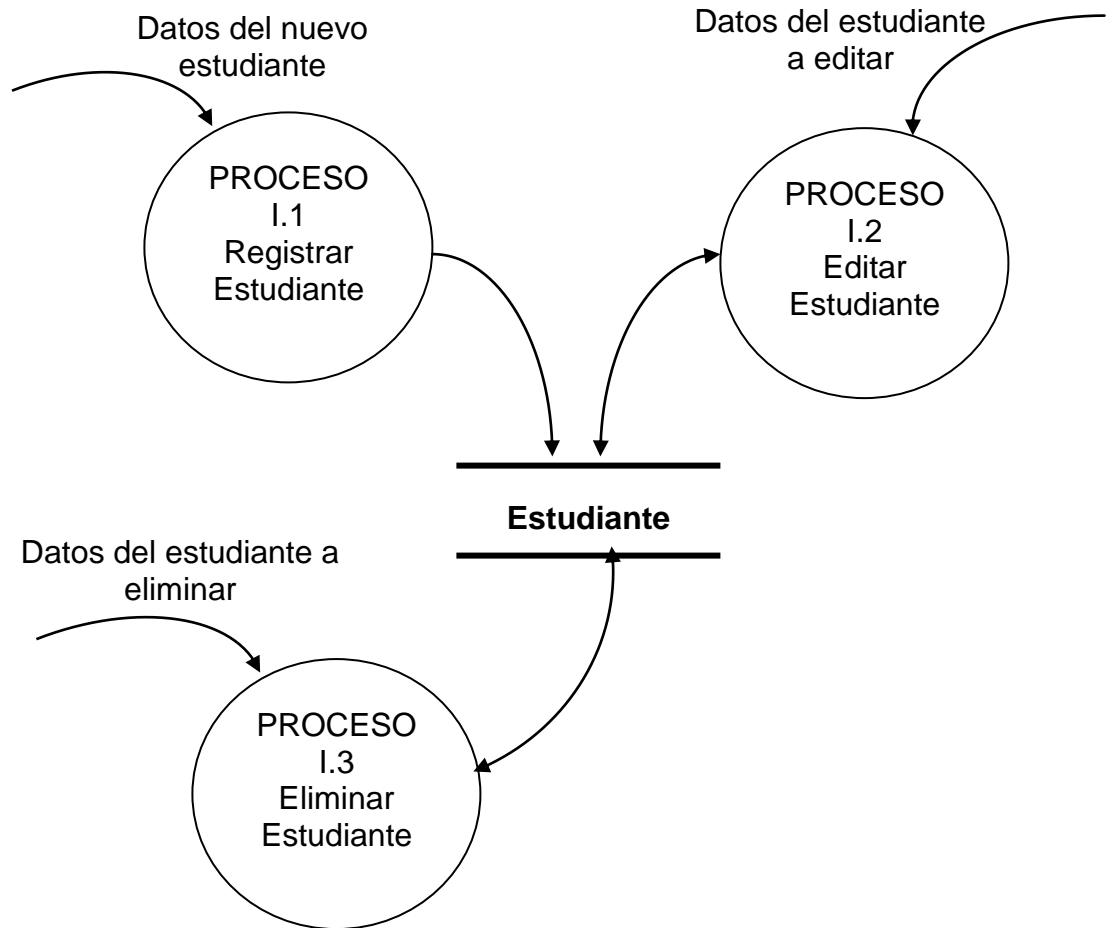
#### **2.5. DIAGRAMA DE DESCOMPOSICIÓN**

**Un diagrama de descomposición, también denominado gráfico de jerarquías, muestra la**

**estructura, o descomposición funcional en sentido  
descendente, de un sistema. [5]**

A continuación detallaremos la descomposición de los procesos funcionales del diagrama de flujo de datos, estudiante y profesor, de nuestro sistema:

## Actualizar Estudiante



*Figura II.3. Actualizar Estudiante*

*Actualizar Profesor*

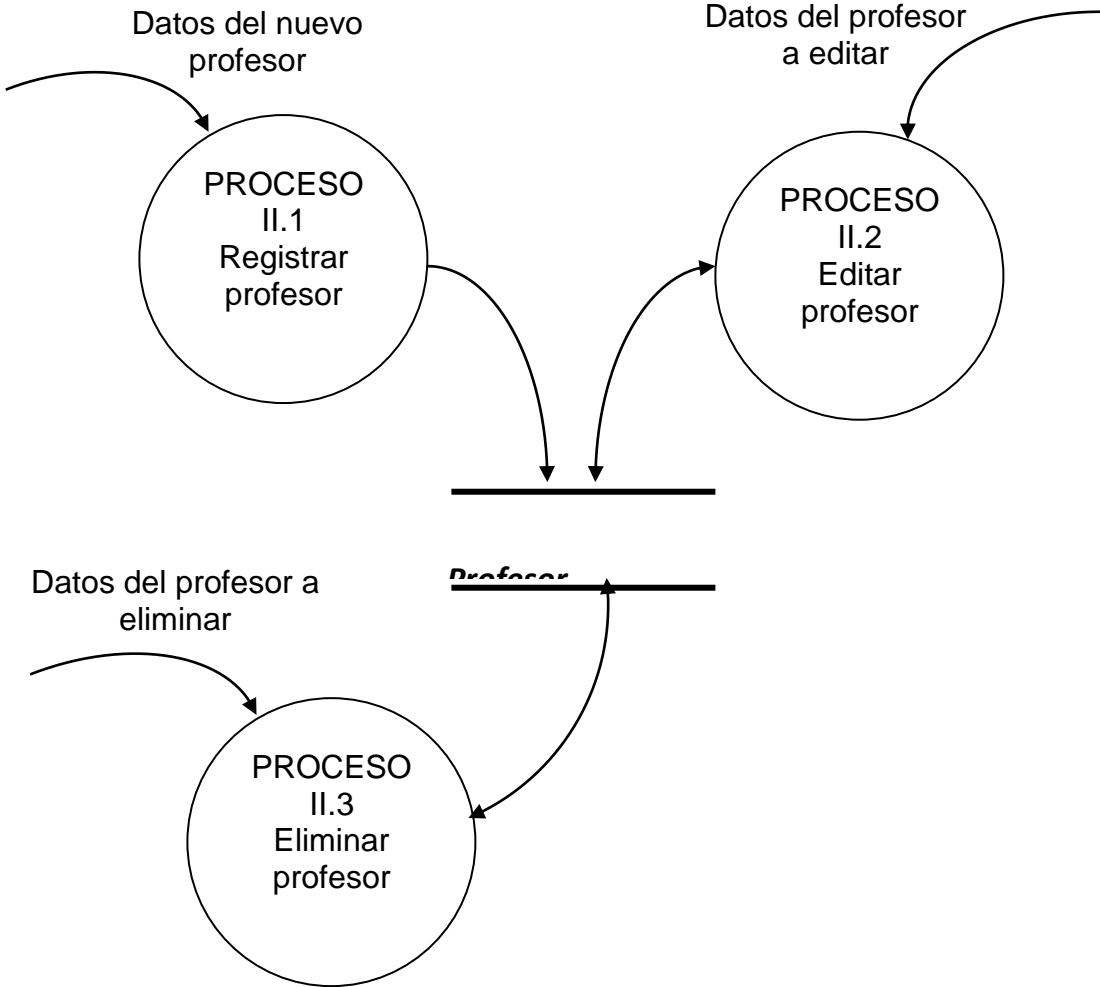




Figura II.4. Actualizar Profesor

**El resto de diagramas de descomposición se apreciarán en los siguientes Anexos C, D, E, F.**

## **2.6. DESCRIPCIÓN TEXTUAL DE LOS FLUJOS DE DATOS**

Los flujos de datos que entran y salen de los procesos están compuestos por:

- Secuencia de datos elementales
- Selección de datos elementales
- Agrupación repetitiva de datos elementales [6]

La siguiente tabla muestra la notación utilizada en la Descripción Textual de los Flujos de Datos que entran y salen del Sistema mostrados en el Diagrama de Contexto

<b>Construcción del dato</b>	<b>Notación</b>	<b>Significado</b>
	=	Esta compuesto de
Secuencia	+	Y
Selección	[ / ]	O bien – o
Repetición	{ }	Repeticiones de
	()	Opcional
	*	Delimitar Comentario

*Figura II.5. Simbología para la Descripción Textual de los flujos de datos.*

### **Proceso I. Actualizar Estudiante**

[ Datos del nuevo estudiante / Datos del estudiante a editar / Datos del estudiante a eliminar ]

**Donde:**

**Proceso I.1. :**

**Datos del nuevo estudiante** = id\_est + { NombreEst + ApellidoEst1 + ApellidoEst2 + DirecciónEst + TeléfonoEst + SexoEst + RazaEst + e\_mailEst + OrganizaciónPolEst + Víaingreso + Extranjero + ProcedenciaSocial + foto }

**Donde:**

OrganizaciónPolEst = U.J.C/P.C.C

SexoEst = Masculino/Femenino

RazaEst = Mestiza/Amarilla/Negra/Blanca/indígena

ProcedenciaSocial = Obrero/Campesino/

Víaingreso =Becado/Autofinanciado/Otro

Extranjero = SI/NO

**Proceso I.2. :**

**Datos del estudiante a editar** = id\_est / [ NombreEst / ApellidoEst1 / ApellidoEst2 / DirecciónEst / TeléfonoEst / SexoEst / RazaEst / e\_mailEst / OrganizaciónPolEst / Víaingreso / Extranjero / ProcedenciaSocial / foto ]

**Proceso I.3.:**

Datos del estudiante a eliminar = **id\_est**

**Proceso II. Actualizar Profesor**

[ Datos del nuevo profesor / Datos del profesor a editar / Datos del profesor a eliminar ]

**Donde:**

**Proceso II.1. :**

**Datos del nuevo profesor** = id\_prof + { NombreP + ApellidoP1 + ApellidoP2 + SexoP + RazaP + DireccionP + TelefonoP + e\_mail + CategoriaDc + CategoriaCient + OrganizacionP + foto + cargo + Departamento }

**Donde:**

SexoP = Masculino / Femenino

RazaP = Mestiza / Amarilla / Negra / Blanca / Indígena

CategoriaDc = Profesor / Licenciado / Ingeniero

CategoriaCient = Dr. Ciencias / Dr. Ciencias Técnicas / Dr. Ciencias Económicas / Dr. Ciencias Forestales / Dr. Ciencias Pedagógicas

OrganizacionP = U.J.C./P.C.C

Cargo = Profesor / Administrador / Chofer / Secretario / Técnicos / Otros

Departamento = Matemática / Telecomunicaciones / Informática

**Proceso II.2. :**

**Datos del profesor a editar** = id\_prof / [ NombreP / ApellidoP1 / ApellidoP2 / DireccionP / TelefonoP / SexoP / RazaP / e\_mail / CategoriaDc / CategoriaCient / OrganizacionP / foto / cargo ]

**Proceso II.3.:**

**Datos del profesor a eliminar** = id\_prof

**Proceso III. Registrar información relevante de Investigaciones y Proyectos**

**Registro de investigaciones** = id\_inv + [ Nombre\_Investigación +  
fecha\_inicio + fecha\_final ]

**Donde:**

Fecha\_inicio = Fecha\_final = [ aa/mm/dd ]

