

**Desarrollo de un Software en Entorno Web
para el Manejo del Departamento Académico
en la Fundación para el Desarrollo de la Investigación
(FDI)**

Carlos Julio Nova Peñaranda

Adriana Patricia Arzayus Ecobar

Alberto Porto Danado

Trabajo presentado en la Asignatura de Investigación Formativa

IV

Universidad Simón Bolívar

Facultad

Ingeniería de Sistemas

Barranquilla



TABLA DE CONTENIDO

| | Pag |
|----------------------------------|-----|
| INTRODUCCIÓN | 1 |
| 1 PLANTEAMIENTO DEL PROBLEMA | 2 |
| 1.1 DESCRIPCIÓN DEL PROBLEMA | 2 |
| 1.2 FORMULACIÓN DEL PROBLEMA | 2 |
| 1.3 SISTEMATIZACIÓN DEL PROBLEMA | 2 |
| 2 OBJETIVOS | 3 |
| 2.1 OBJETIVO GENERAL | 3 |
| 2.2 OBJETIVOS ESPECÍFICOS | 3 |
| 3 JUSTIFICACIÓN DEL PROYECTO | 4 |
| 4 MARCO DE REFERENCIA | 6 |
| 4.1 MARCO TEÓRICO | 6 |
| 4.2 MARCO CONCEPTUAL | 13 |
| 4.3 MARCO LEGAL | 14 |
| 5 HIPÓTESIS | 18 |
| 6 METODOLOGÍA | 19 |
| 6.1 TIPO DE ESTUDIO | 19 |
| 6.2 POBLACIÓN Y MUESTRA | 19 |
| 6.2.1 POBLACION | 19 |
| 6.2.2 MUESTRA | 19 |
| 6.3 RECOLECCIÓN DE INFORMACIÓN | 19 |
| 7 RECURSOS | 20 |
| 7.1 RECURSOS HUMANOS | 20 |
| 7.2 RECURSOS ECONÓMICO | 20 |

| | |
|--|-----------|
| 8 CRONOGRAMAS | 22 |
| 9 INGENIERIA DE REQUISITOS | 23 |
| 9.1 DESCRIPCIÓN DEL SISTEMA ACTUAL | 23 |
| 9.2 MODELADOR DE PROCESOS ACTUAL | 25 |
| 9.3 IDENTIFICACIÓN DE REQUISITOS | 26 |
| 9.4 ANALISIS DE REQUISITOS | 26 |
| 9.5 ESPECIFICACIÓN DE REQUISITOS | 28 |
| 10 INGENIERIA DE INFORMACION | 29 |
| 10.1 MISION | 29 |
| 10.2 VISION | 29 |
| 11 ANALISIS DEL SISTEMA | 30 |
| 11.1 DIAGRAMA DE FLUJOS DE DATOS(CONTEXTO) | 30 |
| 11.1.1 DIAGRAMA DE FLUJOS DE DATOS(1er NIVEL) | 31 |
| 11.1.2 DIAGRAMA DE FLUJOS DE DATOS(2do NIVEL) | 32 |
| 11.2 MAPA DE NAVEGACION | 33 |
| 11.3 MODELO ENTIDAD RELACION | 34 |
| 11.4 MODELO RELACIONAL | 35 |
| 11.5 DICCIONARIO DE DATOS | 36 |
| 11.6 ANÁLISIS DE CONTENIDO | 43 |
| 11.7 ANÁLISIS FUNCIONAL | 43 |
| 11.8 ANÁLISIS DE ITERACIÓN | 46 |
| 11.9 ESTANDARIZACIÓN DE TABLAS | 47 |
| BIBLIOGRAFÍA | 53 |

INTRODUCCION

La fundación para el desarrollo de la investigación FDI quiere ser la mejor en todos y cada uno de los aspectos, es por esto que se quiere superar todas aquellas partes donde se encuentran fallas como es el departamento académico que no tiene debidamente organizada la información y no cuenta con una aplicación que satisfaga todas sus necesidades.

Es por esto que se procede a desarrollar un software capaz de cumplir con todas aquellas necesidades que presentan en estos momentos la fundación para que con esta herramienta se trabaje de la mejor forma y no se les presente ningún contratiempo. Además, cuando se requiera un reporte de notas se realice de una manera rápida y eficiente.

Este proyecto pretende que el FDI supere todas aquellas fallas que presenta, hacer que la fundación reduzca sus costos lo que para nosotros significará aplicar todos los conocimientos adquiridos durante el transcurso de nuestra carrera profesional.

1. Planteamiento del problema

1.1 Descripción del problema

La fundación para el desarrollo de la investigación FDI es una entidad académica dedicada a la enseñanza de ciencias forenses y criminalísticas, esta a presentado un notable crecimiento durante los últimos años. Debido a esto, la información que maneja ha aumentado considerablemente haciéndose muy complicado su procesamiento. Una de las áreas donde se presentan irregularidades es el departamento académico

El departamento académico tiene sus deficiencias, las cuales son: se está utilizando una aplicación que no cumple con todos los requerimientos del FDI. La causa que produce este problema es el tiempo de respuesta para el usuario es muy lento y por dicho razón pierde mucho tiempo a la hora de hacer una consulta. Además, los estudiantes no cuentan con un sistema para consultar sus notas. Debido a esto tienen que acercarse a algún funcionario del departamento para conocer sus calificaciones.

Tienen dificultad con el manejo de notas por parte de los profesores, ya que estas se pasan a una hoja de cálculo y la aplicación que se usa, no está diseñada específicamente para tal tarea. La causa de este problema es que tiene que digitar las notas de cada estudiante y estos archivos no se encuentran debidamente organizados trayendo como consecuencia que el coordinador le asigne por error la nota que no es a un estudiante y por esto su margen de error es muy alto al momento de digitar las notas, además se tarda mucho para que el estudiante conozca sus notas. Sin embargo, no cuentan con un sistema que permita asignar las diversas materias a los profesores y alumnos; debido a esta situación se hace muy complicado asignar las materias.

1.2 Formulación del problema.

¿Cómo facilitar el manejo de los procesos académico con el fin de mejorar la administración de la información en la fundación para el desarrollo de la investigación (FDI)?

1.3. Sistematización del problema.

¿Cómo se maneja la información del departamento de académico?

¿Podrá el departamento académico llevar un mejor control en la organización de las notas de los estudiantes?

¿que manera el proceso actual está afectando el manejo de los recursos?

2 OBJETIVOS

2.1 Objetivos General

Diseñar e implementar una aplicación en entorno web, para que el departamento académico pueda manejar de una forma eficiente y confiable la información de los estudiantes en el FDI

2.2 Objetivos Específicos

- Identificar los controles que se llevan en el registro de las notas, datos personales de los estudiantes, entre otros.
- Suministrar una herramienta que pueda hacer un mejor uso de los recursos que tiene disponible el FDI, para la realización de sus funciones.

3. JUSTIFICACIÓN DEL PROYECTO

El diseño propuesto busca mediante la aplicación de los conceptos básicos el manejo del departamentos académico para encontrar explicaciones a situaciones internas (consulta de notas, manejo de datos etc) que se realizan en la institución. Lo anterior permitirá a los diseñadores contrastar diferentes conceptos en una realidad como el FDI.

El proyecto es de total viabilidad ya que se piensan usar herramientas de software no licenciadas, lo que resulta bastante atractivo para la institución porque lo que se persigue son soluciones óptimas a un bajo costo. Además el desarrollo en entorno web es realmente una propuesta bien pensada para posibles situaciones futuras ya que todo lo que tiene que ver con la informática y la tecnología está moviéndose de manera rápida y constante a las aplicaciones en Internet, esto se puede decir con total confianza sabiendo las intenciones que tiene el FDI con respecto a crear su propio sitio informático, subirlo en la red y de esta forma poder acomodar toda su información (ya sea corporativa y pública).

Con este proyecto se quiere satisfacer todas aquellas necesidades que se vienen presentando en el FDI como: datos personales de alumnos y profesores, notas de alumnos, entre otros. Este diseño no sólo les va permitir organizar y manejar de forma apropiada la información correspondiente a el departamento académico sino que va ha aprovechar todo la tecnología de información para plantear estrategia que le ayude a tomar decisiones precisas y concretas, en el departamento académico y hay la necesidad de organizar la información de profesores y estudiantes para sacarles el mayor provecho posible como: formar al alumno con un alto nivel académico para cuando salga del FDI y se emplee cumpliendo con todas las funciones designadas realizándolas con el mayor profesionalismo posible y por ultimo también permitirá evaluar a cuerpo docente para ver si cumple con los requisitos necesarios para desempeñarse como maestro.

Con las operaciones anteriormente mencionadas se puede hacer que el FDI crezca como una entidad educativa en donde la demanda de las personas que quieren estudiar criminalistica o forense se dispare y el FDI aumente su oferta con la mayor calidad posible que satisfaga todas las necesidades de los estudiantes, permitiéndoles crecer para traspasar las fronteras por medio de la web y en un futuro no solamente se dedique a el campo de la enseñanza sino incursionen en otras áreas a fines que traigan consigo un

aumento de capital notorio y un manejo de información excelente, Para lograr el cumplimiento de los objetivos del diseño se acude al empleo de técnicas de recopilación de información (como entrevista a las personas del departamento académico) para así dar con las deficiencias que tienen estas dos áreas y continuar con los siguientes pasos del diseño de la aplicación.

4. MARCO DE REFERENCIA.

4.1 Marco teórico

Muchas universidades y casas dedicadas a la creación de software han elaborado muchas aplicaciones relacionadas con el manejo del departamento académico en una aplicación, un ejemplo de este tipo de programa son:

* SIRCOA Software para Registro y Control Académico

”A través de SIRCOA los centros educativos administran todos los procesos involucrados con el registro y control académico de los estudiantes, inscripciones y admisiones, control y registro de notas, *traslados, transferencias, matriculas, materias y salones, gestión de créditos y gestión administrativa arrojando reportes y estadísticas que le permitirán conocer el estado financiero de la entidad. SIRCOA es un software abierto, modular y escalable diseñado con herramientas completamente independientes de hardware, sistema operativo y motor de base de datos. Adicionalmente cuenta con soluciones a través de Internet que permiten agilizar los procesos masivos como inscripción de estudiantes y generación de aprobaciones entre otros”*

*SINU

”Es la herramienta de apoyo a la gestión de custodia y certificación de la información generada en las diferentes áreas como resultado de los procesos académicos y administrativos de las instituciones de educación superior.

Concebido para apoyar los requerimientos del proceso de ACREDITACION Y AUTO EVALUACION INSTITUCIONAL cuyo objetivo es ser la herramienta de apoyo y control de los procesos académicos y administrativos de las entidades de educación superior para satisfacer todas las necesidades de esta.

Esta aplicación apoya las funciones de la institución en lo relacionado con el manejo de la información de aspirantes, alumnos, egresados, docentes, recursos físicos y también sirve de control a la función administración llevando un registro detallado de las actividades financieras, contables, presupuestales y de manejo de recursos físicos que se desprenden de la actividad académica.

* <http://www.catalogodesoftware.com>).

Este sistema presenta las siguientes características: es totalmente parametrizable, modular y en ambiente gráfico, enfocado a facilitar medios de control de la información y automatización de los procesos operativos. La integración de los módulos garantiza un registro único y la definición de comprobantes automáticos, simplificando la administración y apoyando la toma de decisiones. El diseño permite la fácil adaptación a cualquier entidad, integrado por los siguientes módulos: Inscripciones, Admisiones, Gestión de Planes de Estudios, Historia Académica, Matricula (Académica y Financiera), Gestión de Notas,

Grados y Egresados, Recursos Docentes y Físicos, Consultorio Jurídico, Cartera

Financiera y Gestión WEB. Los módulos generan la información para las diferentes unidades y la requerida por los organismos de control, apoyan el acopio de información en el proceso de acreditación y auto evaluación de la institución. El sistema es parte de SINU (SISTEMA DE INFORMACION UNIVERSITARIA) al integrarse con el Sistema Administrativo Financiero SAF - ERP (Contabilidad General, Presupuesto, Cuentas por pagar y Cuentas por cobrar, Cartera, Tesorería, Bancos, Cotizaciones, Compras, Contratos, Inventarios, Activos fijos, Facturación, Tributario, Personal y Nomina.)^{*}

Debido a la gran importancia que tiene la información en el departamento académico. **La fundación para el desarrollo de la investigación (FDI)**, organización dedicada a enseñar todo lo relacionado con la ciencias forenses, investigación y criminalística ha determinado implantar una aplicación que involucre a este departamento vital para el manejo de información del FDI ya que por falta de automatización de procesos hay una pérdida de recursos bastante considerables.

El mal manejo de flujo de información en el departamento académico hace que se presenten fallas que pueden afectar la fiabilidad de las notas, la adecuada utilización de los recursos y sobre todo la agilidad en la realización de cada uno de los procesos. Por dicha razón el trabajo que se realice en esta área es vital para cualquier institución académica porque nos permite realizar actividades como: saber cuantos estudiantes hay en la institución, como es el desarrollo de cada uno de los estudiantes por medio

^{*} <http://www.catalogodesoftware.com>).

*de exámenes que se realizaran duran el transcurso del semestre para evaluar los conocimientos adquiridos en ese periodo , capacitar al cuerpo docente para que cumpla con todas las metas trazadas según lo dice la ley “la formación de educadores tendrá como fines generales:

- Formar un educador de la mas alta calidad científica y ética
- Desarrollar la teoría y la practica pedagógica como parte fundamental del saber del educador.
- Fortalecer la investigación en le campo pedagógico y en el saber específico.
- Preparar a los educadores a nivel de pregrado y de postgrado para los diferentes niveles y formas de presentación del servicio”*. además organizar los diferentes pensum para los diversos programas analizando cuales son las materias mas acorde y esenciales para la formación de un profesional ,almacenar todos los datos personales y notas académicas de cada uno de los estudiantes, la realización de matricula que de acuerdo a la ley nos dice “es el acto que formaliza la vinculación del educando al servicio educativo. Se realizara por una sola vez, al ingresar el alumno a un establecimiento educativo, pudiéndose establecer renovaciones para cada periodo”* ,inscripciones ,admisiones y la elaboración de constancias de estudio y certificados de notas.

Las irregularidades que presenta el FDI en esta área no la dejan crecer y presentarse como la mejor escuela especializada en su campo ,es por esto que es necesario aplicar la tecnología para optimizar la información y esto se va a realizar partiendo de un modelo lineal secuencial que permite ir realizando paso a paso el desarrollo de software de manera que cuando comienza el desarrollo de una etapa no se avanza a otro paso sin haber terminado totalmente el trabajo de la etapa anterior.

El modelo lineal secuencial consta de seis etapas muy importantes que son: Ingeniería y Modulado del Sistema \Información, Análisis de los requisitos del software, Diseño, Generación de código , Prueba y Mantenimiento.

Uno de los propósitos de FDI es tener una aplicación completa y eficiente para el departamento académico , para la realización de esto se deben unir muchos componentes por medio de la **Ingeniería y**

* ley general de la educación 115

* ley general de la educación 115

modulado del sistema \información que consiste en "como el software siempre forma parte de un sistema mas grande (o empresas), el trabajo comienza estableciendo requisitos de todos los elementos del sistema asignando al software un subgrupo de estos requisitos. Esta visión de sistemas es esencial cuando el software se debe interconectar con otros elementos: como el hardware, usuarios, base de datos . La ingeniería de información abarca los requisitos que se recogen en el nivel del sistema, en el nivel de la empresa estratégico y en el nivel de área de negocio con una pequeña parte de análisis y diseño."7(este es la numeración de *, para llevar a cabo esta función es necesario que tengamos un conocimiento amplio de cómo funciona este departamento en el FDI mediante el **Análisis de los requisitos de software** que tiene como objeto principal hacer que "conocer como funcionan actualmente los procesos, para así poder visualizar cuales son las diferentes ventajas y desventajas que presenta el proceso, teniendo estos puntos claros podemos saber cuales son los campos a sistematizar y además podemos recolectar datos necesarios para la realización del trabajo"* , después de conocer como trabaja el FDI hay que estructurar y organizar los datos obtenidos a través del **Diseño** que se basa en "la realización de muchos pasos que se centran en cuatro atributos distintos de programa: estructura de datos, arquitectura del software, representación de interfaz y el detalle procedimental"* ,la finalidad del diseño es estructurar los datos de forma eficaz que permita un manejo adecuado de la información que se recolecto, ya que se organizo la información es importante pasarla a un código fuente para ver si el diseño corresponde con todos aquellos requisitos que plantea el FDI y esto se realiza mediante la **Generación del código** que es" el diseño se debe pasar a un lenguaje maquina para que la aplicación realice todas aquellas tareas especificadas por el programador, para que así, el software funcione de acuerdo con las necesidades del cliente"* , luego de realizar la generación de código tenemos que ver si la aplicación satisface todas la necesidades de FDI mediante la **Prueba** que es " Una vez que se ha generado el código, comienzan las pruebas del programador. El proceso de prueba se centra en los proceso lógicos internos del software, asegurando que todas las sentencias se han comprobado, y en los procesos externos funcionales; es decir realizar las pruebas de detección de errores y asegurar que la entrada definida produce resultados reales de acuerdo con los resultados requeridos." Y

* ingeniería de software
* ingeniería de software
* ingeniería de software
* ingeniería de software

por ultimo el programa que se realizara debe ser flexible para pueda satisfacer todos los requisitos futuros del FDI a través del **Mantenimiento** que se fundamenta en “ los cambios que sufrirá el software después de ser entregado al cliente. Estos cambios se producirán debido a que se encontrarán errores, por que el software debe adaptarse para complacer los cambios de su entorno externo o por que el cliente requiere mejoras funcionales o de rendimiento. El soporte y el mantenimiento del software vuelve a aplicar cada una de las fases precedentes a un programa ya existente, y no a uno nuevo”.

Conociendo todos aquellos pasos que son fundamentales para la realización de un programa es necesario que el producto entregado sea totalmente comprendido por los usuarios finales, por tal razón debemos identificar ciertos aspectos como:

- ¿Quién es el usuario?
- ¿Cómo aprende el usuario a interactuar con el nuevo sistema ?
- ¿Cómo interpreta el usuario la información la información producida?
- ¿Qué espera el usuario?

Para construir un interfaz de usuario eficiente tenemos que conocer su edad, sexo, capacidades físicas, estudios, historial cultural y étnico, etc, conociendo las características anteriormente mencionadas pasamos a clasificar a los usuarios de acuerdo a su conocimiento y a las capacidades de interactuar con el computador, los diferentes tipos de usuarios son:

*Usuarios novatos: poco conocimiento del computador y de la aplicación un, ejemplo de esta clase de usuarios en el FDI es la secretaria académica que tiene muy limitados conocimientos acerca de las aplicaciones que se manejan en el FDI .

*Usuarios esporádico: conocimiento de la aplicación pero la recuerdan vagamente, un ejemplo de este tipo de usuario es el director de FDI que a veces se acerca al departamento académico y revisa las notas de los estudiantes pero debido a que su contacto con la aplicación es poco no tiene mucha destreza en el manejo de esta .

*Usuarios expertos: muy buenos conocimientos de la computadora y de aplicación, un ejemplo de esta clase de usuarios es el coordinador académico .

Teniendo claro cuales son los tipos de usuarios que podemos encontrar en el FDI debemos establecer como aprenderá el usuario a interactuar



con el nuevo, este aprendizaje se realizara con varios componentes que son:

***Dialogo:** son mensajes, inducciones y respuestas utilizadas para llevar a cabo una conversación entre el sistema y el usuario, que determina la información a introducir y la forma en que se hacen la respuestas, por ejemplo cuando el coordinador académico va a imprimir los horarios de clase de los alumno y la impresora no tiene papel esta envía un mensaje indicándole a usuario que tiene que colocar las hojas para poder imprimir la información requerida.

***Entrada:** es la información proporcionada por lo usuarios par solicitar una acción que inicie una respuesta por el sistema, un ejemplo de este componente es cuando el coordinador académico ingresa las notas de los estudiantes en Excel para sacarles un promedio de las calificaciones que tienen los estudiantes.

***Respuesta:** un mensaje de procesamiento que se resultado de una entrada proporcionada por el usuario , basándonos en el ejemplo anterior la respuesta seria el promedio de los estudiante.

Para que el usuario introduzca la información y obtenga las repuestas necesarias a sus requerimientos es indispensables contar con una serie de dispositivos como lo son: el teclado, ratón, escáner, impresora, entre otros, pero no solamente hay que tener estos dispositivos sino que el usuario sepa interpretar las respuesta que le da la aplicación a sus diferentes cuestionamientos. Es por esto que los mensajes que se le van a mostrar al usuario deben ser sencillos y fáciles de comprender para que lo usuarios finales tenga claro que hacer al momento de tomar un decisión, saber el estado de ciertos eventos y actividades, cuando se ha terminado un tarea, cuando ocurren errores y como solucionarlos.

En la interfaz de usuario no solamente es importante que los usuarios entiendan completamente la aplicación sino que este cumpla con todos los requerimientos solicitados y que su interfase sea agradable(ya sea grafica, ambiente texto o menús anidados) ,para así cumplir con todas las expectativas de los usuarios finales.

El FDI no solo quiere un interfaz agradable sino traspasar fronteras y para realizar esto se va ser uso de PHP (Hiper Text Processor) que es un lenguaje de programación de estilo clásico, con esto quiero decir que es un lenguaje de programación con variables, sentencias

condicionales, bucles, funciones.... No es un lenguaje de marcas como podría ser HTML, XML o WML. Está mas cercano a JavaScript o a C, para aquellos que conocen estos lenguajes.

Pero a diferencia de Java o JavaScript que se ejecutan en el navegador, PHP se ejecuta en el servidor, por eso nos permite acceder a los recursos que tenga el servidor como por ejemplo podría ser una base de datos. El programa PHP es ejecutado en el servidor y el resultado enviado al navegador. El resultado es normalmente una página HTML pero igualmente podría ser una pagina WML.

Al ser PHP un lenguaje que se ejecuta en el servidor no es necesario que su navegador lo soporte, es independiente del navegador, pero sin embargo para que sus páginas PHP funcionen, el servidor donde están alojadas debe soportar PHP.

Para concluir debemos de tener en cuenta todas las teorías planteadas para poder realizar una aplicación que cumpla con los requerimientos establecidos por el FDI.

4.2 MARCO CONCEPTUAL

- * Datos: Son los valores que almacenados en el computador podrán procesarse de manera que puedan ser gestionados de forma mas eficiente.
- * Departamento académico: Es el área encargada de llevar el control de las notas de los estudiantes, así mismo debe controlar las asignaturas vistas por cada alumno de modo que se puedan hacer proyecciones con esta información.
- * Diagrama de flujo de datos: Es una técnica gráfica que describe el movimiento de la información y las transformaciones que se aplican a los datos conforme se mueven de la entrada a la salida.
- * Diagrama de contexto: Es aquel que contiene el proceso como un todo, se usa para definir el sistema que se va a estudiar de modo que se puedan determinar fronteras .
- * Diccionario de datos: Es la gramática casi formal para describir el contenido de los elementos de información mencionados en el diagrama de flujo de datos.
- * Documentación: Son los manuales impresos y otra información descriptiva que explica el uso y/o la operación del sistema.
- * Entidades: También llamadas tablas, es la forma organizada de guardar la información de manera que en la base de datos no exista redundancia y se guarde integridad para poder llevar un correcto manejo de esta.
- * Fuente o destino de datos externos: Son todos los datos que pueden provenir o dirigirse a personas, programas, organizaciones u otras entidades que interactúan con el sistema.
- * Flujo de datos: Movimiento de los datos en determinada dirección (ya sea en documentos, cartas, llamadas telefónicas, etc...)
- * Hardware: Es la parte física sobre la cual se montan las herramientas de software, desde aquí es donde el usuario le indica al computador las instrucciones necesarias para los resultados deseados.
- * Información: Es la compresión de los datos suministrados.

- * Informática: Es la rama de la tecnología que se encarga de la información.
- * Javascript: Es un lenguaje script u orientado a documento, como pueden ser los lenguajes de macros que tienen muchos procesadores de texto.
- * Modelo entidad-relación: Consiste en plantear de forma gráfica el funcionamiento de la organización sobre la cual se va a desarrollar la aplicación. Este modelo consta de entidades y relaciones.
- * Procesamiento: Son los pasos que se definen para el uso específico de cada dato.
- * Relaciones: Son los campos en común que van a tener las tablas para poder comunicarse entre sí.
- * Script: líneas de código.
- * Sistemas de información: Es un conjunto de elementos que permiten al computador almacenar y procesar datos para luego convertirlos en información de acuerdo con las instrucciones definidas en el software.
- * Software: Es el elemento lógico (no físico) del sistema, que por medio de un conjunto de instrucciones permite realizar diversas tareas.

4.3 MARCO LEGAL:

Para el presente proyecto se utilizará las siguientes leyes que respaldan lo que se va a realizar mas adelante.

Para el preente proyecto se utilizará las siguientes leyes que respaldan lo que se va a realizar mas adelante.

* Departamento académico:

Para el departamento académico le detallaremos la ley que respalda la educación "Ley general de la educación: LEY 115".

la Ley 115 nos dice en el artículo 1°, que la educación es un proceso de formación permanente, personal, cultural y social que se fundamenta en una concepción integral de la persona humana, de su dignidad, de sus derechos y sus deberes.

La presente ley señala las normas generales para regular el servicio público de la educación que cumple una función social acorde a las necesidades e intereses de las personas, de las familia y de la sociedad. Se fundamenta en los principios de la constitución política sobre el derecho a la educación que tiene una persona, en las libertades de enseñanza, aprendizaje, investigación y cátedra y en su carácter de servicio público.

De conformidad con el artículo 67 de la constitución política y desarrolla la organización y la prestación de la educación formal en sus niveles preescolar, básica (primaria y secundaria) y media, no formal e informal, dirigida a niños y jóvenes en edad escolar, a adultos, a campesinos, a grupos étnicos, a personas con limitaciones físicas, sensoriales y psíquicas, con capacidades excepcionales, y a personas que requieran rehabilitación social.

La educación superior es regulada por la ley especial excepto lo dispuesto en la presente ley.

Artículo 33. Norma general.

1. No podrán realizarse transferencias temporales ni definitivas de datos de carácter personal que hayan sido objeto de tratamiento o hayan sido recogidos para someterlos a dicho tratamiento con destino a países que no proporcionen un nivel de protección equiparable al que presta la presente Ley, salvo que, además de haberse observado lo dispuesto en ésta, se obtenga autorización previa del Director de la Agencia de Protección de Datos, que sólo podrá otorgarla si se obtienen garantías adecuadas.

2. El carácter adecuado del nivel de protección que ofrece el país de destino se evaluará por la Agencia de Protección de Datos atendiendo a todas las circunstancias que concurran en la transferencia o categoría de transferencia de datos. En particular, se tomará en consideración la naturaleza de los datos, la finalidad y la duración del tratamiento o de los tratamientos previstos, el país de origen y el país de destino final, las normas de derecho, generales o sectoriales, vigentes en el país tercero de que se trate, el contenido de los informes de la Comisión de la Unión Europea, así como las normas profesionales y las medidas de seguridad en vigor en dichos países.

Artículo 34. Excepciones.

Lo dispuesto en el artículo anterior no será de aplicación:

- a) Cuando la transferencia internacional de datos de carácter personal resulte de la aplicación de tratados o convenios en los que sea parte España.
- b) Cuando la transferencia se haga a efectos de prestar o solicitar auxilio judicial internacional.
- c) Cuando la transferencia sea necesaria para la prevención o para el diagnóstico médicos, la prestación de asistencia sanitaria o tratamiento médicos o la gestión de servicios sanitarios.
- d) Cuando se refiera a transferencias dinerarias conforme a su legislación específica.
- e) Cuando el afectado haya dado su consentimiento inequívoco a la transferencia prevista.
- f) Cuando la transferencia sea necesaria para la ejecución de un contrato entre el afectado y el responsable del fichero o para la adopción de medidas precontractuales adoptadas a petición del afectado.
- g) Cuando la transferencia sea necesaria para la celebración o ejecución de un contrato celebrado o por celebrar, en interés del afectado, por el responsable del fichero y un tercero.
- h) Cuando la transferencia sea necesaria o legal-mente exigida para la salvaguarda de un interés público. Tendrá esta consideración la transferencia solicitada por una Administración fiscal o aduanera para el cumplimiento de sus competencias.
- i) Cuando la transferencia sea precisa para el reconocimiento, ejercicio o defensa de un derecho en un proceso judicial.
- j) Cuando la transferencia se efectúe, a petición de persona con interés legítimo, desde un Registro público y aquella sea acorde con la finalidad del mismo.
- k) Cuando la transferencia tenga como destino un Estado miembro de la Unión Europea, o un Estado respecto del cual la Comisión de las

Comunidades Europeas, en el ejercicio de sus competencias, haya declarado que garantiza un nivel de protección adecuado.

5. **HIPOTESIS**

La finalidad del FDI es seguir avanzando en el sector académico para presentarse como la mejor escuela especializada en su campo, es por esto que se hace necesario aplicar la tecnología para organizar la información (realizar consultas precisas y guardar datos con mayor seguridad) especialmente en el departamentos académico, ya que se han presentado retrasos que afectan de manera importante.

El desarrollo eficiente de las actividades rutinarias.

Se realizará una aplicación que permita atender satisfactoriamente todos estos requerimientos, ya que de esta manera existirá un mejor control de registros de los alumnos para que oportunamente obtengan valores reales del estado académico de los estudiantes.

Esta aplicación se realizará en entorno web (compatible a Internet) para que en un futuro todos estos módulos puedan ser mostrados en un sitio exclusivo para la entidad.

6. METODOLOGIA

El método de investigación a utilizar es el inductivo, porque con la información que se logre obtener del FDI (con respecto al departamento académico) se podrán realizar los objetivos que se ha estipulado para este proyecto.

6.1 Tipos de Investigación

El tipo de investigación a implementar es el descriptivo, ya que haciendo un análisis de los procesos se puede alcanzar una enfoque global de la estructura completa de la organización.

6.2 Población y Muestra

6.2.1 Población

La población que participará en este proyecto es el FDI, debido a que esta será la fuente de toda la información que tendrá lugar en la aplicación.

6.2.2 Muestra

La muestra que participará en este proyecto es el departamento y académico, debido a que estos serán los que directamente participarán en el desarrollo de esta aplicación.

6.3 Recolección de información

- ✓ **Fuentes primarias:** Las fuentes primarias utilizadas para la realización de este proyecto fueron las entrevistas realizadas al personal del departamento académico.
- ✓ **Fuentes secundarias:** se hicieron consultas desde Internet con el in de ampliar un poco mas el contexto .