**Registro de Línea\_ investigación** = id\_linea + [ NombreLinea + Proyecto ]

**Registro de Proyecto** = id\_Proj + [ Nombre\_Proj + Jefe\_Proj +  
fecha\_inicial\_Proj + fecha\_final\_Proj ]

Donde

fecha\_inicial\_Proj = fecha\_final\_Proj = [ aa/mm/dd ]

**Registro de programa** = id\_prog + [ Nombre\_prog + institución + instancia  
]

*Proceso IV. Registrar Software*

**Registro de Software** = No\_Registro + [ Nombre\_software + fecha\_soft]

**Proceso V. Registrar Premios**

**Registro de Premios** = Id\_premio + [ instancia\_premio + nombre\_premio +  
institución\_premio ]

**Proceso VI. Registrar Publicaciones**

**Registro de Publicaciones** = ISBN + [ Revista + Título + DirecciónWeb +  
fecha\_Pub]

**Donde:**

fecha\_Pub = [aa/mm/dd]

*Proceso VII. Emitir Reportes al decanato*

**Reporte de Estudiantes** = NombreEst + ApellidoEst1 + ApellidoEst2 + SexEst

**Reporte de Profesores** = NombreP + ApellidoP1 + ApellidoP2 + Provincia + SexoP + CategoriaDc + CategoriaCient + Departamento

## 2.7. GRAFO CONVERSACIONAL (GC)

MetVisual para mostrar una visión más clara del sistema propone el grafo conversacional, en donde se especifica la secuencia de eventos que son deseables desde el punto de vista del usuario. Asumen que la secuencia del sistema estará siempre bajo control del usuario. El GC brinda un modelo visual de la secuencia con que cada función del sistema se mostrará. Por lo tanto no se necesita representar acoplamiento ni parámetros de control. (Ver Anexo G)

## 2.8. DIAGRAMA ENTIDAD RELACIÓN

El modelo Entidad-Relación es en esencia una herramienta para representar el mundo real por medio de simbologías y expresiones determinadas. [7].

El modelo E-R es el modelo conceptual más utilizado para el diseño conceptual de bases de datos. El modelo entidad-relación está formado por un conjunto de

conceptos que permiten describir la realidad mediante un conjunto de representaciones gráficas y lingüísticas. Los elementos esenciales del modelo son las entidades, los atributos y las relaciones entre las entidades. [8] [9].

### **2.8.1. Entidad.**

Cualquier tipo de objeto o concepto sobre el que se recoge información: cosa, persona, concepto abstracto o suceso. Las entidades se representan gráficamente mediante rectángulos y su nombre aparece en el interior. Un nombre de entidad sólo puede aparecer una vez en el esquema conceptual.

Hay dos tipos de entidades: fuertes y débiles. Una *entidad débil* es una entidad cuya existencia depende de la existencia de otra entidad. Una *entidad fuerte* es una entidad que no es débil.



### **2.8.2. Atributo.**

Es una característica de interés o un hecho sobre una entidad o sobre una relación.

Los atributos representan las propiedades básicas de las entidades y de las relaciones. Toda la información extensiva es portada por los atributos.

Gráficamente, se representan mediante bolitas que cuelgan de las entidades o relaciones a las que pertenecen. [7]

### **2.8.3. Relación.**

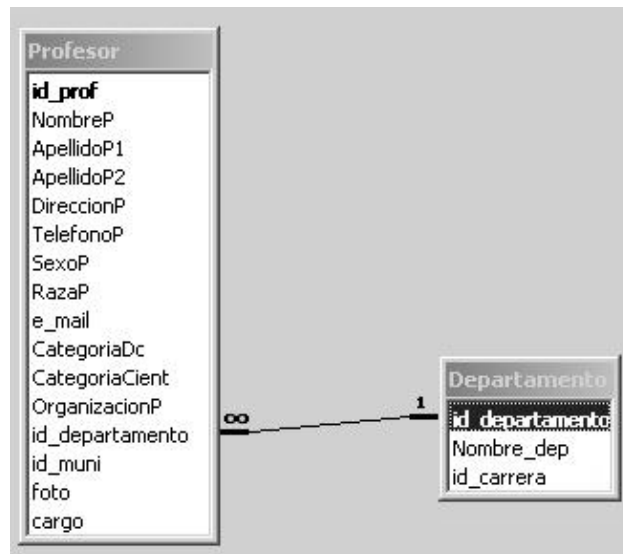
Es una correspondencia o asociación entre dos o más entidades. Cada relación tiene un nombre que describe su función. Las relaciones se representan gráficamente mediante rombos y su nombre aparece en el interior. Las entidades que están involucradas en una determinada relación se denominan entidades participantes. [8]

Entre dos tablas de cualquier base de datos relacional puede haber tres tipos de relaciones:

**2.8.3.1.Relaciones 1-1.-** Cuando las entidades que intervienen en la relación se asocian una a una.

**2.8.3.2.Relaciones 1-n.-** Una ocurrencia de una entidad está asociada con muchas (n) de otra.

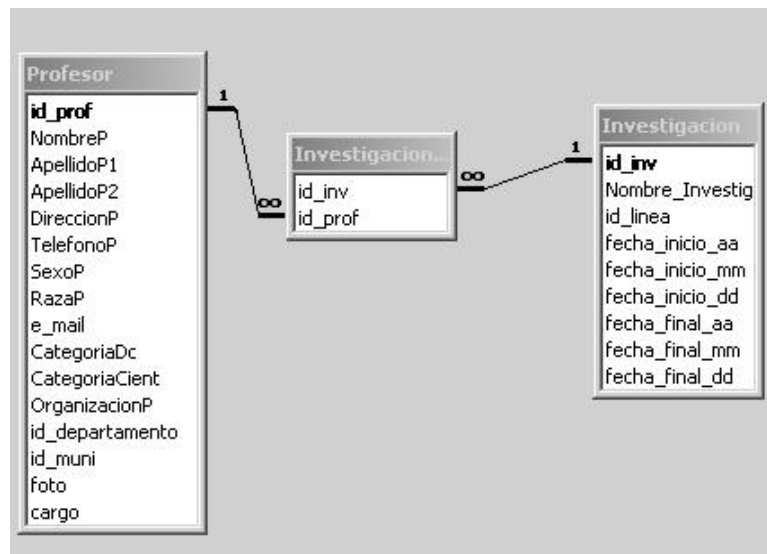
En nuestra base de datos se evidencia esta relación entre la tabla Departamento y la tabla Profesor, donde un Departamento puede tener a su cargo muchos profesores; y un profesor tiene relación con un único departamento.



*Figura II.6. Relaciones 1-n / SAIRFIT*

**2.8.3.3.Relaciones n-n.-** Cada ocurrencia, en cualquiera de las dos entidades de la relación, puede estar asociada con muchas (n) de la otra y viceversa.

Una relación n-n conlleva a la creación de una tabla adicional, la cual por lo general consta con las claves principales de cada tabla, en otros casos además con atributos propios. En nuestro ejemplo la relación de n-n de la tabla Investigación con la tabla Profesores generó una tabla llamada Investigación\_Profesor.



*Figura II.7 Relaciones n-n / SAIRFIT*

## **2.9. TENDENCIAS Y TECNOLOGÍA ACTUALES A CONSIDERAR**

En el desarrollo de S.A.I.R.F.I.T. se utilizaron las siguientes herramientas tecnológicas que detallamos a continuación:

### **2.9.1. HERRAMIENTA CASE ER/STUDIO**

Cuando se trabaja con bases de datos es necesario tener una buena herramienta que nos ayude a modelar nuestro sistema, y tenga la facilidad de pasar del modelo lógico al modelo físico, por esa razón hemos considerado la herramienta CASE ER/Studio.

- 

[10]

### **2.9.1.1. Modelo Lógico**

El diagrama lógico de datos da un panorama del sistema, pero a diferencia del físico es independiente de la implantación, que se centra en el flujo de datos entre los

procesos, sin considerar los dispositivos específicos y la localización de los almacenes de datos o personas en el sistema. Sin indicarse las características físicas (Ver Anexo H)

### **2.9.1.2. Modelo Físico**

Esta etapa depende del SGBD comercial que se utilizará para implementar la base de datos. El diagrama físico de datos da un panorama del sistema en uso, dependiente de la implantación, mostrando cuales tareas se hacen y como son hechas. Incluyen nombres de personas, nombres o números de formato y documento, nombres de departamentos, archivos maestro y de transacciones, equipo y dispositivos utilizados, ubicaciones, nombres de procedimientos. (Ver Anexo I.)

Los Sistemas Gestores de Bases de Datos son un tipo de software muy específico, dedicado a servir de interfaz entre las bases de datos y las aplicaciones que la utilizan.

**Hoy en día, son muchas las aplicaciones que requieren acceder a datos. Bien sea un sencillo programa doméstico, bien una suite para la gestión empresarial. Estos datos se deben almacenar en algún soporte permanente, y las aplicaciones deben disponer de un medio para acceder a ellos.**

**Normalmente, la forma en que un programa accede a un fichero es a través del Sistema operativo. Este provee de funciones como abrir archivo, leer información del archivo, guardar información, etc.**

**No obstante, este procedimiento de acceso a ficheros es altamente ineficaz cuando se trata con un volumen elevado de información.**

Los Sistemas Gestores de Bases de Datos proporcionan un interfaz entre aplicaciones y sistema operativo, consiguiendo, entre otras cosas, que el acceso a los

datos se realice de una forma más eficiente, más fácil de implementar, y, sobre todo, más segura. [11]

que deben cumplir los SGBD tenemos:

- **Abstracción de la información.** Los usuarios de los SGBD ahorran a los usuarios detalles acerca del almacenamiento físico de los datos. Da lo mismo si una base de datos ocupa uno o cientos de archivos, este hecho se hace transparente al usuario. Así, se definen varios niveles de abstracción.
- **Independencia.** La independencia de los datos consiste en la capacidad de modificar el esquema (físico o lógico) de una base de datos sin tener que realizar cambios en las aplicaciones que se sirven de ella.
- **Redundancia mínima.** Un buen diseño de una base de datos logrará evitar la aparición de información repetida o redundante. De entrada, lo ideal es lograr una redundancia nula; no obstante, en algunos casos la complejidad de los cálculos hace necesaria la aparición de redundancias. Eliminando o controlando las redundancias de datos se reduce en gran medida el riesgo de que haya inconsistencias.



- **Consistencia.** En aquellos casos en los que no se ha logrado esta redundancia nula, será necesario vigilar que aquella información que aparece repetida se actualice de forma coherente, es decir, que todos los datos repetidos se actualicen de forma simultánea.

Si un dato está almacenado una sola vez, cualquier actualización se debe realizar sólo una vez, y está disponible para todos los usuarios inmediatamente. Si un dato está duplicado y el sistema conoce esta redundancia, el propio sistema puede encargarse de garantizar que todas las copias se mantienen consistentes.