7. RECURSOS

7.1 Recursos Humanos

Sergio Jimenez Ingeniero de sistemas
Darwin mercado Ingeniero de sistemas
Ivon Romero Ingeniero de sistemas

7.2 Recursos Financiero

| | |
|------------------------|------------------|
| Transporte | \$ 90.000 |
| Internet | \$ 50.000 |
| Transcripciones | \$100.000 |
| Otros | \$ 60.000 |
| Total | \$300.000 |



8. CRONOGRAMA

| Actividades | SEMESTRE | | | | | | | | | | | | | | | | | | | |
|--|----------|---|---|---|---|--------|---|---|---|---|--------|---|---|---|---|---------|---|---|---|---|
| | Octavo | | | | | Noveno | | | | | Décimo | | | | | Onceavo | | | | |
| | Meses | | | | | | | | | | | | | | | | | | | |
| | 1 | 2 | 3 | 4 | 5 | 1 | 2 | 3 | 4 | 5 | 1 | 2 | 3 | 4 | 5 | 1 | 2 | 3 | 4 | 5 |
| Búsqueda del Tema | | | | | | | | | | | | | | | | | | | | |
| Formulación del documento | | | | | | | | | | | | | | | | | | | | |
| Análisis y Diseño | | | | | | | | | | | | | | | | | | | | |
| Codificación de software | | | | | | | | | | | | | | | | | | | | |
| Puesta en marcha, corrección de errores y manuales de usuario. | | | | | | | | | | | | | | | | | | | | |

9. INGENIERIA DE REQUISITOS

9.1 Descripción del Sistema Actual. (Casos de Uso)

Casos de Uso

| | |
|---------------------|--|
| Caso de uso: | Matricula de estudiantes |
| Actores: | Coordinador académico |
| TIPO: | Primario y de alto nivel |
| Descripción: | El coordinador académico le entrega al estudiante un formato para que coloque todos sus datos personales y anexe todos los documentos necesarios para la gestión de matricula, luego el coordinador registra toda esa información en una aplicación cuyo nombre es Sigo. |

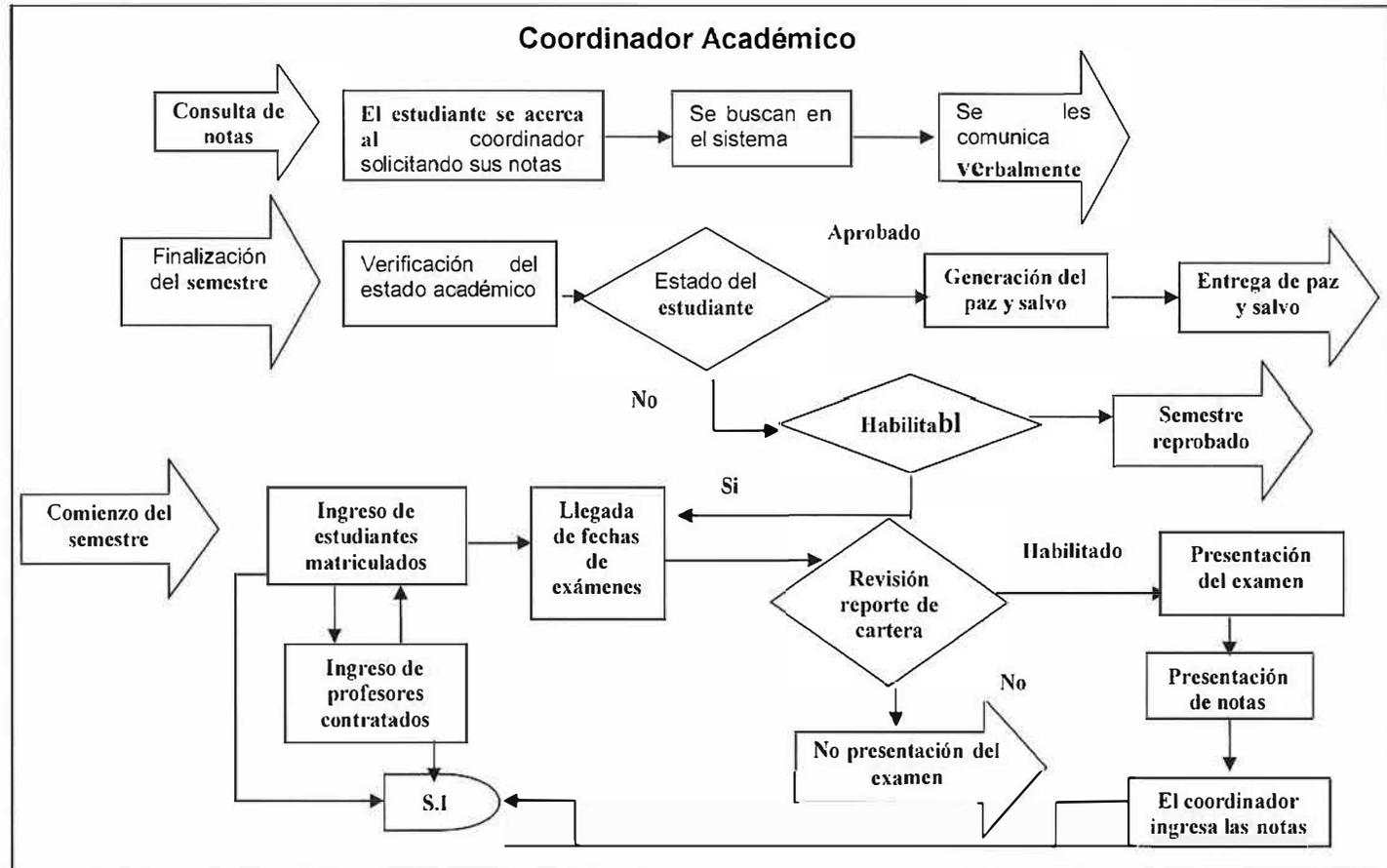
| | |
|---------------------|---|
| Caso de uso: | Entregar notas |
| Actores: | Profesor |
| TIPO: | Primario y de segundo nivel |
| Descripción: | El profesor entrega las notas al coordinador para su respectiva evaluación. |

| | |
|---------------------|--|
| Caso de uso: | Ingresar las notas a una hoja de calculo |
| Actores: | Coordinador académico |
| TIPO: | Primario y de alto nivel |
| Descripción: | El coordinador recibe las notas de los estudiantes de parte del profesor y las almacena en su respectiva hoja de calculo |

| | |
|---------------------|---|
| Caso de uso: | Publicar notas |
| Actores: | Coordinador académico |
| TIPO: | Primario y de alto nivel |
| Descripción: | El coordinador publica las notas para los estudiantes en cada examen parcial presentado |

| | |
|---------------------|--|
| Caso de uso: | Entrega de paz y salvo académico |
| Actores: | Estudiante |
| TIPO: | Secundario |
| Descripción: | Sólo se le entregará el paz y salvo a aquellos estudiantes que hallan cancelado el total de la matricula |

9.2 Modelador de Proceso Actual Dpto Académico



9.3 Identificación de los Requisitos (Departamento de académico).

Requisito Descripción de los requisitos funcionales

| | |
|------|---|
| RA1 | Desarrollar una aplicación fácil de manejar |
| RA2 | Desarrollar una aplicación que permita modificar los datos personales de los estudiantes |
| RA3 | Permitir ingresar los datos de los estudiantes y las notas en una base de datos |
| RA4 | Desarrollar un aplicación que permita al coordinador ingresar todas las notas de los estudiantes |
| RA5 | Generar un programa que le permita al estudiante consultar sus notas sin necesidad de dirigirse al coordinador académico |
| RA6 | Desarrollar un sistema que permita generar planillas ya sea por curso, materia, docente y periodo |
| RA7 | Desarrollar una aplicación que facilite las consultas de datos de estudiante a través del código |
| RA8 | Implantar un chat para mantener comunicados a los integrantes de cada uno de los departamentos |
| RA9 | Desarrollar un sistema que permita ingresar los datos personales de los profesores Desarrollar una aplicación que permita modificar los datos personales de los profesores |
| RA10 | |

Descripción de Requerimientos No Funcionales

| Atributo | Detalles y Restricción de Frontera |
|----------------------|--|
| Metáfora de Interfaz | (Detalle) formularios HTML. (Detalle) Colores alusivos a la empresa. |
| Plataforma | (Detalle) Servidor Web donde se almacenan las aplicaciones Web (Detalle) Lenguajes de Programación y Guiones: Javascript, PHP, HTML. (Detalle) Motor de Base de Datos: MySQL |

9.4 Análisis de Requisitos

Departamento académico

- 1) **Desarrollar una aplicación fácil de manejar:** este sistema tiene que ser entendible para que cualquiera persona con conocimientos básicos de informática lo pueda manejar.
- 2) **Desarrollar una aplicación para la sistematización de los datos personales de los estudiantes:** con este proceso se permitirá ingresar

todo tipo de datos como: cédula, nombre, etc. de los estudiantes.

- 3) **Desarrollar una aplicación para la sistematización de los datos personales de los docentes:** con este proceso se permitirá ingresar todo tipo de datos como: cédula, nombre, etc. de los docentes.
- 4) **Desarrollar una aplicación para la sistematización de las notas de los estudiantes:** con este proceso se permitirá al coordinador ingresar las notas en el sistema de forma que la información esté más organizada y con esto generar mayor funcionalidad para los estudiantes (consultas en cualquier momento) y la fundación.
- 5) **Desarrollar una aplicación que permita generar reportes de los estudiantes:** permitirle al coordinador académico generar reportes del estado académico de los estudiantes sin tener que usar otros medios como hojas electrónicas que no representan real utilidad para el proceso.
- 6) **Desarrollar una aplicación para la sistematización de los pensum de cada programa:** este formulario permitirá tener un control automatizado de cada materia que se esté dictando dentro de la institución y de esta forma se pueda manejar más fácilmente la planeación de cada programa académico que quiera o este realizando, asignándole a este el número de periodos de duración y cada una de las materias que se necesite dictar durante el transcurso del mismo. “Hacemos la aclaración este pensum no es una malla como tal sino es una forma organizada de llevar las materias que se dictan en cada uno de los semestre de la carreras que brinda la institución y de esta forma llevará un mejor control de las notas de los estudiantes”.
- 7) **Desarrollar una aplicación para la sistematización de las matrículas:** este formulario permitirá tener un control automatizado de cada matrícula que esté ingresando a la fundación con el fin de relacionar cada estudiante con un programa y así asignarle las notas correspondientes.
- 8) **Desarrollar una aplicación para la sistematización de la carga académica de cada docente:** con este proceso se permitirá llevar de manera organizada la información correspondiente a este proceso (docente y materias) que requiere de una asignación y verificación óptima por parte de la fundación.
- 9) **Implantar un chat para mantener comunicados los integrantes de cada uno de los departamentos:** diseñar un medio para mantener una

comunicación constante con los diferentes departamentos sin necesidad de desplazarse.

9.5 Especificación de requisitos (Departamento de académico)

○ Departamento académico

- 1) Desarrollar una aplicación fácil de manejar
- 2) Desarrollar una aplicación para la sistematización de los datos personales de los estudiantes
- 3) Desarrollar una aplicación para la sistematización de los datos personales de los docentes
- 4) Desarrollar una aplicación para la sistematización de las notas de los estudiantes
- 5) Desarrollar una aplicación que permita generar reportes de los estudiantes
- 6) Desarrollar una aplicación para la sistematización de los pensum de cada programa
- 7) Desarrollar una aplicación para la sistematización de las matrículas
- 8) Desarrollar una aplicación para la sistematización de la carga académica de cada docente

Los anteriores requisitos expuestos por el FDI son viable puesto que se cuenta con las herramientas necesarias como: HTML., Javascript , PHP y el motor de Base de Datos MySQL, estos componente permitirán llevar a cabo el desarrollo de esta aplicación involucrando el departamento académico.

Los requisitos como: Implantar un chat para mantener comunicado los integrantes de cada uno de los departamentos, este ítem se encuentra fuera del marco de trabajo puesto que sólo se va a desarrollar un aplicación para el departamento académico y no es necesario desarrollar un Chat.

10. INGENIERÍA DE INFORMACIÓN

10.1 MISIÓN

la misión del FDI es cumplir con las necesidades de las instituciones estatales, privada, la comunidad académica y profesional de la Costa Norte Colombiana, a través de una formación única fundamentada en los campos; profesional, técnico, humanístico y práctico, tomando como base la Formación integral, la cual estará cimentada bajo normas jurídicas procedí mentales, investigativas y técnico científicas.

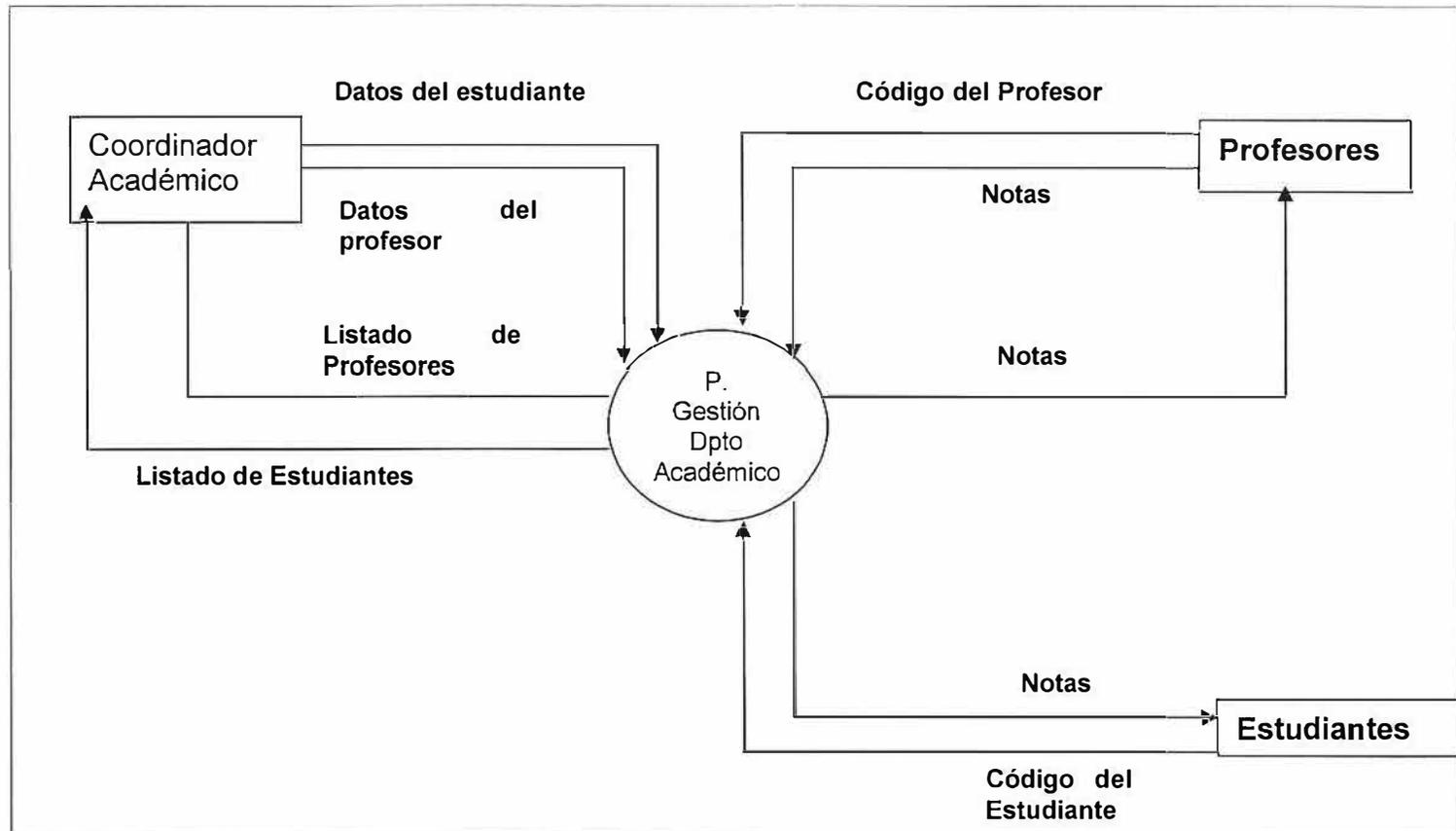
10.2 VISION

Se ven como una institución líder en la Costa Atlántica en los últimos avances de las Ciencias Forenses, la Criminalística y la Investigación Judicial, diseñando estrategias y acciones que permitan influir de manera notoria en la transformación cultural, como eje central para la obtención del perfil ideal que las entidades y la sociedad hoy requieren en la consolidación de sus resultados y de la paz social.

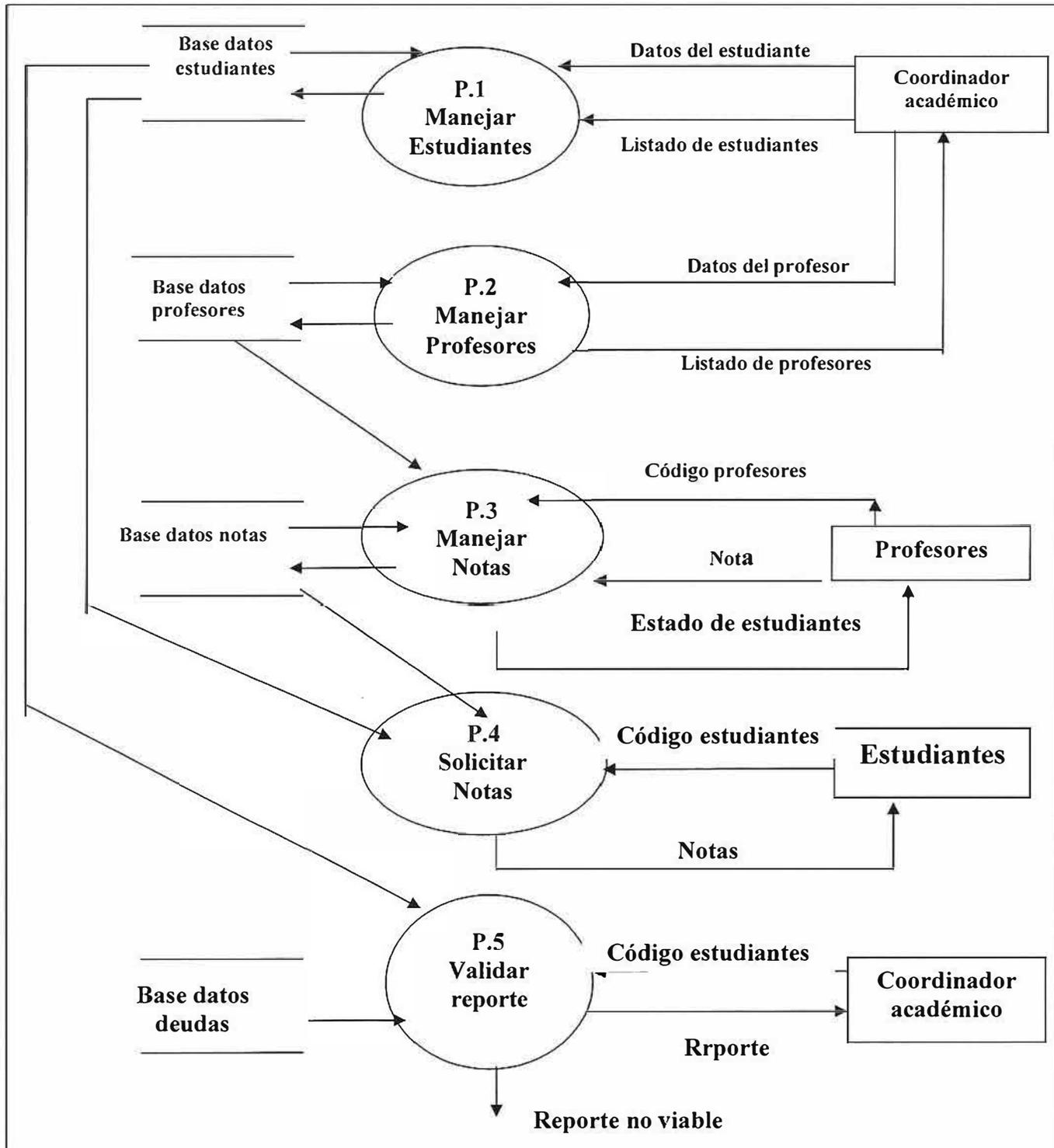
|

11. ANÁLISIS DEL SISTEMA

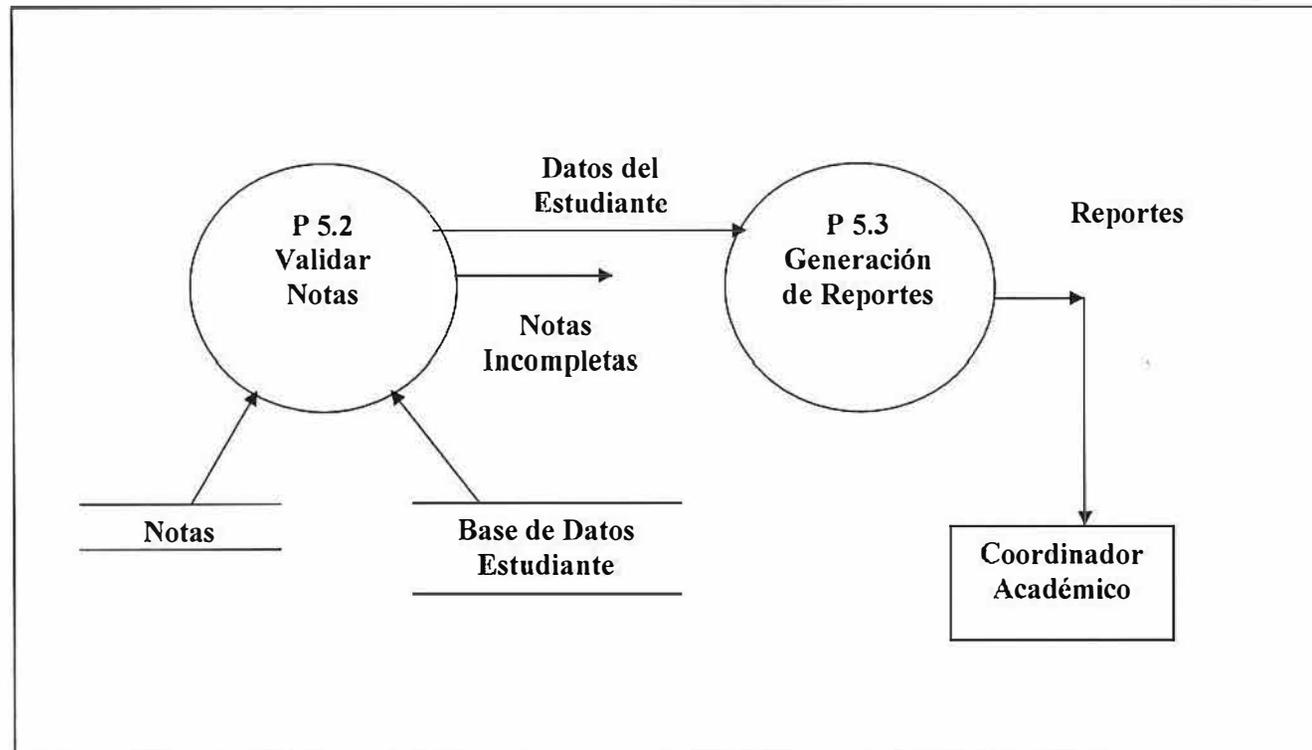
11.1 Diagrama Flujo Datos Dpto Académico(Contexto)



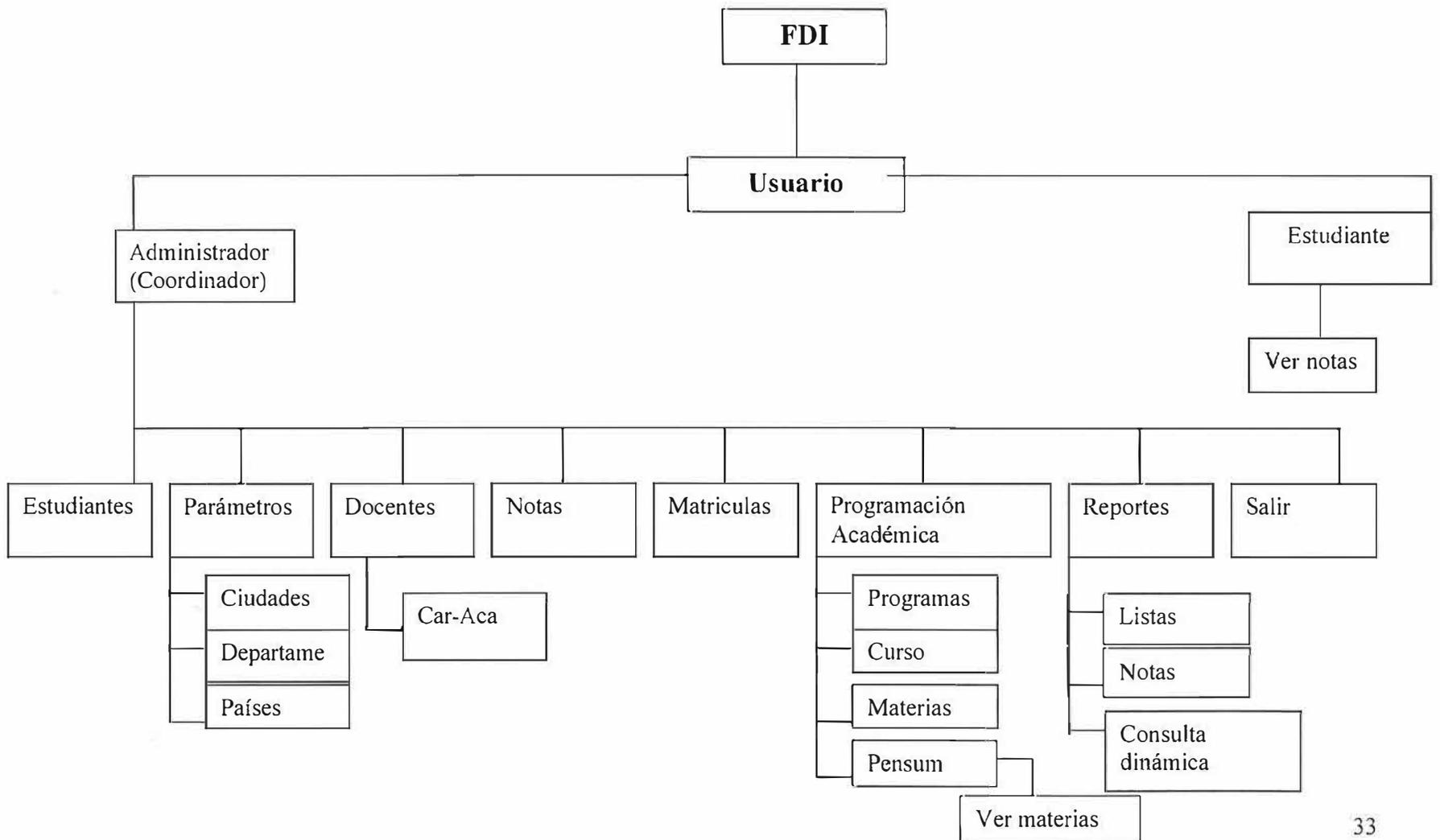
11.1.1 Diagrama Flujo Datos Dpto Académico Primer Nivel



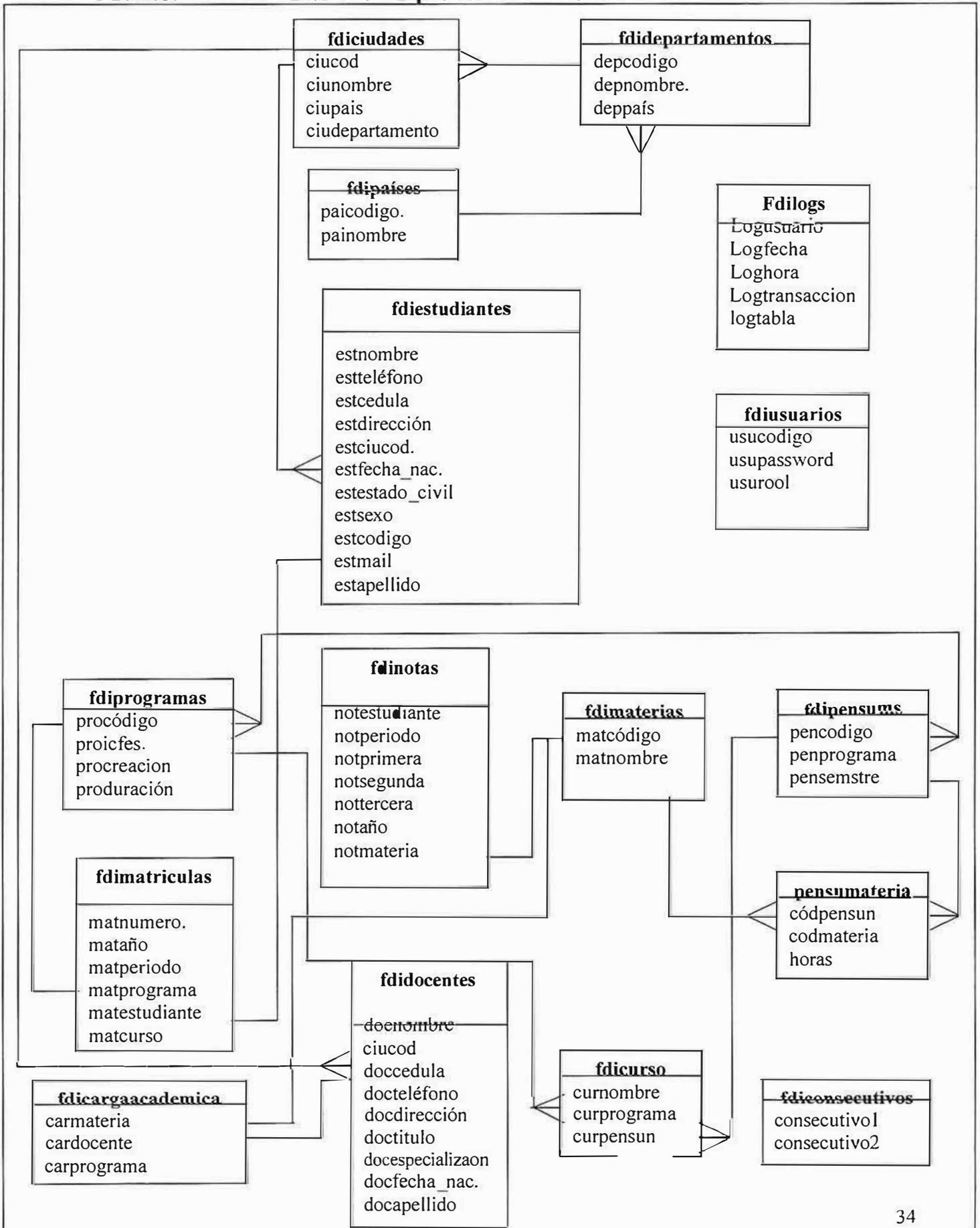
**|1.1.2 Diagrama flujo datos Dpto académico (segundo nivel)
Proceso 5 (Validar paz y salvo).**