- **Seguridad.** La información almacenada en una base de datos puede llegar a tener un gran valor. Los SGBD deben garantizar que esta información se encuentra asegurada frente a usuarios malintencionados, que intenten leer información privilegiada; frente a ataques que deseen manipular o destruir la información; o simplemente ante las torpezas de algún usuario autorizado pero despistado. Normalmente, los SGBD disponen de un complejo sistema de permisos a usuarios y grupos de usuarios, que permiten otorgar diversas categorías de permisos.

- **Integridad.** Se trata de adoptar las medidas necesarias para garantizar la validez de los datos almacenados. Es decir, se trata de proteger los datos ante fallos de hardware, datos introducidos por usuarios descuidados, o cualquier otra circunstancia capaz de corromper la información almacenada. prevenir / detectar / impedir la modificación inadecuada de información. En un entorno comercial, la integridad de los datos es especialmente relevante, puesto que el éxito de una organización depende de lo correctas que son las operaciones que se llevan a cabo y la coherencia en los datos.
  - Integridad semántica: Respeto en todo momento de las reglas de integridad definida en la base de datos.
  - Integridad Operacional: Garantizar la consistencia de la base de datos con respecto al uso concurrente de la misma.
- **Respaldo y recuperación.** Los SGBD deben proporcionar una forma eficiente de realizar copias de seguridad de la información almacenada en ellos, y de restaurar a partir de estas copias los datos que se hayan podido perder.
- **Control de la concurrencia.** En la mayoría de entornos (excepto quizás el doméstico), lo más habitual es que sean muchas las personas que acceden a una base de datos, bien para recuperar información, bien para almacenarla. Y es

también frecuente que dichos accesos se realicen de forma simultánea. Así pues, un SGBD debe controlar este acceso concurrente a la información, que podría derivar en inconsistencias.

- **Tiempo de respuesta.** Lógicamente, es deseable minimizar el tiempo que el SGBD tarda en darnos la información solicitada y en almacenar los cambios realizados. [11]

### **2.9.3. MICROSOFT ACCESS 2000**

Microsoft Access es un gestor de bases de datos relacional (SGBDR) que utiliza distintos objetos para almacenar, organizar, seleccionar y recuperar la información de uso personal o de pequeñas organizaciones. Su funcionamiento se basa en un motor llamado Microsoft Jet, posibilidad adicional es la de crear ficheros con bases de datos que pueden ser consultados por otros programas. [12]

#### **2.9.3.1. Características de Microsoft Access**

A continuación mencionaremos algunas características de Microsoft Access:

- Access tiene la capacidad de organizar, buscar y presentar la información resultante del manejo de sus bases de datos.

- Access es gráfico, por lo que aprovecha al máximo la potencia gráfica de Windows, ofreciendo métodos usuales de acceso a los datos y proporcionando métodos simples y directos de trabajar con la información.
- Access facilita la administración de datos, ya que sus posibilidades de consulta y conexión le ayudan a encontrar rápidamente la información deseada, cualquiera que sea su formato o lugar de almacenamiento.
- Access permite lograr un considerable aumento en la productividad mediante el uso de los asistentes y las macros. Estos permiten automatizar fácilmente muchas tareas sin necesidad de programar. [13]

#### **2.9.4. MOTOR DE BASE DE DATOS JET 4**

Access no es quien crea y gestiona directamente los archivos mdb, sino el motor Jet 4 de Microsoft. El motor Jet está formado por una serie de archivos .dll de distribución gratuita que se instalan en el directorio de sistema de Windows. De hecho, es posible crear en Windows un archivo mdb, o incluso llenarlo de tablas y datos aun sin tener Access. Para ello basta con emplear ODBC o ADO en programas hechos en Visual Basic o en otros lenguajes similares (Delphi) [14]

#### **2.9.4.1.DLL (Bibliotecas de Enlace Dinámico / Dynamic Linking Library)**

DLL es el término con el que se refiere a los archivos con código ejecutable que se cargan bajo demanda del programa por parte del sistema operativo. Esta denominación se refiere a los sistemas operativos Windows siendo la extensión con la que se identifican los ficheros, aunque el concepto existe en prácticamente todos los sistemas operativos modernos.

##### **Ventajas:**

Las DLLs *pueden* verse como la evolución de las bibliotecas estáticas y de forma análoga contienen funcionalidad y/o recursos que utilizan otras aplicaciones. Sin embargo, su uso proporciona algunas ventajas:

- **Reducen el tamaño de los archivos ejecutables:** Gran parte del código puede estar almacenado en bibliotecas y no en el propio ejecutable lo que redundaría en una mejor modularización
- **Pueden estar compartidas entre varias aplicaciones:** Si el código es suficientemente genérico, puede resultar de utilidad para múltiples aplicaciones

(por ejemplo, la MFC es una biblioteca dinámica con clases genéricas que recubren la API gráfica de Windows y que usan gran parte de las aplicaciones).

- **Facilitan la gestión y aprovechamiento de la memoria del sistema:** La carga dinámica permite al sistema operativo aplicar algoritmos que mejoren el rendimiento del sistema cuando se carguen estas bibliotecas. Además, al estar compartidas, basta con mantener una copia en memoria para todos los programas que la utilicen.
- **Brindan mayor flexibilidad frente a cambios:** Es posible mejorar el rendimiento o solucionar pequeños errores distribuyendo únicamente una nueva versión de la biblioteca dinámica. Nuevamente, esta corrección o mejora será aprovechada por todas las aplicaciones que compartan la biblioteca. [14]

### 2.9.5. NORMALIZACIÓN.

Cuando se diseña una base de datos mediante el modelo relacional, al igual que ocurre en otros modelos de datos, tenemos distintas alternativas, es decir, podemos obtener diferentes esquemas relacionales y no todos son equivalentes, ya que algunos van a representar la realidad mejor que otros.

Es necesario conocer qué propiedades debe tener un esquema relacional para representar adecuadamente una realidad y cuáles son los problemas que se pueden derivar de un diseño inadecuado.

La normalización es una técnica para diseñar la estructura lógica de los datos de un sistema de información en el modelo relacional, desarrollada por E. F. Codd en 1972. Es una estrategia de diseño de abajo a arriba: se parte de los atributos y éstos se van agrupando en relaciones (tablas) según su afinidad. Aquí no se utilizará la normalización como una técnica de diseño de bases de datos, sino como una etapa posterior a la correspondencia entre el esquema conceptual y el esquema lógico, que elimine las dependencias entre atributos no deseadas. Las ventajas de la normalización son las siguientes:

- Evita anomalías en inserciones, modificaciones y borrados.
- Mejora la independencia de datos.
- No establece restricciones artificiales en la estructura de los datos.
- Es una técnica para la evaluación, redundancia y consistencia de los datos en un modelo.
- Busca eliminar anomalías de actualización.

- Evita redundancia en la información.

La normalización se lleva a cabo en una serie de pasos. Cada paso corresponde a una forma normal que tiene unas propiedades. Conforme se va avanzando en la normalización, las relaciones tienen un formato más estricto (más fuerte) y, por lo tanto, son menos vulnerables a las anomalías de actualización. El modelo relacional sólo requiere un conjunto de relaciones en primera forma normal. Las restantes formas normales son opcionales. Sin embargo, para evitar las anomalías de actualización, es recomendable llegar al menos a la tercera forma normal. [15]

#### **2.9.5.1. Primera Forma Normal (1FN)**

Una relación está en primera forma normal (1FN) si y sólo si todos los dominios son atómicos. Un dominio es atómico si los elementos del dominio son indivisibles. Es decir, no tenemos grupos de repetición o un conjunto de valores asociados repetidos asociados a una misma tupla.

En la siguiente figura (Fig. II.8) de la tabla Carrera demostramos como se aplicó la 1FN



id_carrera	Nombre_carre	Plan
1	Informatica	Plan C
2	Telecomunicac	Plan C
(Autonumérico)		

*Figura II.8. 1FN / SAIRFIT*

### 2.9.5.2.Segunda Forma Normal (2FN)

Una relación está en segunda forma normal (2FN) si y sólo si está en 1FN y todos los atributos que no sean llaves dependen por completo de llave primaria.

La 2FN se aplica a las relaciones que tienen claves primarias compuestas por dos o más atributos. Si una relación está en 1FN y su clave primaria es simple (tiene un solo atributo), entonces también está en 2FN. Las relaciones que no están en 2FN pueden sufrir anomalías cuando se realizan actualizaciones.

En la figura que se muestra a continuación, demostramos cómo se usó la 2FN en SAIRFIT en la tabla Premios. Donde todos los atributos de la tupla dependen por completo de la clave principal id\_premio.

id_premio	instancia_pre	nombre_premio	institucion_premio
1	Universidad	Relevante	MES
2	Facultad	Intelectual	FC T
3	Ministerio	Destacado en Docensia	MES
4	Municipio	Relevante	FC T

*Figura. II.9. 2FN / SAIRFIT*

### 2.9.5.3. Tercera Forma Normal (3FN)

Una relación está en tercera forma normal (3FN) si y sólo si está en 2FN y todos los atributos no llave dependen de manera no transitiva de la llave primaria. Se dice que existe una dependencia transitiva cuando tenemos el par de dependencias funcionales.

Aunque las relaciones en 2FN tienen menos redundancias que las relaciones en 1FN, todavía pueden sufrir anomalías frente a las actualizaciones. Para pasar una relación de 2FN a 3FN hay que eliminar las dependencias transitivas. Para ello, se eliminan los atributos que dependen transitivamente y se ponen en una nueva relación con una copia de su determinante (el atributo o atributos no clave de los que dependen). [15]

id_departamento	Nombre_dep	id_carrera
8	Matematicas	1
9	Telecomunicaciones	2
11	Informatica	4
(Autonumérico)		0

*Figura II.10. 3FN / SAIRFIT*

### 2.9.6. DELPHI 6

Delphi, no tiene un lugar tan claramente definido; es lo que se llama un lenguaje de propósito *general*. Esto significa que se comporta bien ante tipos diferentes de problemas, y Borland ha puesto mucho empeño en lograr que el rendimiento sea el mejor posible. Las aplicaciones pueden colocarse de forma muy sencilla en la pantalla según el principio de módulos. Para ello se dispone de una paleta dotada de una gran variedad de componentes, algo así como los bloques de construcción de cada programa. Esta paleta es denominada por Borland VCL (Visual Component Library), o biblioteca de componentes visuales. Tiene un aspecto similar a Visual Basic, pero aunque el aspecto externo indica la misma facilidad de uso que Visual Basic, el corazón del sistema Delphi es mucho más potente. [16].

Delphi se puede usar para casi cualquier tipo de programa, obteniéndose un rendimiento excelente con facilidad. Destaquemos entonces algunas características de Delphi:

- Posee un Entorno de Desarrollo Integrado para Windows (IDE), con características de programación visual. Esto significa que la mayor parte del programa se hace gráficamente con el mouse o interactuando con los objetos en tiempo de diseño sin necesidad de compilar cada vez para ver los resultados, ahorrando tiempo y esfuerzo.

- El lenguaje Pascal ha sido ampliado y mejorado, sobre todo para adaptarlo a la programación con Objetos.
- La programación se hace más intuitiva y sencilla con el uso de componentes.
- Incluye un soporte de Bases de Datos poderoso y fácil de usar.
- Genera ejecutables nativos, sin necesidad de librerías de run-time.
- Se pueden crear componentes nuevos que se integran en el entorno de la misma manera que los nativos.