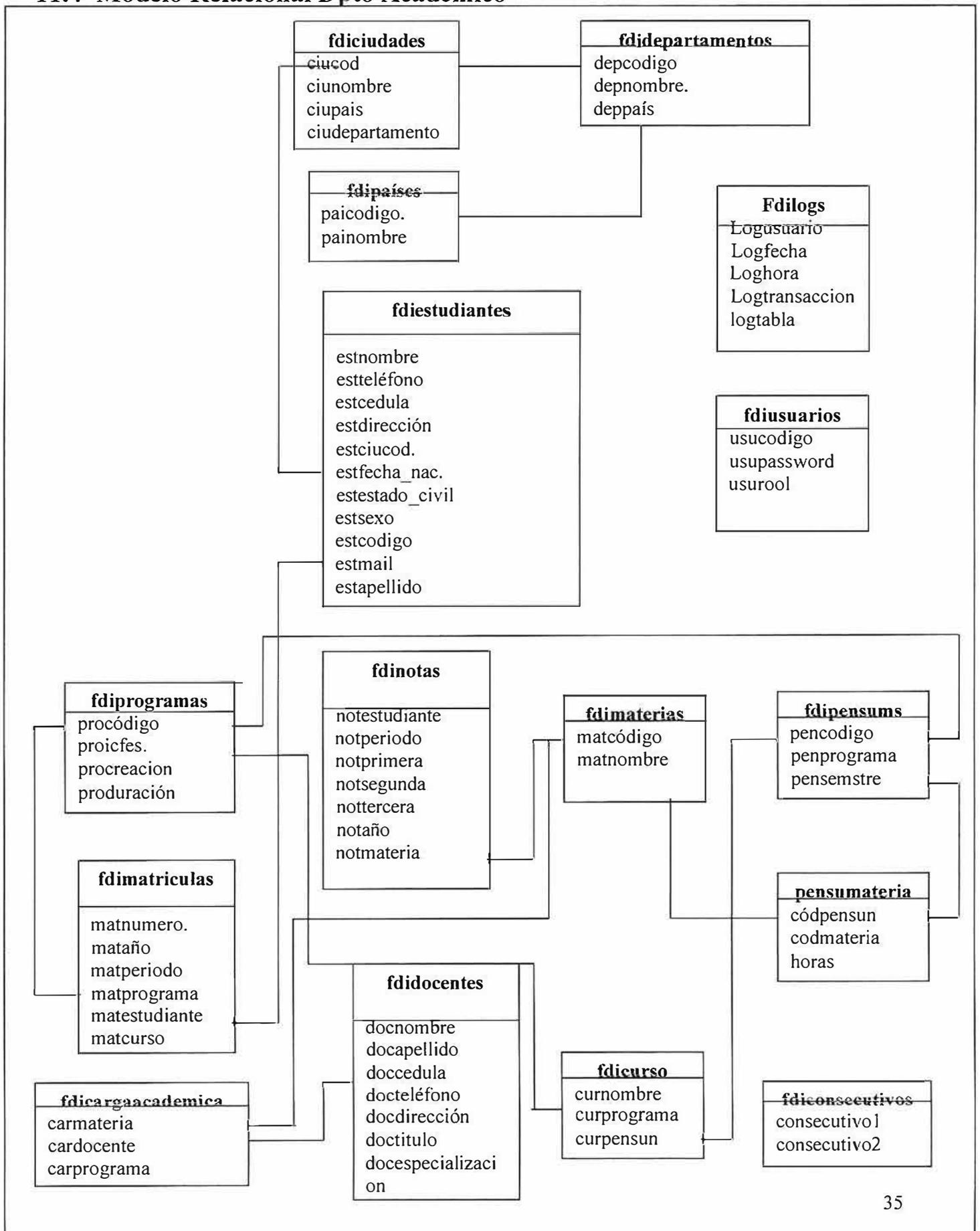
11.2 Mapa de Navegación



11.3 Modelo Entidad Relación Dpto Académico



11.4 Modelo Relacional Dpto Académico



11.5 Diccionario de datos modelo relacional de las tablas

| Nombre de la tabla: Fdiestudiantes | | | | Descripción de la tabla: en esta tabla estarán los datos de los estudiante el departamento | | |
|---|-----------------|-------|---------------|---|------|-------------------------------------|
| Nombre largo: Estudiantes | | | | | | |
| Modulo: Dep. académico | | | | | | |
| Tipo: Maestro | | | | | | |
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Estnombre | | Alfanumérico | 20 | No | Nombre de estudiante |
| 2 | Estcedula | | Alfanumérico | 15 | No | Identificación del est |
| 3 | Esttelefono | | Alfanumérico | 10 | No | Teléfono de est |
| 4 | Estdirección | | Alfanumérico | 20 | No | Dirección del est |
| 5 | Estcodigo | PK | Alfanumérico | 6 | No | Código del est |
| 6 | Estapellido | | Alfanumérico | 20 | No | Apellido del est |
| 7 | Estfecha_nac | | Fecha | | No | Fecha de nacimiento del est |
| 8 | Estestado_civil | | Alfanumérico | 10 | No | Estado civil del est |
| 9 | Ciucod | | Alfanumérico | 15 | No | Ciudad de nacimiento del estudiante |
| 10 | Estsexo | | Carácter | 2 | No | Sexo del est |
| 11 | Estmail | | Alfanumerico | 50 | No | Mail del est |

| Nombre de la tabla: Fdiprogramas | | | | Descripción de la tabla: en esta tabla estarán todos los programas académicos presente en el FDI | | |
|---|-------------|-------|---------------|---|------|--------------------------------|
| Nombre largo: Programas académicos | | | | | | |
| Modulo: Dep. académico | | | | | | |
| Tipo: Maestro | | | | | | |
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Procodigo | Pk | Alfanumérico | 6 | No | Código de la carrera |
| 2 | Pronombre | | Alfanumérico | 30 | No | Nombre de la carrera |
| 3 | Proicfes | | Alfanumérico | 15 | No | Número de icfes |
| 4 | Procreación | | Fecha | | No | Fecha de creación del programa |
| 5 | Produracion | | Alfanumérico | 20 | No | Duración del programa |

| Nombre de la tabla: Fdi matriculas | | | Descripción de la tabla: en esta tabla estarán relacionados todas las matriculas de los alumnos con sus respectivos programas | | | |
|---|----------------|-------|--|------|------|---|
| Nombre largo: Estudiantes matriculas | | | | | | |
| Modulo: Dep. académico | | | | | | |
| Tipo: Maestro | | | | | | |
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Matnumero | Pk | Alfanumérico | 10 | No | Consecutivo de la matricula |
| 2 | Matestudiantes | Fk | Alfanumérico | 15 | No | Cedula del estudiante |
| 3 | Matprograma | Fk | Alfanumérico | 6 | No | Código del estudiante |
| 4 | Matperiodo | | Alfanumérico | 10 | No | Período del año |
| 5 | Matcurso | | Alfanumérico | 20 | No | Curso en el que se va encontrar el estudiante |
| 6 | Mataño | | Alfanumérico | 10 | No | Año en el que se matriculo el estudiante |

| Nombre de la tabla: Fdiciudades | | | Descripción de la tabla: en esta tabla estarán las diferentes ciudades de Colombia y el mundo | | | |
|--|-----------------|-------|--|------|------|---|
| Nombre largo: Ciudades | | | | | | |
| Modulo: Departamento académico | | | | | | |
| Tipo: parámetro | | | | | | |
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Ciucod | PK | Alfanumérico | 15 | No | Código de la ciudad |
| 2 | Ciunombre | | Alfanumérico | 25 | No | Nombre de la ciudad |
| 3 | Ciupais | FK | Alfanumérico | 15 | No | País al que pertenece la ciudad |
| 4 | Ciudepartamento | FK | Alfanumérico | 20 | No | Departamento al que pertenece la ciudad |

| Nombre de la tabla: Fdconsecutivo | | | Descripción de la tabla: en esta tabla estarán los consecutivo de la matricula | | | |
|--|--------------|-------|---|------|------|---------------------------|
| Nombre largo: Consecutivo | | | | | | |
| Modulo: Departamento académico | | | | | | |
| Tipo: parámetro | | | | | | |
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Consecutivo1 | PK | Alfanumérico | 10 | No | Consecutivo de estudiante |
| 2 | Consecutivo2 | | Alfanumérico | 10 | No | Consecutivo de matricula |

| Nombre de la tabla: Fdcurso | | | Descripción de la tabla: en esta tabla estarán los diferentes cursos de los diversos programas | | | |
|---------------------------------------|-------------|-------|---|------|------|---------------------|
| Nombre largo: Curso | | | | | | |
| Modulo: Departamento académico | | | | | | |
| Tipo: Detalle | | | | | | |
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Curnombre | PK | Alfanumérico | 10 | No | Nombre del curso |
| 2 | Curprograma | FK | Alfanumérico | 20 | No | Código del programa |
| 3 | Curpensum | FK | Alfanumérico | 20 | No | Código del pensum |

| Nombre de la tabla: Fddepartamento | | | Descripción de la tabla: en esta tabla estarán los diferentes departamentos de Colombia y el mundo | | | |
|---|-----------|-------|---|------|------|-------------------------|
| Nombre largo: Departamento | | | | | | |
| Modulo: Departamento académico | | | | | | |
| Tipo: parámetro | | | | | | |
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Depcodigo | PK | Alfanumérico | 20 | No | Código del departamento |
| 2 | Depnombre | | Alfanumérico | 20 | No | Nombre del departamento |
| 3 | Deppais | FK | Alfanumérico | 25 | No | Código del país |

| Nombre de la tabla: Fdipaises | | | Descripción de la tabla: en esta tabla estarán los diferentes del mundo | | | |
|---------------------------------------|-----------|-------|--|------|------|-----------------|
| Nombre largo: Países | | | | | | |
| Modulo: Departamento académico | | | | | | |
| Tipo: parámetro | | | | | | |
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Paicodigo | PK | Alfanumérico | 20 | No | Código del país |
| 2 | Painombre | | Alfanumérico | 20 | No | Nombre del país |

| Nombre de la tabla: Fdipensumateria | | | Descripción de la tabla: en esta tabla estarán los diferentes pensum relacionados con las materias | | | |
|--|------------|-------|---|------|------|-------------------------------|
| Nombre largo: Pensum de la materias | | | | | | |
| Modulo: Departamento académico | | | | | | |
| Tipo: Detalle | | | | | | |
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Codpensum | FK | Alfanumérico | 10 | No | Código del pensum |
| 2 | Codmateria | FK | Alfanumérico | 10 | No | Código de la materia |
| 3 | Hora | | Alfanumérico | 10 | No | Número de horas de la materia |

| Nombre de la tabla: Fdinotas | | | Descripción de la tabla: en esta tabla estarán las notas de los estudiante durante un semestre en específico | | | |
|---------------------------------------|---------------|-------|---|------|------|---|
| Nombre largo: Notas | | | | | | |
| Modulo: Departamento académico | | | | | | |
| Tipo: Maestro | | | | | | |
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Notestudiante | Fk | Alfanumérico | 20 | No | Código de estudiante |
| 2 | Notprimera | | Float | | No | Primera nota |
| 3 | Notsegunda | | Float | | No | Segunda nota |
| 4 | Nottercera | | Float | | No | Tercera nota |
| 5 | Notmateria | Fk | Alfanumérico | 20 | No | Código de la materia |
| 6 | Notano | | Alfanumérico | 20 | No | Notas de año en el que está el estudiante |
| 7 | Notperiodo | | Alfanumérico | 20 | No | Período en el que está el estudiante |

| Nombre de la tabla: Fdimaterias | | | Descripción de la tabla: en esta tabla estarán los códigos y descripciones de las materias que dan los estudiante en los diferentes programas | | | |
|--|-----------|-------|--|------|------|----------------------|
| Nombre largo: Materias | | | | | | |
| Modulo: Departamento académico | | | | | | |
| Tipo: Maestro | | | | | | |
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Matcodigo | Pk | Alfanumérico | 6 | No | Código de la materia |
| 2 | Matnombre | | Alfanumérico | 20 | No | Nombre de la materia |



| Nombre de la tabla: Fdidocentes Nombre largo: profesores Modulo: Departamento académico Tipo: Maestro | | | Descripción de la tabla: en esta tabla estarán los datos de los profesores | | | |
|--|--------------------|-------|---|------|------|-----------------------------------|
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Docnombre | | Alfanumérico | 20 | No | Nombre de los profesores |
| 2 | Doccedula | Pk | Alfanumérico | 12 | No | Cedula de los profesores |
| 3 | Docdireccion | | Alfanumérico | 25 | No | Dirección del profesor |
| 4 | Doctelefono | | Alfanumérico | 20 | No | Teléfono del profesor |
| 5 | Docapellidos | | Alfanumérico | 20 | No | Apellido del profesor |
| 6 | Doctitulo | | Alfanumérico | 20 | No | Título del profesor |
| 7 | Docespecializacion | | Alfanumérico | 20 | No | Especialización del profesor |
| 8 | Docfecha_nac date | | Fecha | | No | Fecha de nacimiento del Profesor |
| 9 | Ciucod | | Alfanumérico | 15 | No | Ciudad de nacimiento del Profesor |

| Nombre de la tabla: Fdicarga_académica Nombre largo: carga_académica Modulo: Departamento académico Tipo: Detalle | | | Descripción de la tabla: en esta tabla estarán los códigos de las materias relacionados con los respectivos profesores que dictan dichas materias . | | | |
|--|-------------|-------|--|------|------|----------------------|
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Carmateria | Fk | Alfanumerico | 20 | No | Código de la materia |
| 2 | Cardocente | Fk | Alfanumerico | 15 | No | Cedula del profesor |
| 3 | Carprograma | FK | Alfanumerico | 20 | No | Código del programa |

| Nombre de la tabla: Fdilogs Nombre largo: logs Modulo: Departamento académico Tipo: Detalle | | | Descripción de la tabla: en esta tabla estarán toda la modificaciones que el usuario realice en la base de datos | | | |
|--|----------------|--------------|---|------------------|-------------|-------------------------|
| N o | Nombre | Llave | Tipo de datos | Lon g | Nulo | Descripción |
| 1 | Logusuario | | Alfanumérico | 20 | No | Código del usuario |
| 2 | Logfecha | | Fecha | | No | Fecha de la transacción |
| 3 | Loghora | | Tiempo | | No | Hora de la transacción |
| 4 | Logtransaccion | | Alfanumérico | 20 | No | Movimientos que se hizo |
| 5 | Logtabla | | Alfanumérico | 20 | No | Tabla afectada |

| Nombre de la tabla: Fdiusuarios Nombre largo: usuarios Modulo: Departamento académico Tipo: Detalle | | | Descripción de la tabla: en esta tabla estarán todos los usuarios que hagan uso de la aplicación | | | |
|--|---------------|--------------|---|------------------|-------------|------------------------|
| N o | Nombre | Llave | Tipo de datos | Lon g | Nulo | Descripción |
| 1 | Usucodigo | | Alfanumérico | 32 | No | Código del usuario |
| 2 | Usupassword | | Alfanumérico | 32 | No | Contraseña del usuario |
| 3 | Usurool | | Alfanumérico | 32 | No | Tipo de usuario |

11.6 Análisis De Contenido

| Nombre de contenido | Descripción | Tipo | Formato |
|--------------------------|--|----------------|--------------|
| Escudo de la institución | Escudo oficial característico del FDI | Imagen | Jpg |
| Estudiantes | Un formulario utilizado para que los estudiantes haga la consulta de sus notas | Texto/link | Jpg/html/Php |
| Coordinador | Varios formularios serán utilizados por el coordinador para que el maneje todo lo relacionado con la gestión académica | Texto/link | Jpg/html |
| Base de datos | Lugar donde se guardará toda la información necesaria para el manejo del departamento académico | almacenamiento | |

11.7 Análisis Funcional

Departamento académico

| | |
|--------------------|---|
| Caso de uso | Generación de programas y pensums |
| Actores | Altos |
| Descripción | En esta función el coordinador podrá crear los diferentes programas con sus respectivos pensums |

| | |
|--------------------|---|
| Caso de uso | Modificación de programas y pensums |
| Actores | Altos |
| Descripción | En esta función el coordinador podrá modificar los diferentes programas con sus respectivos pensums |

| | |
|--------------------|--|
| Caso de uso | Eliminación de programas y pensums |
| Actores | Altos |
| Descripción | En esta función el coordinador podrá eliminar los diferentes programas con sus respectivos pensums |

| | |
|--------------------|---|
| Caso de uso | Ingresar los datos personales de los estudiantes |
| Actores | Altos |
| Descripción | En esta función el coordinador ingresará todos los datos personales de los alumnos. |

| | |
|--------------------|---|
| Caso de uso | Eliminar los datos personales de los estudiantes |
| Actores | Altos |
| Descripción | En esta función el coordinador eliminará todos los datos personales de los alumnos. |

| | |
|--------------------|--|
| Caso de uso | Modificar los datos personales de los estudiantes |
| Actores | Altos |
| Descripción | En esta función el coordinador modificará todos los datos personales de los alumnos cuando se presente algún cambio. |

| | |
|--------------------|---|
| Caso de uso | Consultar los datos personales de los estudiantes |
| Actores | Altos |
| Descripción | En esta función el coordinador consultará todos los datos personales de los alumnos cuando sea necesario. |

| | |
|--------------------|--|
| Caso de uso | Generación de matriculas |
| Actores | Altos |
| Descripción | En esta función el coordinador podrá matricular a los diversos estudiantes en su respectivo programa académico |

| | |
|--------------------|---|
| Caso de uso | Modificación de matriculas |
| Actores | Altos |
| Descripción | En esta función el coordinador podrá modificar las matriculas existentes. |

| | |
|--------------------|---|
| Caso de uso | Generar los listados de los estudiantes matriculados en los diferentes programas |
| Actores | Altos |
| Descripción | En esta función le coordinador podrá listar o hacer reportes de los diferentes estudiantes correspondiente a un curso en especifico |

| | |
|--------------------|---|
| Caso de uso | Ingreso de las notas de los estudiantes |
| Actores | Altos |
| Descripción | En esta función el coordinador ingresarán todas las notas de los estudiantes. |

| | |
|--------------------|---|
| Caso de uso | Modificar las notas de los estudiantes |
| Actores | Altos |
| Descripción | En esta función el coordinador modificarán todas las notas de los estudiantes, siempre y cuando sea justificadamente. |

| | |
|--------------------|--|
| Caso de uso | Consultar las notas de los estudiantes |
| Actores | Altos |
| Descripción | En esta función le coordinador consultarán todas las notas de los estudiantes. |

| | |
|--------------------|--|
| Caso de uso | Ingresar los datos personales de los profesores |
| Actores | Altos |
| Descripción | En esta función el coordinador ingresará todos los datos personales de los profesores. |

| | |
|--------------------|--|
| Caso de uso | Eliminar los datos personales de los profesores |
| Actores | Altos |
| Descripción | En esta función el coordinador eliminará todos los datos personales de los profesores. |

| | |
|--------------------|---|
| Caso de uso | Modificar los datos personales de los profesores |
| Actores | Altos |
| Descripción | En esta función el académico modificará todos los datos personales de los profesores cuando se presente algún cambio. |

| | |
|--------------------|--|
| Caso de uso | Consultar los datos personales de los profesores |
| Actores | Altos |
| Descripción | En esta función el coordinador consultará todos los datos personales de los profesores cuando sea necesario. |

| | |
|--------------------|---|
| Caso de uso | Asignación de materias |
| Actores | Altos |
| Descripción | En esta función el coordinador podrá asignar a cada uno de los profesores las diferentes materias que dictará en los diversos programas |

| | |
|--------------------|---|
| Caso de uso | Modificación de asignación de materias |
| Actores | Altos |
| Descripción | En esta función el coordinador podrá modificar las asignación de materias a los profesores. |

| | |
|--------------------|--|
| Caso de uso | Eliminación de asignación de materias |
| Actores | Altos |
| Descripción | En esta función el coordinador podrá eliminar las asignaciones de las materias a los profesores. |

| | |
|--------------------|--|
| Caso de uso | Consulta de notas |
| Actores | Bajos |
| Descripción | En esta función los estudiantes podrán consultar todas sus notas del semestre que están cruzando . |

11.8 Análisis de Interacción

Consultar por parte de los estudiantes sus notas correspondientes: los estudiantes tendrán la oportunidad de consultar sus notas ingresando su código estudiantil y debe estar previamente registrado en la base de datos de la aplicación.

Permitirle al coordinador la realización de sus diferentes funciones: el coordinador tendrá la oportunidad de realizar todas sus funciones como: modificar , ingresar, consultar los datos personales de los estudiantes , notas, generar planillas ya sea por curso, materia, docente o periodo e ingresar, modificar los datos personales de los profesores.

El tipo de usuario que va a interactuar con la aplicación son los **usuarios registrados** en la aplicación y a la vez este tipo de usuarios se clasifica en:

- Usuarios altos: son aquellos que puede realizar cualquier procedimiento presente en la aplicación y no tiene ninguna restricción para hacer l a modificación que deseen.
- Usuarios bajos: son aquellos que solamente tiene permiso de leer una determinada información por ejemplo los estudiantes que sólo pueden leer sus notas.

11.9 Estandarización de las tablas

Estandarización de tablas

Departamento de académico

```
# phpMyAdmin MySQL-Dump
# http://phpwizard.net/phpMyAdmin/
#
# Host: localhost Database : FDI

# -----
#
# Table structure for table 'fdicargaacademica'
#

CREATE TABLE fdicargaacademica (
  carmateria varchar(20) DEFAULT '0' NOT NULL,
  cardocente varchar(15) NOT NULL,
  carprograma varchar(20) NOT NULL
);

# -----
#
# Table structure for table 'fdiciudades'
#

CREATE TABLE fdiciudades (
  ciucod varchar(15) DEFAULT '0' NOT NULL,
  ciunombre varchar(25) NOT NULL,
  ciupais varchar(15),
  ciudepartamento varchar(20),
  PRIMARY KEY (ciucod)
);

# -----
#
# Table structure for table 'fdiconsecutivo'
#
```

```
CREATE TABLE fdiconsecutivo (  
    consecutivo1 varchar(10) DEFAULT '0' NOT NULL,  
    consecutivo2 varchar(10) NOT NULL,  
    PRIMARY KEY (consecutivo1)  
);
```

```
# -----  
#  
# Table structure for table 'fdicursos'  
#
```

```
CREATE TABLE fdicursos (  
    curnombre varchar(10) DEFAULT '0',  
    curprograma varchar(20),  
    curpensum varchar(20)  
);
```

```
# -----  
#  
# Table structure for table 'fdidepartamentos'  
#
```

```
CREATE TABLE fdidepartamentos (  
    depcodigo varchar(20) DEFAULT '0',  
    depnombre varchar(20),  
    deppais varchar(25)  
);
```

```
# -----  
#  
# Table structure for table 'fdidocentes'  
#
```

```
CREATE TABLE fdidocentes (  
    doccedula varchar(12) DEFAULT '0',  
    docnombre varchar(20) NOT NULL,  
    docapellidos varchar(20),  
    doctelefono varchar(20) NOT NULL,  
    docdireccion varchar(25) NOT NULL,  
    doctitulo varchar(20) NOT NULL,
```

```
docespecializacion varchar(20) NOT NULL,  
docfecha_nac date DEFAULT '0000-00-00' NOT NULL,  
ciucod varchar(15) NOT NULL  
);
```

```
# -----  
#  
# Table structure for table 'fdiestudiantes'  
#
```

```
CREATE TABLE fdiestudiantes (  
  estcodigo varchar(6) NOT NULL,  
  estcedula varchar(15) NOT NULL,  
  estnombre varchar(20) DEFAULT '0',  
  estapellidos varchar(20),  
  esttelefono varchar(10),  
  estdireccion varchar(20),  
  ciucod varchar(15) NOT NULL,  
  estfecha_nac date DEFAULT '2000-00-00',  
  estestado_civil varchar(10),  
  estsexo char(2),  
  estmail varchar(50),  
  PRIMARY KEY (estcodigo)  
);
```

```
# -----  
#  
# Table structure for table 'fdilogs'  
#
```

```
CREATE TABLE fdilogs (  
  logusuario varchar(20) NOT NULL,  
  logfecha date DEFAULT '0000-00-00' NOT NULL,  
  loghora time DEFAULT '00:00:00' NOT NULL,  
  logtransaccion varchar(20) NOT NULL,  
  logtabla varchar(20) NOT NULL  
);
```

```
# -----  
#
```

```
# Table structure for table 'fdimaterias'
```

```
#
```

```
CREATE TABLE fdimaterias (  
  matcodigo varchar(6) DEFAULT '0' NOT NULL,  
  matnombre varchar(20) NOT NULL,  
  PRIMARY KEY (matcodigo)  
);
```

```
# -----
```

```
#
```

```
# Table structure for table 'fdimaticulas'
```

```
#
```

```
CREATE TABLE fdimaticulas (  
  matnumero varchar(10) NOT NULL,  
  matestudiantes varchar(15) NOT NULL,  
  matprograma varchar(6) NOT NULL,  
  matperiodo varchar(10) NOT NULL,  
  matcurso varchar(20) NOT NULL,  
  mataño varchar(10) NOT NULL  
);
```

```
# -----
```

```
#
```

```
# Table structure for table 'fdinotas'
```

```
#
```

```
CREATE TABLE fdinotas (  
  notano varchar(20) NOT NULL,  
  notperiodo varchar(20) NOT NULL,  
  notestudiante varchar(20) NOT NULL,  
  notmateria varchar(20) NOT NULL,  
  notprimera float DEFAULT '0' NOT NULL,  
  notsegunda float DEFAULT '0' NOT NULL,  
  nottercera float DEFAULT '0' NOT NULL  
);
```

```
# -----
```

```
#
```



```
# Table structure for table 'fdipaises'  
#
```

```
CREATE TABLE fdipaises (  
  paicodigo varchar(20) DEFAULT '0',  
  painombre varchar(20)  
);
```

```
# -----  
#  
# Table structure for table 'fdipensumateria'  
#
```

```
CREATE TABLE fdipensumateria (  
  codpensum varchar(10) DEFAULT '0',  
  codmateria varchar(10),  
  horas varchar(10)  
);
```

```
# -----  
#  
# Table structure for table 'fdipenums'  
#
```

```
CREATE TABLE fdipenums (  
  penprograma varchar(6) DEFAULT '0' NOT NULL,  
  pencodigo varchar(20),  
  pensemestre varchar(20)  
);
```

```
# -----  
#  
# Table structure for table 'fdiprogramas'  
#
```

```
CREATE TABLE fdiprogramas (  
  procodigo varchar(6) DEFAULT '0' NOT NULL,  
  pronombre varchar(30),  
  proicfes varchar(15) NOT NULL,  
  procreacion date DEFAULT '0000-00-00' NOT NULL,
```

```
    produccion varchar(20),  
    PRIMARY KEY (procodigo)  
);
```

```
# -----  
#  
# Table structure for table 'fdiusuarios'  
#
```

```
CREATE TABLE fdiusuarios (  
    usucodigo varchar(32),  
    usupassword varchar(32),  
    usurool varchar(32)
```

BIBLIOGRAFIA

[AndreuRicartValor] Estrategia y Sistemas de Información, Segunda Edición, Rafael Andreu, Joan E. Ricart, Josep Valor, McGraw-Hill de Management, 1996, ISBN 9-788448-105082

[GarciaChamorroMolina] Informática de Gestión y Sistemas de Información, F. García, F. Chamorro, J. M. Molina, McGraw-Hill, 2000, ISBN: 84-481-2767-6

"The role of the CIO and IT function in ERP", Leslie P. Willcocks and Richard Sykes, Communications of the ACM, April 2000/Vol. 43, No. 4, p. 32.

"Are ERPs Universal", Communications of the ACM, April 2000/Vol. 43, No. 4, p. 47
[GestionInformacion] Informática de Gestión y Sistemas de Información, Fernando García Pérez, Félix Chamorro Atance, José Manuel Molina López, McGraw-Hill, 2000, ISBN 84-481-2767-6

e-Commerce con Linux - Guía práctica de comercio electrónico, Conectiva, Prentice Hall, 2000, ISBN 958-699-039-7

Smart: Lo fundamental y lo más efectivo acerca de la Gerencia del Conocimiento, Thomas M. Koulopoulos, Carl Frappaolo, McGraw-Hill, 2000, ISBN: 958410152-8, cap. 1, 2 y 3

**Desarrollo de un Software en Entorno Web
para el Manejo del Departamento Académico
en la Fundación para el Desarrollo de la Investigación
(FDI)**

Carlos Julio Nova Peñaranda

Adriana Patricia Arzayus Ecobar

Alberto Porto Danado

Universidad Simón Bolívar

Facultad

Ingeniería de Sistemas

Barranquilla





TABLA DE CONTENIDO

| | Página |
|--|--------|
| INTRODUCCIÓN | 2 |
| 1. INSTALACION..... | 3 |
| 2. REQUERIMIENTOS BASICOS..... | 4 |
| 3. MODELO DE ENTIDAD RELACIÓN | 5 |
| 3.1. DESCRIPCION DE MODELO ENTIDAD RELACIÓN | 6 |
| 3.2. DICCIONARIO DE DATOS MODELO DE ENTIDAD RELACION DE LAS TABLAS | 7 |
| 4. SCRIPT DE CREACIÓN | 13 |
| 5. DESCRIPCIÓN PROCEDIMIENTOS DEL SISTEMA..... | 18 |
| 5.1. MODIFICACIÓN DE ESTUDIANTES..... | 18 |
| 5.2. MODIFICACIÓN DE DOCENTES..... | 22 |
| 5.3. MODIFICACIÓN DE MATRICULAS | 28 |
| 5.4. MODIFICACIÓN DE PAISES | 33 |
| 5.5. MODIFICACIÓN DE DEPARTAMENTOS..... | 36 |
| 5.6. MODIFICACIÓN DE CIUDADES..... | 40 |
| 5.7. AUDITORÍA (LOGS DE EVENTOS)..... | 44 |
| 5.8. MODIFICACIÓN DE PROGRAMAS..... | 47 |
| 5.9. MODIFICACIÓN DE PENSUMS | 50 |
| 5.10. MODIFICACIÓN DE CURSOS | 53 |
| 5.11. MODIFICACIÓN DE MATERIAS | 57 |
| 5.12. MODIFICACIÓN DE USUARIOS..... | 60 |
| 5.13. MODIFICACIÓN DE NOTAS POR CURSO | 63 |
| 5.14. MODIFICACIÓN DE NOTAS POR ESTUDIANTES | 66 |
| 5.15. REPORTE LISTA DE ASISTENCIA..... | 69 |
| 5.16. REPORTE DE NOTAS | 70 |
| 5.17. REPORTES DINAMICOS | 71 |



INTRODUCCIÓN

Los sistemas administradores de información son una serie de herramientas que permiten mantener de una forma rápida y eficiente todo el sistema de información que es utilizado en las empresas e instituciones que aplican esta tecnología. De esta manera La fundación para el desarrollo de la investigación FDI quiere ser participe en este sistema para mejorar cada uno de los aspectos.