#### **2.9.6.1.Componentes**

**Delphi introdujo la idea del uso de componentes, que son piezas reutilizables de código (clases) que pueden interactuar con el EID en tiempo de diseño y desempeñar una función específica en tiempo de ejecución. Desde un enfoque más técnico, se catalogan como componentes todos aquellos**

**objetos que heredan desde la clase TComponent.**

**Una gran parte de los componentes disponibles para**

**Delphi son controles (derivados de TControl), que**

**encapsulan los elementos de interacción con el**

**usuario (Botones, menús, barras de desplazamiento,**

**etc). [17].**

Además de poder utilizar en un programa los componentes estándares (botones, grillas, conjuntos de datos, etc.), es posible crear nuevos componentes o mejorar los ya existentes, extendiendo la funcionalidad de la herramienta.

### **2.9.6.2.Eventos**

Delphi nos permite de manera sencilla ejecutar trozos de código en respuesta a acciones o eventos que ocurren durante el tiempo que un programa se ejecuta. Por ejemplo, cuando se presiona un botón, la VCL captura la notificación estándar de windows, y detecta si hay algún método asociado al evento OnClick del botón. Si lo hay, manda ejecutar dicho método. De esta manera podemos responder a esta acción del usuario ejecutando un proceso, cerrando el formulario, etc.

Los eventos pueden generarse debido a la recepción de señales desde elementos de hardware como el ratón o el teclado, o pueden producirse al realizar alguna operación sobre un elemento de la propia aplicación (como abrir un conjunto de datos, que genera los eventos BeforeOpen/AfterOpen). La VCL ha comprobado estar bien diseñada y el control que se tiene a través de los eventos de la misma es suficiente para la gran mayoría de aplicaciones. [17].

### **2.9.6.3.Desarrollo Visual**

Como entorno visual, la programación en Delphi consiste en diseñar los formularios que componen al programa colocando todos sus controles (botones, etiquetas, campos de texto, etc.) en las posiciones deseadas, normalmente usando un ratón. Luego se asocia código a los eventos de dichos controles y también se pueden crear

módulos de datos, que regularmente contienen los componentes de acceso a datos y las reglas de negocio de una aplicación. [17]

#### **2.9.6.4. Depurador Integrado**

Es una potente característica que nos permite establecer puntos de ruptura (breakpoints), la ejecución paso a paso de un programa, el seguimiento de los valores de las variables y de la pila de ejecución, así como la evaluación de expresiones con datos de la ejecución del programa. Con su uso, un programador experimentado puede detectar y resolver errores lógicos en el funcionamiento de un aplicativo desarrollado con Delphi. [17]

#### **2.9.6.5. Ambiente de Desarrollo Integrado (IDE)**

El ambiente de desarrollo integrado (IDE por las siglas de Integrated Development Enviroment) es un ambiente en el que están presentes todas las herramientas necesarias para diseñar, ejecutar o probar una aplicación, además estas herramientas se encuentran bien relacionadas para facilitar el desarrollo de programas.

El IDE consta de un editor de código, un depurador de errores, una barra de herramientas, un editor de imágenes y de herramientas de base de datos, todos ellos operando de manera integrada. La integración ofrece al desarrollador un conjunto de



herramientas que operan en armonía y se complementan entre sí. El resultado es un desarrollo de aplicaciones complejas más rápido y libre de errores. [18]

#### **2.9.6.6. Acceso desde Delphi a las Bases de Datos**

El esquema de acceso a los datos utilizado en Delphi hace uso de varias *capas lógicas* entre el archivo físico y los controles (editores, etiquetas, etc) con los cuales interactúa el usuario. De esta manera se logra una cierta independencia no sólo del formato, sino también de la forma de acceder a los mismos. El programador puede disponer de la forma de acceso que considere adecuada a cada situación, mientras mantiene el núcleo de la interface sin cambios. Así si crea su programa utilizando tablas y luego decide utilizar consultas SQL o algún otro tipo de acceso más adecuado a las condiciones de ese momento puede hacerlo sin mayores complicaciones.

**Por compatibilidad con diferentes Sistemas Gestores de Bases de Datos nuestras opciones son el ADO y el ODBC. Que son sistemas ideados para programar de una forma transparente en una**

**aplicación, sin importarnos que BD tenemos  
realmente. [19]**

### **2.9.6.7.ADO / Delphi 6**

ADO (Microsoft ActiveX Data Objects) es un componente de Microsoft que permite acceder a muchos gestores de base de datos entre ellos Access de forma sencilla. ADO no esta asociado con ningún lenguaje de programación y puede usarse desde cualquiera que soporte COM. Para que una aplicación que use ADO pueda finalmente acceder a los datos necesita hacerlo mediante un proveedor OLE DB. Un proveedor OLE DB es un interfaz de bajo nivel para acceder a datos en una organización. Es un estándar abierto, a semejanza del ODBC. [20]

ODBC es el sistema planteado inicialmente por Microsoft en sus Sistemas Operativos Windows para el acceso a Bases de Datos. Una de las quejas más concurrentes era su lentitud, y con ese fin nació ADO, además de tener una mayor orientación hacia las Bases de Datos relacionales. Realmente, el futuro de conectividad para Bases de Datos de una forma general sería fundamentalmente a través de ADO. Hay que tener en cuenta un punto a la hora de desarrollar aplicaciones, sería sobre que plataforma deseamos ejecutar nuestra aplicación. Si decidimos que sobre una plataforma Windows, ADO es perfectamente recomendado. [18]

Usar los componentes ADO permite que los desarrolladores de Delphi elaboren aplicaciones para bases de datos que no dependan del Borland Database Engine (BDE). [21]

### 2.9.6.8. Objetos ADO

Los objetos ADO que más figuran son:

- **DataSource:** es el enlace entre los componentes de datos y los de acceso a tablas. Se relaciona con un tDataset (Delphi trae definidas las clases ADOTable y ADOQuery, pero podemos crear el que necesitemos) a través de la propiedad DataSet.
- **ADOTable:** nos da acceso a una tabla con métodos para navegar por ella y acceder a los registros. Se relaciona con el archivo físico (la tabla real) a través de las propiedades DatabaseName y TableName. Los controles de datos se comunican con estos componentes a través de un tDataSource.
- **ADOQuery:** nos da acceso a los datos de una Base de Datos a través de comandos SQL, con los cuales podemos acceder a más de una tabla relacionada, crear campos calculados, etc. Las características SQL son las que pone a nuestra disposición el motor de datos, por lo que podremos aprovechar todas las opciones avanzadas que da cada servidor particular. [21]

### **2.9.6.9.REPORTES QREPORT**

Imprimir tablas o construir reportes sobre las mismas, es también una tarea común en los sistemas de información basados en bases de datos.

**Lo bueno es que Delphi tiene una librería de componentes llamada QReport cargada de componentes visuales especializados en la impresión de reportes. [22]**

#### **2.9.6.9.1. Componentes de QReport**

**El QReport consta con algunos componentes de los cuales detallaremos a continuación los usados para generar los reportes en nuestro sistema:**

- **QuickRep.** Se encuentra en la pestaña QReport. Es el componente principal para impresión. Toma la forma y tamaño de una hoja de impresora dentro de Form1.
- **QRLabel.** Similar a label, contienen textos o mensajes estáticos. Colocado dentro del rectángulo o sección de encabezado.

**Propiedades: Caption = NOMBRE + Font = dar color y tamaño de font**

- **QRDBText.** Componente para desplegar cada dato de una celda o campo o columna de una tabla, es similar a DBEdit pero es estático, es decir solo despliega. Se deberán poner tantos componentes QRDBText en esta banda o sección de detalle,

**como columnas de datos existan o se quieran desplegar de la tabla.**

- **QRDBImage. Componente para desplegar e imprimir campos de tipo Graphics (BMP, ICO, WFW). [22]**

#### **2.9.6.9.2. Componentes de Bandas de QReport**

**QReport trae incorporado un componente llamado QRBand que permite dividir en partes, secciones o bandas todo el reporte. Este componente QRBand, en su propiedad BandType, permite construir los siguientes seis tipos importantes de banda:**

- **PAGE HEADER** Banda de encabezados de páginas. **Aparece en todas las hojas impresas. Pone dentro componentes para número de página, etc.**
- **TITLE** Banda de Título o encabezados del Reporte. **Sólo aparece en la primera hoja impresa. Permite poner componentes qlabel con el nombre de la compañía, el propósito de la tabla, fecha, etc.**
- **COLUMM HEADER** Banda de o para encabezados de columnas. **Aparece en todas las hojas impresas. Pone dentro componentes qlabel con el encabezado que lleva cada columna de la tabla.**
- **DETAIL** Banda de detalle. **Es la más importante, porque es la que contiene los datos de los renglones de la tabla. Es la que constituye el**



**grueso de las hojas impresas. Solo se colocan dentro de esta banda los componentes de datos impresos, QRDBText y QRDBImage.**

- **SUMMARY Banda de resumen. Sólo aparece en la hoja final de impresión. Pone dentro de ella componentes de operaciones como es QRExpr.**
- **PAGE FOOTER Banda de pie de página. Aparece en todas las hojas impresas. Su función es similar a la banda de encabezado. De hecho, en un reporte solo se usa la banda de encabezado o la banda de pie de página, pero no las dos. Sólo pone dentro de esta banda, componentes de número de pagina, o de fechas, o de hora, etc. [23]**

#### **2.9.6.10. Ventajas de Delphi 6 con respecto a Visual Basic.**

- Delphi es más lenguaje de "todo propósito" que Visual Basic
- El rendimiento de Delphi no lo limita a simples aplicaciones de interfaz de usuario. Con Delphi uno puede crear DLLs, objetos ActiveX, servidores y aplicaciones de servicios, etc.
- Visual Basic es la elección de los ejecutivos, los principiantes y de quienes se ven forzados a utilizarlo para conseguir un o simplemente para seguir las masas. Delphi es la opción de los desarrolladores independientes.
- Delphi produce aplicaciones independientes que son pequeñas y entregan gran rendimiento, haciéndolas más amistosas con los recursos del sistema y los anchos de banda de los módems, haciendo de Delphi una opción ideal sobre Visual Basic para los desarrolladores independientes.
- Kylix lleva Delphi a la plataforma Linux, adonde VB no irá.

- Visual Basic hace más fáciles las cosas fáciles, Delphi hace más fáciles las cosas difíciles" <sup>4</sup>

---

<sup>4</sup> Computerworld, 1998

## **CAPITULO III**

### **DESARROLLO DEL SISTEMA**

En este último capítulo detallaremos como está constituido el Software, su funcionamiento y sus aspectos relevantes y fundamentales para una mejor comprensión y manipulación del mismo por parte de los beneficiados con este sistema.

#### **3.1. PANORÁMICA GENERAL DE SAIRFIT**

El Sistema de Almacenamiento de Información Relevante de la Facultad de Informática y Telecomunicaciones (**SAIRFIT**) fue creado en el lenguaje de cuarta generación (Delphi 6) y su base de datos en Microsoft Access 2000 en la plataforma Windows, permitiendo la creación de una interfaz y a la vez haciendo muy fácil su manejo tanto para el usuario como para el Administrador del sistema.