Los sistemas administradores de información deben ser manipulados por el usuario, y para que esta etapa se pueda desarrollar de manera satisfactoria, el sistema debe estar en la capacidad de poder brindar una ayuda en línea para que así el usuario pueda comunicarse con claridad y eficacia. También cabe anotar que por parte de los diseñadores deben entregar un manual del usuario y un manual del sistema, para que el usuario pueda documentarse y familiarizarse con toda la información con la que cuenta este sistema.

Los sistemas administradores de información tienen una finalidad y es permitir que el usuario manipule de forma eficiente y segura toda la información que posee este sistema.



Fdi manual del sistema

1. INSTALACION

Para instalar la aplicación en entorno web, de La fundación para el desarrollo de la investigación FDI, en su ordenador debe utilizar el lenguaje de programación PHP , la base de datos y toda la biblioteca que contiene los sistemas de ayuda para este programa.

El proceso de instalación de este software es muy sencillo, solo debemos seguir los pasos del asistente de la instalación del programa.

Debemos tener en cuenta que para realizar la instalación de la aplicación en entorno web tenemos que conocer las características del equipo en el cual se va a trabajar. De esta manera tendremos una forma eficiente y confiable la información de los estudiantes en el FDI.

Existen herramientas para el soporte de información como son:

Motor de bases de datos MySQLFRONT

Herramientas de diseño e implementación de software(PHP, Javastrip,HTML.).



2. REQUERIMIENTOS BASICOS

La aplicación en entorno web, de La fundación para el desarrollo de la investigación FDI, es una aplicación que opera a 32 bits, permitiendo de esta manera requerir la ayuda de un hardware y un software que permitan una buena instalación y eficiente operación

Esta configuración realizada a 32 bits, permite un mejoramiento en el momento de procesar la información, de esta manera es necesario instalar dicha aplicación sobre una plataforma que sea capaz de soportar archivos de 32 bits

La plataforma operacional Windows XP, reúne todas las especificaciones antes mencionadas.

Por medio de la información anterior debemos tener en cuenta cual es la velocidad del procesador durante el procesamiento del equipo en el cual se haya definido la instalación de la aplicación en entorno a web. Por este motivo debemos contar con procesadores o de mayor capacidad.

| REQUERIMIENTOS | CARACTERISTICAS |
|-------------------------------------|--------------------|
| Sistema Operativo | Windows XP. |
| Espacio disponible en el disco duro | 1 GB |
| Memoria Ram | 256 MB en adelante |
| Procesador | 266Mhz |

Para tener un adecuado funcionamiento en el sistema debemos contar con un computador que reúna las siguientes características:

Hardware:

Procesador 256 Mhz en adelante.

Espacio disponible en el disco duro de 1GB.

Memoria Ram 256 MB en adelante.

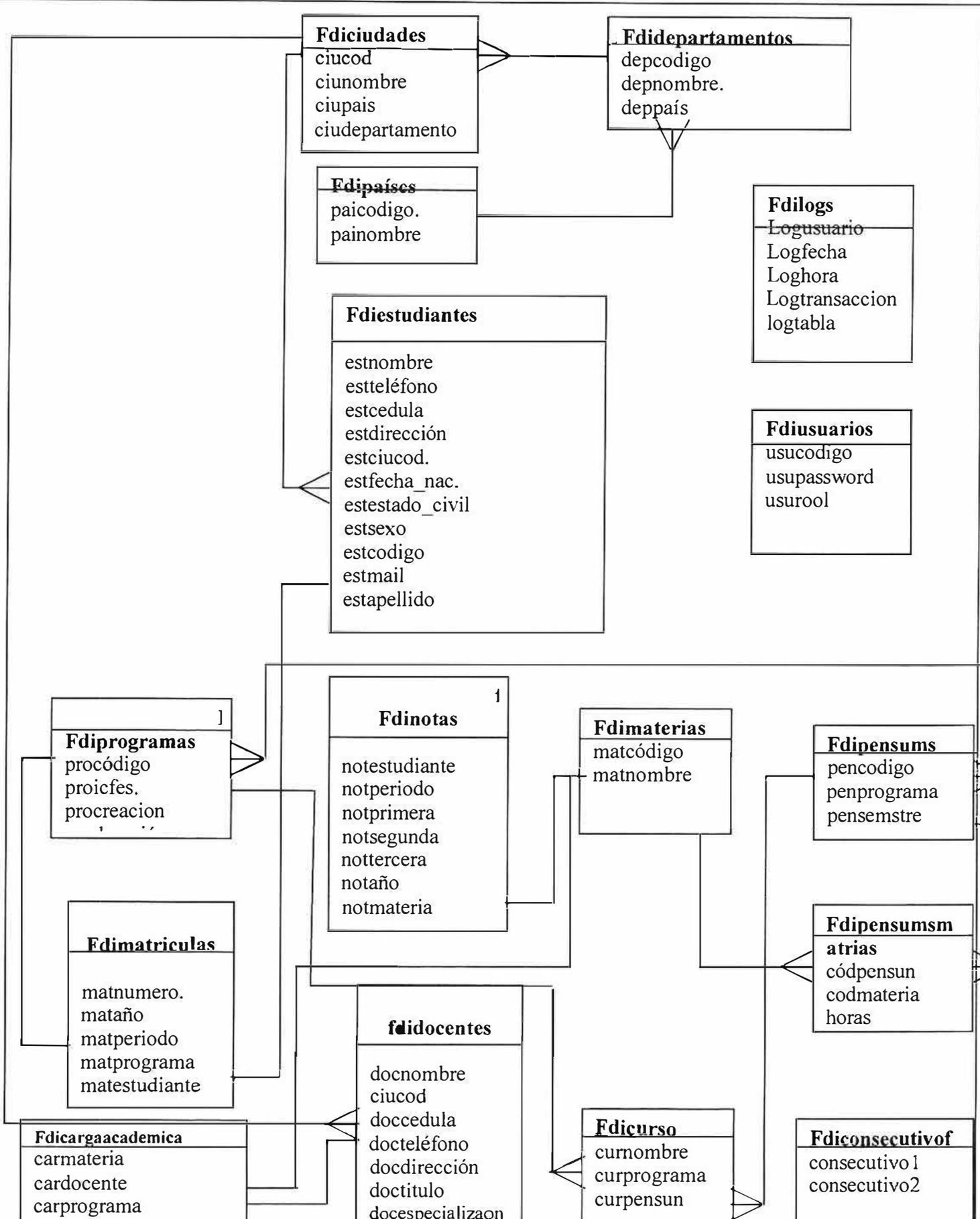
SOFTWARE:

Sistema Operativo: Windows XP.



Fdi manual del sistema

3. MODELO DE ENTIDAD RELACIÓN





Fdi manual del sistema

3.1.DESCRIPCION DE MODELO ENTIDAD RELACIÓN

Las entidades con las que cuenta nuestro Modelo Entidad Relación son:

| ENTIDADES | |
|-------------------------|---|
| FDIESTUDIANTES | Hace referencia a los datos personales de los estudisntes |
| FDIDOCENTES | Hace referencia a los datos personales de los profesores y las mater que dicta |
| FDINOTAS | Hace referencia a las notas de los estudiantes |
| FDIPROGRAMAS | Hace referencia a cada facultad de la universidad |
| FDIMATRICULAS | Aquí estarán relacionadas todas las matriculas de los alumnos con s respectivos programas |
| FDIMATERIAS | Hace referencia a cada asignatura que esta relacionada con los prog de la universidad |
| FDICURSOS | Hace referencia a aulas de los diferentes programas que tiene el FD |
| FDIPENSUMS | Hace referencia a todos los semestres de las carreras que se se dicta el FDI. |
| FDIPENSUMMATERIA | Hace referencia a todas las materias que se ven en un semestre determinado de una carrera |
| FDICARGA_ACAD. | Hace referencia a todas las materias que se le asignan a los profesore |
| FDICIUDADES | Hace referencia a las ciudades de procedencia, alumnos y profesore |
| FDIDEPARTAMENTOS | Hace referencia a los departamentos de procedencia, alumnos y profesores. |
| FDIPAISES | Hace referencia a los países de procedencia, alumnos y profesores. |
| FDICONSECUTIVOS | Hace referencia a la secuencia numérica correspondiente a la parte matricula. |
| FDILOGS | Hace referencia a todos los movimientos que se realizan en la aplic |
| FDIUSUARIOS | Hace referencia a las personas que van a hacer uso de la aplicación |



3.2.DICCIONARIO DE DATOS MODELO ENTIDAD RELACION DE LAS TABLAS

| Nombre de la tabla: Fdiestudiantes | | | Descripción de la tabla: en esta tabla estarán los datos de los estudiante el departamento | | | |
|---|-----------------|-------|---|------|------|-------------------------------------|
| Nombre largo: Estudiantes | | | | | | |
| Modulo: Dep. académico | | | | | | |
| Tipo: Maestro | | | | | | |
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Estnombre | | Alfanumérico | 20 | No | Nombre de estudiante |
| 2 | Estcedula | | Alfanumérico | 15 | No | Identificación del est |
| 3 | Esttelefono | | Alfanumérico | 10 | No | Teléfono de est |
| 4 | Estdirección | | Alfanumérico | 20 | No | Dirección del est |
| 5 | Estcodigo | PK | Alfanumérico | 6 | No | Código del est |
| 6 | Estapellido | | Alfanumérico | 20 | No | Apellido del est |
| 7 | Estfecha_nac | | Fecha | | No | Fecha de nacimiento del est |
| 8 | Estestado_civil | | Alfanumérico | 10 | No | Estado civil del est |
| 9 | Ciucod | | Alfanumérico | 15 | No | Ciudad de nacimiento del estudiante |
| 10 | Estsexo | | Carácter | 2 | No | Sexo del est |
| 11 | Estmail | | Alfanumerico | 50 | No | Mail del est |

| Nombre de la tabla: Fdiprogramas | | | Descripción de la tabla: en esta tabla estarán todos los programas académicos presente en el FDI | | | |
|---|-------------|-------|---|------|------|--------------------------------|
| Nombre largo: Programas académicos | | | | | | |
| Modulo: Dep. académico | | | | | | |
| Tipo: Maestro | | | | | | |
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Procodigo | Pk | Alfanumérico | 6 | No | Código de la carrera |
| 2 | Pronombre | | Alfanumérico | 30 | No | Nombre de la carrera |
| 3 | Proicfes | | Alfanumérico | 15 | No | Número de icfes |
| 4 | Procreación | | Fecha | | No | Fecha de creación del programa |
| 5 | Produracion | | Alfanumérico | 20 | No | Duración del programa |



Fdi manual del sistema

| Nombre de la tabla: Fdi matriculas Nombre largo: Estudiantes matriculas Modulo: Dep. académico Tipo: Maestro | | | Descripción de la tabla: en esta tabla estarán relacionados todas las matriculas de los alumnos con sus respectivos programas | | | |
|---|----------------|-------|--|------|------|---|
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Matnumero | Pk | Alfanumérico | 10 | No | Consecutivo de la matricula |
| 2 | Matestudiantes | Fk | Alfanumérico | 15 | No | Cedula del estudiante |
| 3 | Matprograma | Fk | Alfanumérico | 6 | No | Código del estudiante |
| 4 | Matperiodo | | Alfanumérico | 10 | No | Período del año |
| 5 | Matcurso | | Alfanumérico | 20 | No | Curso en el que se va encontrar el estudiante |
| 6 | Mataño | | Alfanumérico | 10 | No | Año en el que se matriculo el estudiante |

| Nombre de la tabla: Fdiciudades Nombre largo: Ciudades Modulo: Departamento académico Tipo: parámetro | | | Descripción de la tabla: en esta tabla estarán las diferentes ciudades de Colombia y el mundo | | | |
|--|-----------------|-------|--|------|------|---|
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Ciucod | PK | Alfanumérico | 15 | No | Código de la ciudad |
| 2 | Ciunombre | | Alfanumérico | 25 | No | Nombre de la ciudad |
| 3 | Ciupais | FK | Alfanumérico | 15 | No | País al que pertenece la ciudad |
| 4 | Ciudepartamento | FK | Alfanumérico | 20 | No | Departamento al que pertenece la ciudad |



Fdi manual del sistema

| Nombre de la tabla: Fdiconsecutivo Nombre largo: Consecutivo Modulo: Departamento académico Tipo: parámetro | | | Descripción de la tabla: en esta tabla estarán los consecutivo de la matricula | | | |
|--|--------------|-------|---|----------|------|---------------------------|
| N o | Nombre | Llave | Tipo de datos | Lon g | Nulo | Descripción |
| 1 | Consecutivo1 | PK | Alfanumérico | 10 | No | Consecutivo de estudiante |
| 2 | Consecutivo2 | | Alfanumérico | 10 | No | Consecutivo de matricula |

| Nombre de la tabla: Fdicurso Nombre largo: Curso Modulo: Departamento académico Tipo: Detalle | | | Descripción de la tabla: en esta tabla estarán los diferentes cursos de los diversos programas | | | |
|--|-------------|-------|---|----------|------|---------------------|
| N o | Nombre | Llave | Tipo de datos | Lon g | Nulo | Descripción |
| 1 | Curnombre | PK | Alfanumérico | 10 | No | Nombre del curso |
| 2 | Curprograma | FK | Alfanumérico | 20 | No | Código del programa |
| 3 | Curpensum | FK | Alfanumérico | 20 | No | Código del pensum |

| Nombre de la tabla: Fdidepartamento Nombre largo: Departamento Modulo: Departamento académico Tipo: parámetro | | | Descripción de la tabla: en esta tabla estarán los diferentes departamentos de Colombia y el mundo | | | |
|--|-----------|-------|---|----------|------|-------------------------|
| N o | Nombre | Llave | Tipo de datos | Lon g | Nulo | Descripción |
| 1 | Depcodigo | PK | Alfanumérico | 20 | No | Código del departamento |
| 2 | Depnombre | | Alfanumérico | 20 | No | Nombre del departamento |
| 3 | Deppais | FK | Alfanumérico | 25 | No | Código del país |



Fdi manual del sistema

| Nombre de la tabla: Fdipaises | | Descripción de la tabla: en esta tabla estarán los diferentes del mundo | | | | |
|---------------------------------------|-----------|--|---------------|------|------|-----------------|
| Nombre largo: Países | | | | | | |
| Modulo: Departamento académico | | | | | | |
| Tipo: parámetro | | | | | | |
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Paicodigo | PK | Alfanumérico | 20 | No | Código del país |
| 2 | Painombre | | Alfanumérico | 20 | No | Nombre del país |

| Nombre de la tabla: Fdipensumateria | | Descripción de la tabla: en esta tabla estarán los diferentes pensum relacionados con las materias | | | | |
|--|------------|---|---------------|------|------|-------------------------------|
| Nombre largo: Pensum de la materias | | | | | | |
| Modulo: Departamento académico | | | | | | |
| Tipo: Detalle | | | | | | |
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Codpensum | FK | Alfanumérico | 10 | No | Código del pensum |
| 2 | Codmateria | FK | Alfanumérico | 10 | No | Código de la materia |
| 3 | Hora | | Alfanumérico | 10 | No | Número de horas de la materia |

| Nombre de la tabla: Fdinotas | | Descripción de la tabla: en esta tabla estarán las notas de los estudiante durante un semestre en específico | | | | |
|---------------------------------------|---------------|---|---------------|------|------|---|
| Nombre largo: Notas | | | | | | |
| Modulo: Departamento académico | | | | | | |
| Tipo: Maestro | | | | | | |
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Notestudiante | Fk | Alfanumérico | 20 | No | Código de estudiante |
| 2 | Notprimera | | Flota | | No | Primera nota |
| 3 | Notsegunda | | Flota | | No | Segunda nota |
| 4 | Nottercera | | Flota | | No | Tercera nota |
| 5 | Notmateria | Fk | Alfanumérico | 20 | No | Código de la materia |
| 6 | Notano | | Alfanumérico | 20 | No | Notas de año en el que está el estudiante |
| 7 | Notperiodo | | Alfanumérico | 20 | No | Período en el que está el estudiante |