## **3.2. OBJETOS UTILIZADOS PARA CREAR LA INTERFAZ**

### **3.2.1. Tablas**

Una tabla de datos es un objeto que se define y utiliza para almacenar los datos, una tabla contiene información sobre cada tema, se almacenara en un espacio de memoria que será asignada en el disco, un ejemplo es la tabla de premios los datos como el Id de Premio, Nombre, Instancia e Institución, todos estos campos vienen a constituir un conjunto de de registros.

Un registro es el conjunto de información referida a un mismo premio u objeto. Un registro vendría a ser algo así como una ficha.

El campo Id o código nos permite identificar y localizar un registro de manera ágil y organizada, su característica (Autonumérica) mientras que Nombre e Institución tienen su característica (Texto).

Además cada campo posee características adicionales que eventualmente dependen de su tipo. Por ejemplo, los campos textuales poseen una extensión máxima en número de caracteres que se requieran.



La **llave o clave primaria** nos permitirá identificar a la tabla, permitiéndonos para que se cumpla estas características:

- Determina el orden de los registros en la tabla.
- No nos permite ingresar dos registros con igual contenido de llaves idénticas (Ver Fig.: III.1)

Nombre del Campo      Tipo de dato

Llave o Clave Primaria →

Nombre del campo	Tipo de datos
id_premio	Autonumérico
instancia_premio	Texto
nombre_premio	Texto
institucion_premio	Texto

Características

General	Búsqueda
Tamaño del campo	Entero largo
Nuevos valores	Incrementalmente
Formato	
Título	
Indexado	Sí (Sin duplicados)
Etiquetas inteligentes	

*Figura III.1 Tabla Premios.*

### 3.2.2. Formularios

Son los tipo de objeto que se utiliza para la interfaz fundamentalmente para entrada y salida de datos, a la vez para la manipulación de los de datos; además los formularios nos permiten un funcionamiento más ágil y sencillo para acceder a la información de

una manera rápida a la información que se requiera, realizando búsquedas y consultas.

En este sistema se crearon formularios de acuerdo a las necesidades para ingresar, eliminar, editar los registros que se requieren de la base de datos de una o varias tablas y a la vez permitiéndonos la salida de reportes y consultas, como ejemplo tenemos el formulario Profesores el cual se relaciona con las tablas Provincia, Municipio, Departamento, Investigación y Registro de software.

### **3.2.3. Subformularios**

Un subformulario es creado con la finalidad de permitirlo abrir cuando sea necesario, o al momento que se lo requiera dentro del formulario principal.

Cuando combinamos estos formulario/subformulario se los denomina a menudo jerárquico, principal/detalle o principal/secundario.

Estos subformularios son muy efectivos cuando se los utiliza con datos de tablas o consultas que se necesitan como ejemplo tenemos el Formulario de Usuario donde podemos obtener datos de los alumnos, profesor o lo que el usuario requiera.

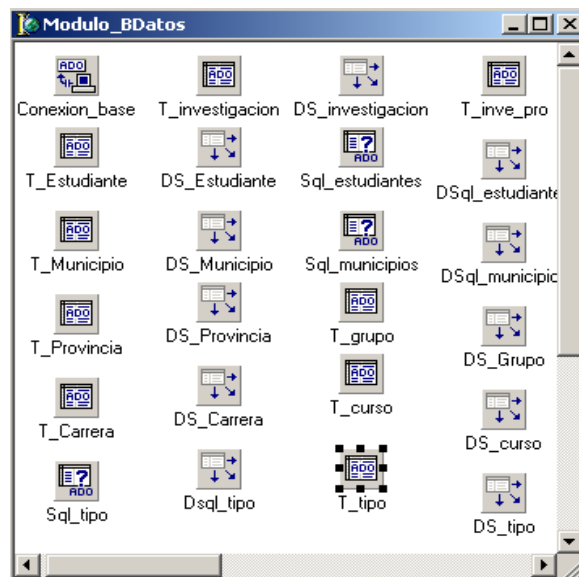


### 3.2.4. Informes

Los informes son muy esenciales en el sistema para poder obtener datos, esta dirigido a la emisión e impresión del contenido de una tabla o consulta, como ejemplo tenemos la emisión de las listas de alumnos de cada uno de los años de cada carreras.

### 3.2.5. Módulos

Los módulos nos permiten crear procedimientos y poner controles de conexión para poder utilizarlos dentro de cualquier formulario que se lo requiera en el sistema, Ejemplo del modulo de conexión de la base de datos (Ver Fig.: III.2).



*Figura III.2: Modulo de Conexión a los Datos*

### 3.2.6. Los controles

Los controles permiten el manejo de los datos de una forma rápida y fácil, tanto para el ingreso de datos como para la decoración de cada uno de los formularios del sistema existen dos tipos de controles los dependientes e independientes.

#### Tipos de Controles

- **Dependientes.-** Son aquellos que están conectados a un campo de una tabla o consulta mediante el Data Source y asignado al campo mediante el Data Field, se utilizan para mostrar, introducir, editar y actualizar valores de los campos de la base de datos como ejemplo tenemos el control DBEdit1.

Estos controles son puestos en cada uno de los formularios de acuerdo a los campos que se utilizan de acuerdo a los registros de base de datos.

- **Independientes.-** Estos campos se utilizan de acuerdo a la necesidad, permiten asignar datos directamente, se pueden utilizar para mostrar información, para agrupar un conjunto de controles, asignar imágenes, independientemente de de la tabla.

A continuación mostraremos los controles que se utilizaron en el desarrollo del sistema:



### **Etiqueta**

Este componente se utiliza para desplegar textos o mensajes estáticos dentro de las formas, textos tales como encabezados, solicitud al usuario del sistema.



### **Edit**

Este control es importante de visual Borland Delphi, su función principal es manejar todos los procesos de entrada y salida (input/output) al programa.



### **MaskEdit**

Este control es muy similar en su uso al control Edit, excepto que proporciona una máscara especializada para el formato de datos, es decir

se puede usar para que el usuario proporcione datos con formato bien definidos.



## **ComboBox**

Este control nos permite definir en primera instancia un conjunto de datos o valores respuestas asociados a una caja de edición cualesquiera, así ahora el usuario tendrá la oportunidad de seleccionar un dato del conjunto de datos o respuestas ya predefinidos.



## **GroupBox**

Este control es otra forma standard de agrupamiento de controles del sistema, se usa para agrupar componentes relacionados dentro de un formulario.



## **CheckBox**

Este control permite seleccionar una opción al usuario del programa o tomar una decisión, directamente en el formulario.



## **MainMenu**

Permite insertar menús dentro de la aplicación. En este menú se puede poner el control de la aplicación.



**Botton**

Este control permite el control del sistema.



**Image**

Permite insertar imágenes en todos los formularios que se requieran en el sistema.



**ADOCnection**

Este control permite conectarnos a una base de datos, entre ellas Access de forma sencilla, colocando dentro de un data módulo para ser utilizada dentro del todo el sistema.



**ADOTable**

Este control es usado para extraer y operar un conjunto de datos de una sola tabla; se conecta a la base de datos de manera directa o con un TADOConnection.



### **ADOQuery1**

Este control tiene como finalidad extraer y operar sobre conjuntos de datos producidos por una sentencia SQL. También puede ejecutar sentencias de definición de datos (DDL). Su conexión es similar a la de ADOTable.



### **DataSource**

Este control nos permite la conexión de la ADOTable con el ADOQuery para posteriormente conectar con cada uno de los controles del Data Control.



### **DBText1**



Este control permite desplegar títulos o conectar mediante el Data Source para que nos desplace el nombre del campo.



### **DBEdit1**

Este control es importante ya que permite el ingreso de datos, para editar se conecta mediante el Data Source a cada campo.



### **DBNavigator1**

Control que permite la navegación de cada uno de los datos existentes dentro de una tabla.



### **DBGrid**

En este control nos permite presentar todos los renglones a la vez al usuario para su procesamiento, su característica interesante es que trae asociado un editor de columnas.



### **DBImage**

Este control nos permite desplegar los gráficos o imágenes, que se han convertido en una de las grandes atracciones y características de las bases de datos.



### **DBComboBox**

Control que permite definir en primera instancia un conjunto de datos o valores respuestas asociados a una caja de edición cualesquiera, así ahora el usuario tendrá la oportunidad de seleccionar para posteriormente ser asignados al campo que este se encuentre enlazado.



### **DBLookupComboBox**

Control que permite desplegar datos de una tabla para ser asignados posteriormente en otra tabla que requiera ese campo.

## **QReport**



Control que permite la creación de impresiones de reportes, que sean necesarios en el sistema.

### **3.2.7. Propiedades del formulario**

Las propiedades del formulario nos permiten controlar que estos tengan una buena presentación, por lo cual son muy importantes (Ver Fig.: III.3).

- **ActiveControl:** En este se pone el nombre del objeto que tendrá el enfoque cuando se ejecute la aplicación, de no poner nada, tendrá el enfoque el objeto que se insertó primero.
- **AutoScroll:** Si tiene el valor True, el formulario presentará barras de desplazamiento que permitan acceder a todas las partes del mismo.
- **AutoSize:** Si tiene el valor True, el ancho y alto del formulario se ajustará automáticamente para que ocupe el menor espacio posible conteniendo a todos los componentes.
- **BorderIcons:** Propiedad compuesta que determina si los botones de maximizar, minimizar, salir, estarán visibles en el formulario.

- **BorderStyle:** Determina como será el borde del formulario.
- **BorderWidth:** Contiene el tamaño en píxeles del borde de la ventana.
- **Caption:** Texto que aparece en la parte superior del formulario.
- **Color:** Color de fondo del formulario.
- **Name:** Nombre del formulario en código.
- **Position:** Posición que ocupará en la pantalla el formulario.



*Figura III.3: Propiedades del formulario*

### **3.2.8. Métodos de los formularios.**

Son las acciones que se pueden realizar sobre un formulario, entre las más comunes tenemos las siguientes (Ver Fig.: III.4).

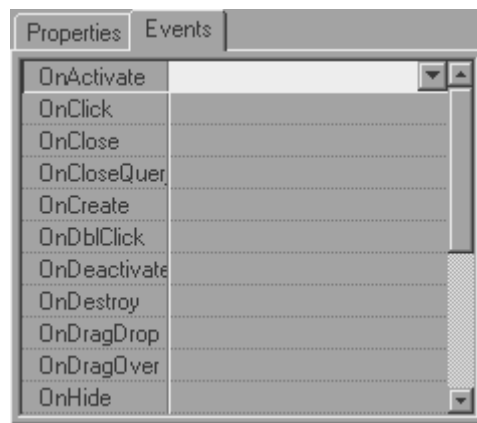
- Close: Invocando a este procedimiento, el formulario se cierra.
- Show: Este procedimiento hace que el formulario aparezca en pantalla.
- ShowModal: Esta función hace aparecer el formulario, y le da el control de la aplicación, para que el usuario pueda acceder a sus botones, cuadros de texto. Mientras el usuario no cierre este formulario, no puede seguir con la aplicación.
- Hide: Oculta el formulario. Es el procedimiento contrario a Show.

### **3.2.9 Eventos.**

Es la otra mitad de Object Inspector. Se relaciona con el programador y los diferentes eventos a los que puede responder un objeto o formulario (Ver Fig.:4).

- OnActive: permite realizar algunas líneas de instrucción que se ejecutan cuando se ha abierto el formulario.

- OnClik: permite realizar algunas líneas de instrucción dentro de cualquier objeto y se ejecutan cuando se de un clic en objeto.
- Realiza la mismo la misma función cualquier evento dependiendo de cada línea de instrucción que se la asigne y ejecuta cuando se de clic dentro del formulario o objeto.



*Figura III.4: Eventos de un formulario*

### **3.3. INTERFAZ DEL USUARIO.**

La interfaz que ha sido creada en SAIRFIT tiene un Formulario de entrada al sistema, como fondo tiene el logo tipo que lo representa, tenemos un grupo de botones (Administrador, Usuario y Salir) que indican a donde podemos ingresar y cada uno de estos realiza una función específica (Ver Fig.: III.5).



Figura III.5: Formulario de entrada del sistema

- **ADMINISTRADOR.** El Botón Administrador del formulario principal nos permite desplegar un cuadro de diálogo en el que ingresaremos los datos del administrador, si verifica que los datos ingresados en el cuadro de diálogo son correctos abre el Formulario Administrador (Ver Fig.: III.6). Caso contrario nos permitirá hacer tres intentos y se bloqueara el botón administrador del formulario.

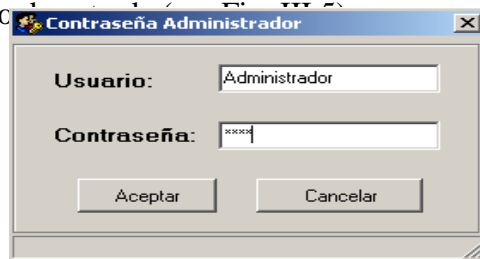


Figura III.6: Cuadro de dialogo

El Formulario Administrador es uno de los principales del sistema ya que nos permite desplegar los Subformularios de entrada de datos a cada uno de las tablas, en su fondo tenemos el sello de la universidad a cual pertenece la F.I.T., tiene un

menú principal el cual contiene cada uno de los Subformularios para realizar las diferentes tareas a las que fueron asignadas, como son el Registro de Profesores, de Estudiantes, de premios, registro de software, publicaciones e Investigaciones y Proyectos, además de una opción denominada Adicionar que permite agregar datos a tablas que son útiles dentro del sistema como ejemplo Departamento y las visitas que este recibe. (Ver Fig.: III.7).



*Figura III.7: Formulario Administrador*

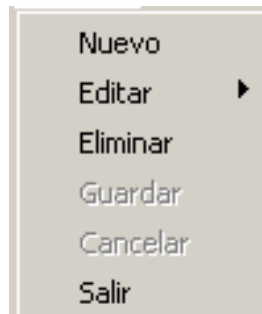
A continuación detallaremos algunos de los formularios principales de SAIRFIT dentro del Administrador.

**Formulario Registro de Profesores**, consta de diferentes tareas importantes como son:



## Archivo

- Ingresar nuevos datos de profesor.
- Editar datos de profesores que ya existen.
- La eliminación de datos de profesores.



## Foto.

- Además este formulario nos permite ingresar una fotografía del profesor previamente guardada en la carpeta galería dentro del sistema identificada con el número del Carné de Identidad correspondiente.



## Herramienta.

- Aquí se oculta o no la barra para navegar por los registros
- También permite navegar por todos los registros mediante el DBNavigator.



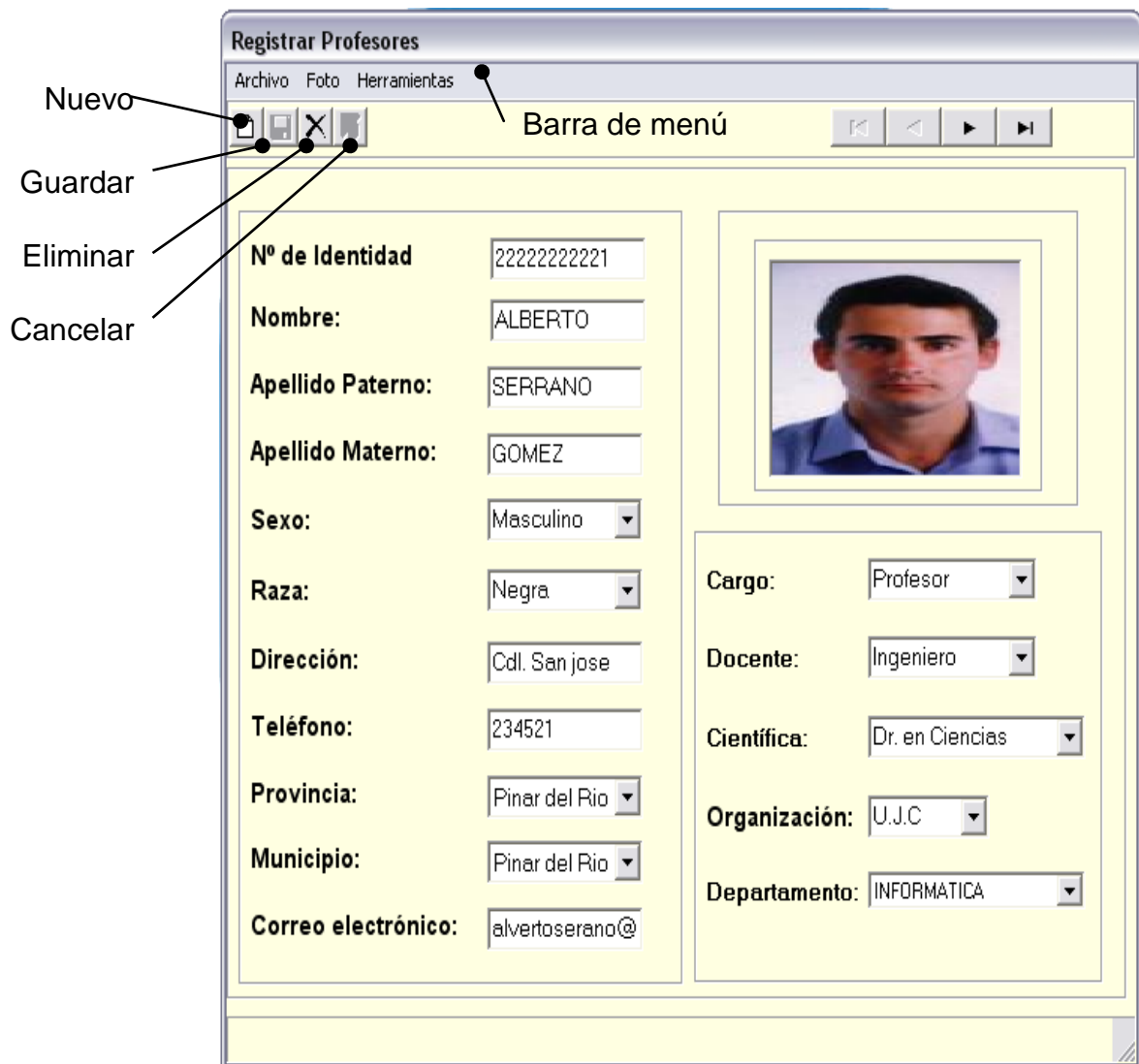


Figura III.8: Registro de Profesores

**Controles de Datos.** El control del campo Foto permite que no se produzcan errores en la ejecución del programa ya que si no existe el archivo le asignara un archivo existente el cual muestra que la fotografía no está disponible. (Ver Fig.: 9).



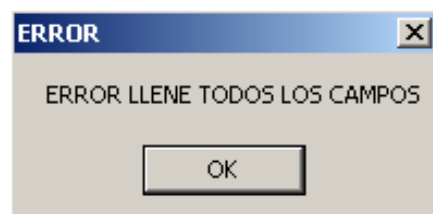
*Figura III.9: Campo de Asignación de Foto*

Cada uno de los cuadros de textos se encuentra debidamente controlado para ingresar sólo caracteres o sólo números dependiendo de la necesidad de cada campo y de la tabla a la cual están conectadas, estos controles de errores son similares en cada uno de los formularios de Administrador, permitiéndonos brindar consistencia en los datos ingresados. ste mensaje controla que no debe repartirse las claves primarias en los formularios. (Ver Fig.: 10).



*Figura III.10: Mensaje de claves duplicadas*

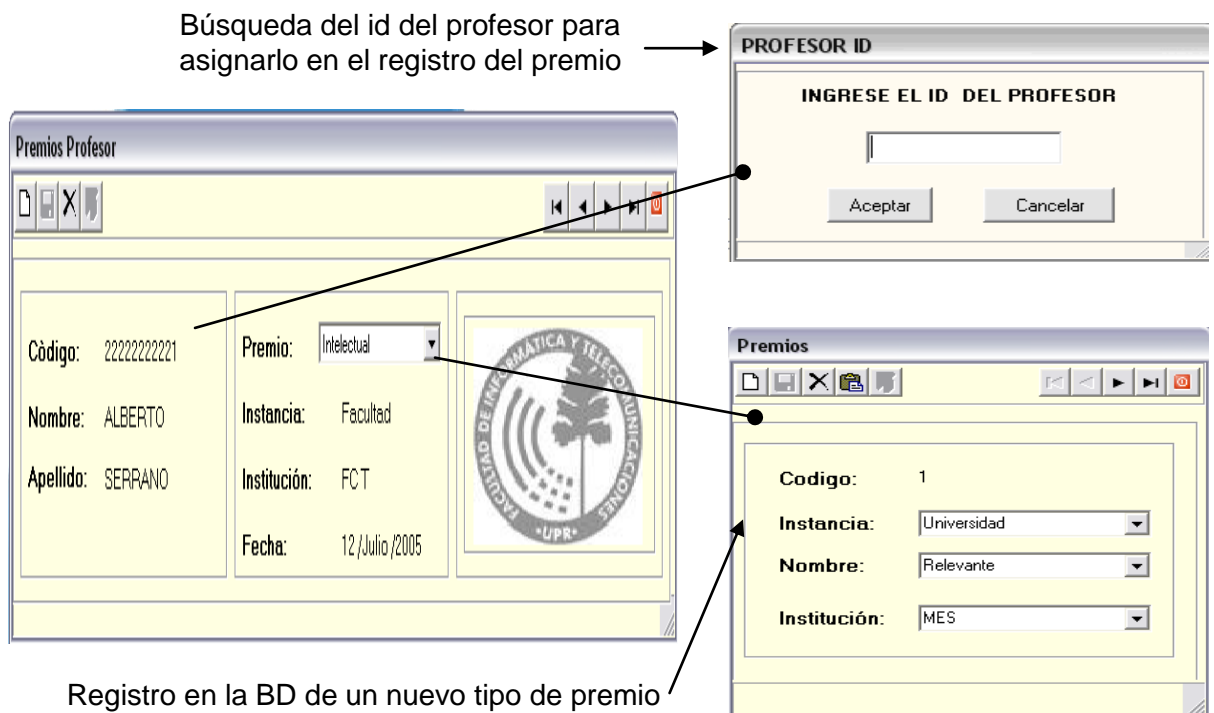
Nos permite controlar que todos los campos estén llenos este mensaje se produce cuando queremos grabar campos importantes y están vacíos (ver Fig.: 11).



*Figura III.11: Mensaje de claves duplicadas*

**Formulario Registro Estudiante**, en este formulario se ingresan los datos de los estudiantes de la F.I.T. Sus funciones son similares al de registro de profesor y su estructura es parecida.

**Formulario Premios Profesor**, realiza las funciones más comunes para el registro de datos, abre un subformulario en el cual ingresamos el Id de Profesor permitiéndonos buscar y asignar los datos del mismo para posteriormente ser guardados en la tabla, y después almacenar los datos del premio que dicho profesor ha obtenido, en caso de no haber el premio tanto como para el Profesor como el Estudiante, podemos registrarlo en el formulario premios el cual se lo abre cuando sea requerido. (Ver Fig.: III.12).



*Figura III.12: Registro de Premios\_Profesor (el registro de Premios del Estudiante se lo realiza de la misma manera)*

**Formulario Registro de Software**, nos permite almacenar datos de los Software que se han creado dentro de la Facultad, se registrará además los datos de las personas que lo han desarrollado, mediante el botón Agregar Profesor que nos permite ingresar a todas las personas que han desarrollado dicho software, el cual abre un subformulario para la búsqueda de datos del profesor para posteriormente ser asignados, este formulario también realiza las funciones principales de nuevo, editar, eliminar, etc. (Ver Fig.: III.13).

Registrar Software

Archivo Herramientas

Nº REGISTRO: SFW00001

Nombre: SISGECO

Código: 22222222222

Nombre: OSMAI

Apellido: CABREPA

PROFESOR ID

INGRESE EL ID DEL PROFESOR

Aceptar Cancelar

Agregar Profesor

Nuevo Software

*Figura III.13: Registro de Software*

**Formulario Registrar Publicaciones**, este formulario permite realizar las mismas funciones que los anteriores, registrando cada una de las publicaciones realizadas por los profesores de la Facultad, el botón **Agregar Profesor** nos permitirá ingresar la cantidad de profesores y la fecha en la que realizaron la publicación. (Ver Fig.: III.14)

*Figura III.14: Registro de Publicaciones*

**Formulario Registro de Investigaciones y Proyectos**, permite ingresar datos de investigaciones realizadas y de proyectos que se estén desarrollando en la Facultad.

En este formulario las validaciones de errores son similares al formulario de registro de Profesores con la diferencia de que en este formulario controlamos fecha de inicio y fin, donde la fecha final no puede ser menor a que la fecha que culminan las Investigaciones. Además el campo del día está validado de una manera que controle que los meses tengan 28, 29, 30 o 31 días dependiendo del mes escogido, estas validaciones también son utilizadas en el Registro de Proyectos. (Ver Fig.: 15).



*Figura III.15: Registro de Investigaciones*

En el menú principal de administrador hay una opción denominada Adicionar que permite agregar datos a otras tablas que son útiles dentro del sistema para brindar información, como ejemplo Departamento y las visitas que este recibe, municipios existentes, etc.

**Formulario Registro de Visitas**, registra cada una de las visitas que se han realizado a cada uno de los departamentos, con su respectiva fecha, visitador e



instancia, realiza también todas las funciones principales de almacenamiento. (Ver Fig.: 16).

Registrar visitas

Archivo

N° De visita: V00007

Institucion: Universidad

Instancia: FC T

Funcionario: Dr. Francisco Orbea

Departamento:

MATEMATICAS  
TELECOMUNICACIONES  
INFORMATICA  
DECANATO

Fecha: 18 /Agosto /2004

Agregar Departamento

Figura III.16: Registro de Visitas

Otro de los formularios que se encuentran dentro de la opción Adicionar es el de ingresar Provincia, Municipio como lo mostramos a continuación (Ver Fig. III.17)

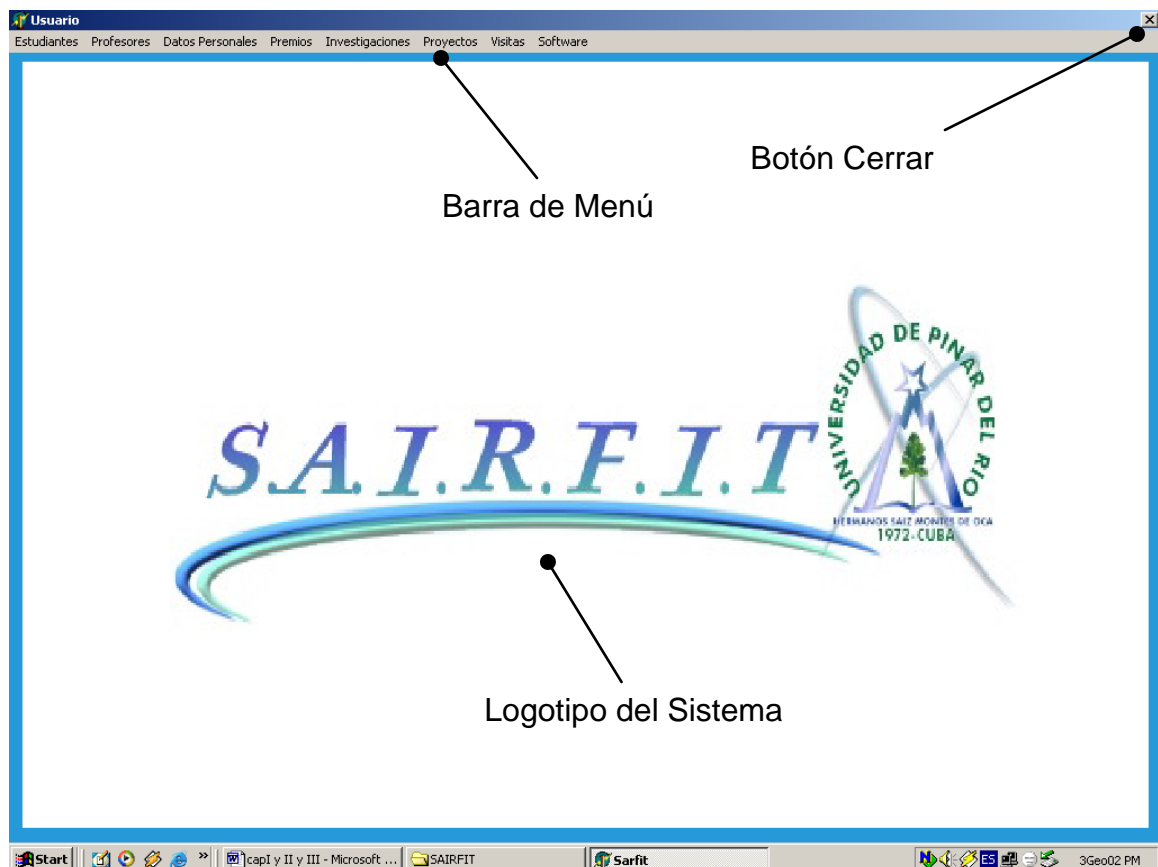
Provincia

Provincia | Municipio | Busqueda

Codigo: 2

Nombre: Pinar del Rio

- **USUARIO.** El Botón Usuario (ver Fig.: III.5) nos permite desplegar el Formulario de Usuario el cual contiene consultas y reportes, restringe que los datos sean cambiados al momento que estos se muestran, este formulario tiene de fondo el logotipo del sistema, una barra de menú en la parte superior para desplegar lo que el usuario desea consultar y un botón en la parte superior derecha que nos permitirá cerrar este formulario y retornar al formulario de entrada del sistema. (Ver Fig.: III.18).



*Figura III.18: Formulario Usuario*

**Lista de Estudiantes** nos permite realizar una consulta de los datos de los estudiantes que tienen una carrera en determinado curso, grupo y tipo de curso, presionando el botón nueva búsqueda en la parte inferior izquierda y seleccionando las opciones de búsqueda, además cuenta con el botón imprimir que nos permite llamar al reporte generado en la consulta (Ver. Fig. III.20) e imprimirlo si es necesario o a su vez guardarlo para cuando el usuario lo requiera, y por supuesto el botón salir del formulario (Ver Fig.: III.19).

Facultad de Informática y Telecomunicaciones

Carrera: Informática    Curso: 1  
Tipo: Diurno    Grupo: A

Nombre	Apellido Paterno	Apellido Materno	Sexo
Luis	Escobar	Bandt	Masculino
Eduardo	Alcocer	Valgas	Masculino
Cesar	Alvarez	Zuñiga	Masculino

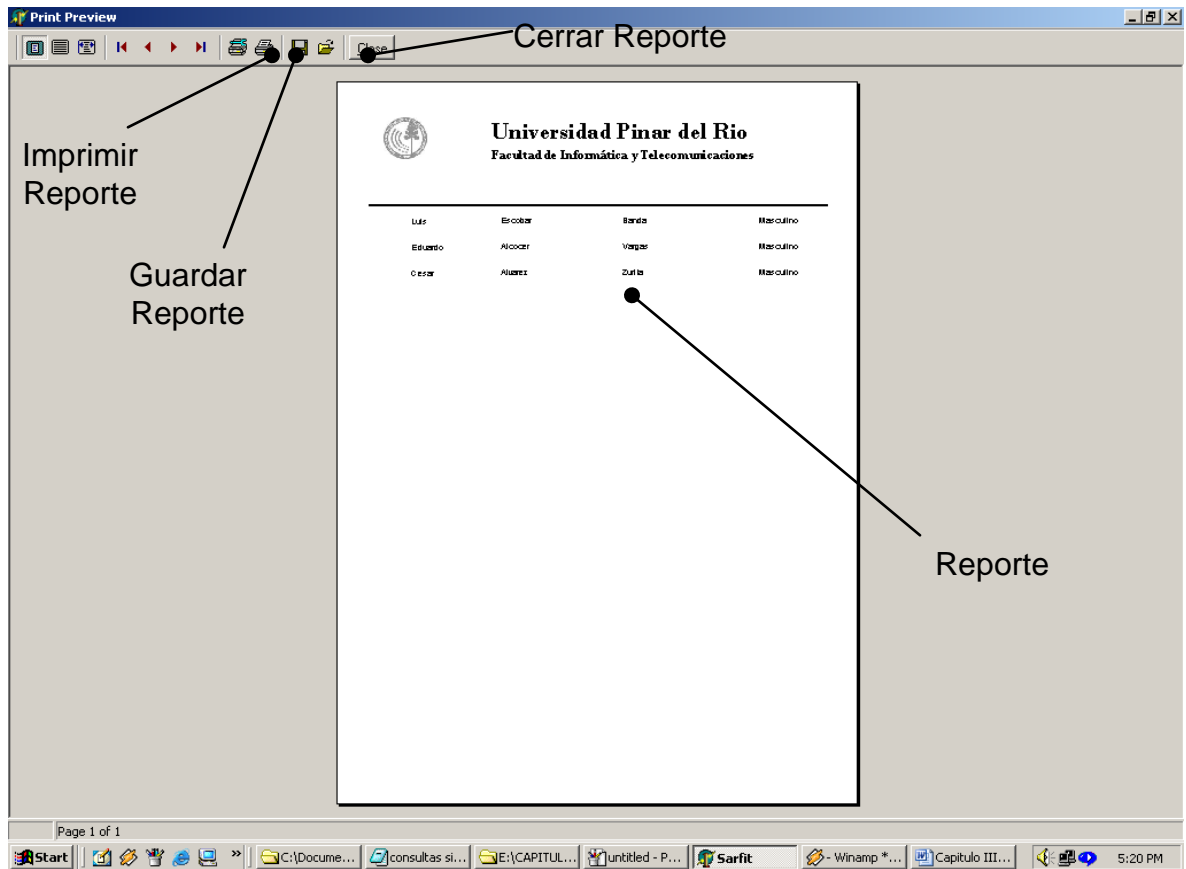
Nueva Lista    Imprimir    Salir

Datos a buscar

Lista obtenida

Botones

*Figura III.19. Lista de Estudiantes*



*Figura III.20. Reportes de Estudiantes (Carrera)*

**Lista de Profesores**, realiza la búsqueda de los datos de los profesores de la Facultad por departamento o todos en general. Como el formulario anterior también emite reportes que pueden o no ser impresos por el usuario. (Ver Fig. III.21)



*Figura III.21. Reportes de Profesores (por Departamento)*

**Búsqueda Datos personales**, Nos permite realizar una búsqueda tanto de Profesores o Estudiantes mediante su Identificación, su Nombre o su Apellido, además se valida para que tenga ingresado algún dato para la búsqueda caso contrario no realizará la búsqueda y saldrá un mensaje (ver Fig. III.22)



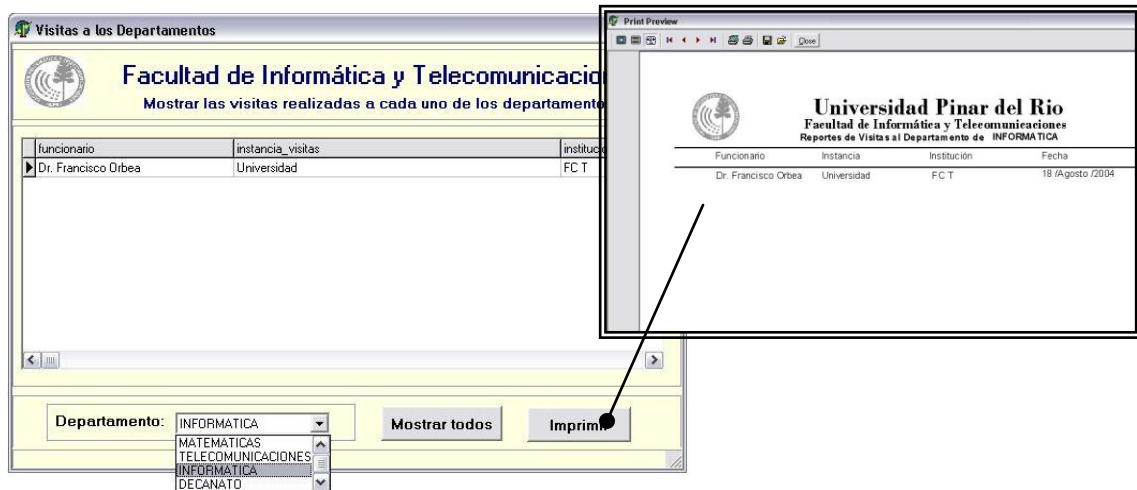
Figura III.22. Búsqueda por Datos personales. (Profesores/Estudiantes)

**Formulario Mostrar Premios.** Mostrará los datos de las personas que han obtenido premios. Debemos seleccionar las opciones Profesor o Estudiante, y como realizaremos la búsqueda por la identificación o por la fecha del premio, además podemos desplegar todos los premios que se han obtenido. (Ver Fig.:III.23)



Figura III.23. Consultas de Premios

**Formulario Mostrar Visitas.** Mostrará las visitas que se han realizado a todos los departamentos de la facultad, o a uno en particular dependiendo de la opción que escoja el usuario. Permite además generar reportes. (Ver Fig.:24)



*Figura III.24. Consultas de Visitas a los Departamentos de la FIT.*



## CONCLUSIONES

- Se identificó con el concurso de los directivos de la FIT qué información es relevante para la organización y por lo tanto necesaria su conservación tanto para la toma de decisiones como para el registro del historial de la facultad,
- Se revisó en copias duras y en formatos magnéticos la información almacenada sobre el desempeño de profesores y estudiantes de la FIT en los cuatro cursos de su funcionamiento, evidenciándose falta de homogeneidad y exactitud en los datos.
- Se diseñó una base de datos única que almacena adecuadamente los datos relevantes del desempeño de la FIT.
- Se diseñó e implementó un software *SAIRFIT* que gestiona la base de datos de manera eficiente.
- Se consultó una amplia bibliografía actualizada y se generó un documento con aspectos teóricos vinculados a la ingeniería de sistemas útil para la consulta del pre y postgrados.

## **RECOMENDACIONES**

- Hacer el instalador.
- Incorporar módulos para ser utilizados en las carreras presentes en la universalización.
- Aplicarlo en todas la facultades.
- Realizar las pruebas necesarias que permitan comprobar la eficiencia de la operatividad del sistema.

## REFERENCIAS BIBLIOGRÁFICAS

<sup>1</sup> **Estrategia Maestra Principal de la FIT 2003-2007**

<sup>2</sup> **Boar, B.H. “Application Prototyping : a requerments definition strategy for the 80s” Jhon Wiley & Sons, 1984.**

<sup>3</sup> **Connell, J.L y Brice, L. “Structured Rapid Prototyping” Prentice Hall, 1989**

<sup>4</sup> **Álvarez Cárdenas, Sofía. “METVISUAL E”,1999**

<sup>5</sup> **Computerworld, 1998**



## **BIBLIOGRAFÍA**

[1] Álvarez Cárdenas, Sofía. “METVISUAL E”(1999)

[2] Camacho, María Rosa. REVISTA CUBANA DE INFORMÁTICA MÉDICA “ARTÍCULOS ORIGINALES”, (2000).

[http://www.cecam.sld.cu/pages/rcim/revista\\_1/articulos\\_pdf/r0100a01.pdf](http://www.cecam.sld.cu/pages/rcim/revista_1/articulos_pdf/r0100a01.pdf)

[3] INGENIERÍA DE SOFTWARE – Monografías,

<http://www.monografias.com/trabajos5/inso/inso2.shtml>

[4] METODOLOGIA PARA LA ELABORACION DE UN PLAN DE SISTEMAS DE INFORMACIÓN / DIAGRAMA DE FLUJO DE DATOS, Instituto Nacional de Estadística e Informática, 1997

<http://www.inei.gob.pe/web/metodologias/attach/lib606/LIB606.htm>INEI

[5] García, Jaime Giraldo. LÍNEA DE PROFUNDIZACIÓN I EN SISTEMAS DE INFORMACIÓN PARA LAS ORGANIZACIONES, Facultad Ingeniería y Arquitectura Universidad Nacional de Colombia, 2005

**<http://www.virtual.unal.edu.co/cursos/sedes/manizales/4100010/Lecciones/Cap4/DFDPasoAPaso.htm>**

[6] Presuman, S Roger. FOLLETO DE INGENIERÍA DEL SOFTWARE, “UN ENFOQUE PRÁCTICO”, p. 23. 1999

[7] Castro, Jesús Antonio. BASE DE DATOS I, Unidad II: MODELO ENTIDAD-RELACIÓN, Instituto Tecnológico de La Paz,

[http://www.itlp.edu.mx/publica/tutoriales/basedat2/hdos2\\_1.htm](http://www.itlp.edu.mx/publica/tutoriales/basedat2/hdos2_1.htm)

[8] Marqués, María Mercedes APUNTES DE FICHEROS Y BASES DE DATOS, 2001-02-12, <http://www3.uji.es/~mmarques/f47/apun/node83.html>

[9] Herrera, Alvaro. EL MODELO ENTIDAD-RELACIÓN, 2004

<http://alvherre.atentus.cl/modBasico/node3.html>

[10] COMPUTACIÓN AVANZADA Y SISTEMAS EMPRESARIALES S.A. DE C.V.

[11] BASE DE DATOS, Wikipedia. la enciclopedia libre. 2005

[http://es.wikipedia.org/wiki/Sistema\\_de\\_gesti%C3%B3n\\_de\\_base\\_de\\_datos](http://es.wikipedia.org/wiki/Sistema_de_gesti%C3%B3n_de_base_de_datos)

[12] MICROSOFT\_ACCESS, Wikipedia.2005

[http://es.wikipedia.org/wiki/Microsoft\\_Access](http://es.wikipedia.org/wiki/Microsoft_Access).

[13] MANUALES OFFICE ACCESS, Solodrivrs 1998.

[http://ayudatecnica.solodrivrs.com/manuales\\_office\\_access.htm](http://ayudatecnica.solodrivrs.com/manuales_office_access.htm)

[14] Arsys Internet S.L 2004. <http://www.arsys.es/soporte/programacion/sql.htm>

[15] Marqués, María Mercedes. NORMALIZACIÓN, 2001

<http://www3.uji.es/~mmarques/f47/apun/node90.html>

[16] Gallo, Claudio. INTRODUCCIÓN A DELPHI , 2003

[http://www.arrakis.es/~ppriego/delphi/que\\_es.htm](http://www.arrakis.es/~ppriego/delphi/que_es.htm)

[17] LENGUAJE DE PROGRAMACIÓN DELPHI, Wikipedia La Enciclopedia Libre.

2005

[http://es.wikipedia.org/wiki/Lenguaje\\_de\\_programaci%C3%B3n\\_Delphi](http://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n_Delphi)

[18] López, Mario Carballo. EL AMBIENTE DE DESARROLLO INTEGRADO (IDE),

[http://www.itlp.edu.mx/publica/revistas/revista\\_isc/anteriores/mar98/delphi02.htm](http://www.itlp.edu.mx/publica/revistas/revista_isc/anteriores/mar98/delphi02.htm)

[19] Pons, Iván. COMPONENTES DE ACCESO DE DATOS DE DELPHI. Club Delphi - El punto de encuentro de los programadores en Delphi.htm,

[20] ADO y DELPHI. Copyright High Tech Learning Solutions, 2005.

<http://www.learningsolutions.com.mx/delphi3.cfm>

[21] Martínez, Mario. COMPONENTES ADO EN DELPHI.

<http://www.fortunecity.com/skyscraper/rofl/434/articulos/ene01/02010101.htm>

[22] QREPORT BORLAND DELPHI, Tutorial Borland Delphi de Programación,

<http://www.programacionfacil.com/delphi/trece4.htm>

[23] REPORTE DELPHI, Tutorial Borland DELPHI de programación,

<http://www.programacionfacil.com/delphi/catorce4.htm>

[24] <http://www.programacionfacil.com/delphi/indice.htm>