Fdi manual del sistema

| Nombre de la tabla: Fdimaterias Nombre largo: Materias Modulo: Departamento académico Tipo: Maestro | | | Descripción de la tabla: en esta tabla estarán los códigos y descripciones de las materias que dan los estudiante en los diferentes programas | | | |
|--|-----------|-------|--|------|------|----------------------|
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Matcodigo | Pk | Alfanumérico | 6 | No | Código de la materia |
| 2 | Matnombre | | Alfanumérico | 20 | No | Nombre de la materia |

| Nombre de la tabla: Fdidocentes Nombre largo: profesores Modulo: Departamento académico Tipo: Maestro | | | Descripción de la tabla: en esta tabla estarán los datos de los profesores | | | |
|--|--------------------|-------|---|------|------|-----------------------------------|
| No | Nombre | Llave | Tipo de datos | Long | Nulo | Descripción |
| 1 | Docnombre | | Alfanumérico | 20 | No | Nombre de los profesores |
| 2 | Doccedula | Pk | Alfanumérico | 12 | No | Cedula de los profesores |
| 3 | Docdireccion | | Alfanumérico | 25 | No | Dirección del profesor |
| 4 | Doctelefono | | Alfanumérico | 20 | No | Teléfono del profesor |
| 5 | Docapellidos | | Alfanumérico | 20 | No | Apellido del profesor |
| 6 | Doctitulo | | Alfanumérico | 20 | No | Titulo del profesor |
| 7 | Docespecializacion | | Alfanumérico | 20 | No | Especialización del profesor |
| 8 | Docfecha_nacdate | | Fecha | | No | Fecha de nacimiento del Profesor |
| 9 | Ciucod | | Alfanumérico | 15 | No | Ciudad de nacimiento del Profesor |



Fdi manual del sistema

| Nombre de la tabla: Fdicarga_académica | | Descripción de la tabla: en esta tabla estarán los códigos de las materias relacionados con los respectivos profesores que dictan dichas materias . | | | | |
|---|-------------|--|---------------|----------|------|----------------------|
| Nombre largo: carga_académica | | | | | | |
| Modulo: Departamento académico | | | | | | |
| Tipo: Detalle | | | | | | |
| N o | Nombre | Llave | Tipo de datos | Lon g | Nulo | Descripción |
| 1 | Carmateria | Fk | Alfanumerico | 20 | No | Código de la materia |
| 2 | Cardocente | Fk | Alfanumerico | 15 | No | Cedula del profesor |
| 3 | Carprograma | FK | Alfanumerico | 20 | No | Código del programa |

| Nombre de la tabla: Fdilogs | | Descripción de la tabla: en esta tabla estarán toda la modificaciones que el usuario realice en la base de datos | | | | |
|---------------------------------------|----------------|---|---------------|----------|------|-------------------------|
| Nombre largo: logs | | | | | | |
| Modulo: Departamento académico | | | | | | |
| Tipo: Detalle | | | | | | |
| N o | Nombre | Llave | Tipo de datos | Lon g | Nulo | Descripción |
| 1 | Logusuario | | Alfanumérico | 20 | No | Código del usuario |
| 2 | Logfecha | | Fecha | | No | Fecha de la transacción |
| 3 | Loghora | | Tiempo | | No | Hora de la transacción |
| 4 | Logtransaccion | | Alfanumérico | 20 | No | Movimientos que se hizo |
| 5 | Logtabla | | Alfanumérico | 20 | No | Tabla afectada |

| Nombre de la tabla: Fdiusuarios | | Descripción de la tabla: en esta tabla estarán todos los usuarios que hagan uso de la aplicación | | | | |
|--|-------------|---|---------------|----------|------|------------------------|
| Nombre largo: usuarios | | | | | | |
| Modulo: Departamento académico | | | | | | |
| Tipo: Detalle | | | | | | |
| N o | Nombre | Llave | Tipo de datos | Lon g | Nulo | Descripción |
| 1 | Usucodigo | | Alfanumérico | 32 | No | Código del usuario |
| 2 | Usupassword | | Alfanumérico | 32 | No | Contraseña del usuario |
| 3 | Usurool | | Alfanumérico | 32 | No | Tipo de usuario |



4. SCRIPT DE CREACIÓN

```
# phpMyAdmin MySQL-Dump
# http://phpwizard.net/phpMyAdmin/
#
# Host: localhost Database : FDI

# -----
#
# Table structure for table 'fdicargaacademica'
#

CREATE TABLE fdicargaacademica (
  carmateria varchar(20) DEFAULT '0' NOT NULL,
  cardocente varchar(15) NOT NULL,
  carprograma varchar(20) NOT NULL
);

# -----
#
# Table structure for table 'fdiciudades'
#

CREATE TABLE fdiciudades (
  ciucod varchar(15) DEFAULT '0' NOT NULL,
  ciunombre varchar(25) NOT NULL,
  ciupais varchar(15),
  ciudepartamento varchar(20),
  PRIMARY KEY (ciucod)
);

# -----
#
# Table structure for table 'fdiconsecutivo'
#

CREATE TABLE fdiconsecutivo (
  consecutivo1 varchar(10) DEFAULT '0' NOT NULL,
  consecutivo2 varchar(10) NOT NULL,
  PRIMARY KEY (consecutivo1)
);

# -----
```



Fdi manual del sistema

```
#  
# Table structure for table 'fdicursos'  
#  
  
CREATE TABLE fdicursos (  
  curnombre varchar(10) DEFAULT '0',  
  curprograma varchar(20),  
  curpensum varchar(20)  
);  
  
# -----  
#  
# Table structure for table 'fdidepartamentos'  
#  
  
CREATE TABLE fdidepartamentos (  
  depcodigo varchar(20) DEFAULT '0',  
  depnombre varchar(20),  
  deppais varchar(25)  
);  
  
# -----  
#  
# Table structure for table 'fdidocentes'  
#  
  
CREATE TABLE fdidocentes (  
  doccedula varchar(12) DEFAULT '0',  
  docnombre varchar(20) NOT NULL,  
  docapellidos varchar(20),  
  doctelefono varchar(20) NOT NULL,  
  docdireccion varchar(25) NOT NULL,  
  doctitulo varchar(20) NOT NULL,  
  docespecializacion varchar(20) NOT NULL,  
  docfecha_nac date DEFAULT '0000-00-00' NOT NULL,  
  ciucod varchar(15) NOT NULL  
);  
  
# -----  
#  
# Table structure for table 'fdiestudiantes'  
#  
  
CREATE TABLE fdiestudiantes (  
  estcodigo varchar(6) NOT NULL,
```



Fdi manual del sistema

```
estcedula varchar(15) NOT NULL,  
estnombre varchar(20) DEFAULT '0',  
estapellidos varchar(20),  
esttelefono varchar(10),  
estdireccion varchar(20),  
ciucod varchar(15) NOT NULL,  
estfecha_nac date DEFAULT '2000-00-00',  
estestado_civil varchar(10),  
estsexo char(2),  
estmail varchar(50),  
PRIMARY KEY (estcodigo)  
);
```

```
# -----  
#  
# Table structure for table 'fdilogs'  
#
```

```
CREATE TABLE fdilogs (  
  logusuario varchar(20) NOT NULL,  
  logfecha date DEFAULT '0000-00-00' NOT NULL,  
  loghora time DEFAULT '00:00:00' NOT NULL,  
  logtransaccion varchar(20) NOT NULL,  
  logtabla varchar(20) NOT NULL  
);
```

```
# -----  
#  
# Table structure for table 'fdimaterias'  
#
```

```
CREATE TABLE fdimaterias (  
  matcodigo varchar(6) DEFAULT '0' NOT NULL,  
  matnombre varchar(20) NOT NULL,  
  PRIMARY KEY (matcodigo)  
);
```

```
# -----  
#  
# Table structure for table 'fdimatriculas'  
#
```

```
CREATE TABLE fdimatriculas (  
  matnumero varchar(10) NOT NULL,  
  matestudiantes varchar(15) NOT NULL,
```



Fdi manual del sistema

```
matprograma varchar(6) NOT NULL,  
matperiodo varchar(10) NOT NULL,  
matcurso varchar(20) NOT NULL,  
mataño varchar(10) NOT NULL  
);  
  
# -----  
#  
# Table structure for table 'fdinotas'  
#  
  
CREATE TABLE fdinotas (  
  notano varchar(20) NOT NULL,  
  notperiodo varchar(20) NOT NULL,  
  notestudiante varchar(20) NOT NULL,  
  notmateria varchar(20) NOT NULL,  
  notprimera float DEFAULT '0' NOT NULL,  
  notsegunda float DEFAULT '0' NOT NULL,  
  nottercera float DEFAULT '0' NOT NULL  
);  
  
# -----  
#  
# Table structure for table 'fdipaises'  
#  
  
CREATE TABLE fdipaises (  
  paicodigo varchar(20) DEFAULT '0',  
  painombre varchar(20)  
);  
  
# -----  
#  
# Table structure for table 'fdipensumateria'  
#  
  
CREATE TABLE fdipensumateria (  
  codpensum varchar(10) DEFAULT '0',  
  codmateria varchar(10),  
  horas varchar(10)  
);  
  
# -----  
#
```



Fdi manual del sistema

```
# Table structure for table 'fdipensums'  
#
```

```
CREATE TABLE fdipensums (  
  penprograma varchar(6) DEFAULT '0' NOT NULL,  
  pencodigo varchar(20),  
  pensemestre varchar(20)  
);
```

```
# -----  
#  
# Table structure for table 'fdiprogramas'  
#
```

```
CREATE TABLE fdiprogramas (  
  procodigo varchar(6) DEFAULT '0' NOT NULL,  
  pronombre varchar(30),  
  proicfes varchar(15) NOT NULL,  
  procreacion date DEFAULT '0000-00-00' NOT NULL,  
  produracion varchar(20),  
  PRIMARY KEY (procodigo)  
);
```

```
# -----  
#  
# Table structure for table 'fdiusuarios'  
#
```

```
CREATE TABLE fdiusuarios (  
  usucodigo varchar(32),  
  usupassword varchar(32),  
  usurool varchar(32)  
);
```

5. DESCRIPCIÓN PROCEDIMIENTOS DEL SISTEMA

5.1. MODIFICACIÓN DE ESTUDIANTES



En la construcción de este formulario se hizo uso de HTML, PHP, SQL y JAVASCRIPT que nos ayudaron a darle una buena apariencia al diseño y a permitir manejar datos almacenados en una Base de Datos.

En este formulario se hicieron validaciones con JAVASCRIPT para que el usuario confirme algún movimiento como cuando va a borrar un determinado registro o va a digitar los valores correspondiente a cada campo del formulario y no los deje vacío por ejemplo en el campo nombre sólo se puede digitar letras porque si coloca números le saldrá un mensaje de error hecho en JAVASCRIPT sugiriéndole que coloque letras para poder avanzar. Unas de las funciones que aparece a continuación son usadas para validar los campos alfabético:

```
function isAlphabetic (s,p)
{
  var i;
  for (i = 0; i < s.length; i++)
  {
    // Verifica que el actual caracter sea letra.
    var c = s.charAt(i);

    if (!isLetter(c)){
      alert("El campo solo permite letras")
      document.forms[0].elements[p].focus()
      return false;
    }
  }
}
```



Fdi manual del sistema

```
return true;
}

function isLetter (c)
{
var uppercaseLetters = "ABCDEFGHIJKLMNOPQRSTUVWXYZÁÉÍÓÚÑ .-,"
var lowercaseLetters = "abcdefghijklmnopqrstuvwxyzáéíóúñü"

return( ( uppercaseLetters.indexOf( c ) != -1 ) ||
( lowercaseLetters.indexOf( c ) != -1 ) )
}
}
```

Esta forma contiene 10 campos que son: cedula, nombre, apellidos teléfono, dirección, ciudad, fecha_nac, estada_civil, sexo y e-mail, todos estos campos son llenados por el usuario final a excepción de código de estudiante que no aparece en el formulario debido a que es generado automáticamente por la aplicación a través de las siguientes funciones de SQL:

```
$consulta="select consecutivo2 from FDIconsecutivo ";
$resultado = mysql_db_query("fdi",$consulta);
$row = mysql_fetch_row($resultado) ;
$codigo=$row[0]+1;

$query="update FDIconsecutivo set consecutivo2=consecutivo2+1";
mysql_db_query ("fdi",$query);
```

Como podemos ver en las funciones anteriores el consecutivo es manejado por una tabla diferente a FDIESTUDIANTES (tabla que contiene todos los datos de los estudiantes), porque de esta manera se tiene una mayor organización y efectividad en cuanto a la secuencia correspondiente al código del estudiante, además los campos estado_civil, sexo, y ciudad son de selección es decir en donde el usuario sólo colocará lo que este disponible en esa lista pero en el campo ciudad hacemos uso de SQL porque los datos en ese campo se tienen que cargar automáticamente de acuerdo con las ciudades existentes en FDI CIUDADES (tabla que contiene todas la ciudades registradas por el coordinador académico).

En esta parte de la aplicación se manejan inserciones, borrados, actualizaciones y consulta de los datos a través de SQL, cuando se va a realizar movimientos como:

- **Insertar:** El usuario final digita todos los datos correspondiente al estudiante a ingresar y presiona el botón **Insertar** para que se realice la inserción de los datos a la Base de Datos y se genere automáticamente el código de estudiante por medio de las funciones SQL (este funciones se encuentra en el párrafo anterior).
- **Borrar:** Con este link coordinador borra el registro del estudiante seleccionado no solamente de la tabla FDIESTUDIANTES (tabla que contiene todos los datos de los estudiantes) sino también de la tabla FDIUSUARIOS (tabla que contiene todos los usuarios que hacen uso de la aplicación).
- **Consultar:** Con este botón se puede consultar a todos los estudiantes o a uno en específico por ejemplo si se quiere ver todos los estudiantes que se llamen Alejandro sólo tiene que colocar este nombre en el campo correspondiente y por medio del siguiente código podrá ver la consulta que requirió:



Fdi manual del sistema

```
if(!empty ($nombre)){
  if(empty($sql)){
    $sql="SELECT FDlestudiantes. *,
      FDlciudades.ciunombre
      FROM FDlestudiantes,
      FDlciudades
      where FDlciudades.ciucod=FDlestudiantes.ciucod and
      FDlestudiantes.estnombre='$nombre'";
  }else{
    $sql=$sql." and FDlestudiantes.estnombre='$nombre'";
  }
}
```

Para que se vean los datos consultados más organizados se hizo uso del siguiente código que nos permite paginar los datos y la vez los introduce dentro de una tabla:

```
$result = mysql_db_query("fdi", $sql);
$num_record=mysql_num_rows($result);
$num_page=ceil($num_record/8);

echo "<center><h4>CONSULTA DE ESTUDIANTES</h4></center>";
echo "<table align='center' border=2 bgcolor='white' bordercolor='gray'> \n";
echo "<tr>
  <td><center><h6>Codigo</h6></center></td>
  <td><center><h6>Cedula</h6></center></td>
  <td><center><h6>Nombre</h6></center></td>
  <td><center><h6>Apellidos</h6></center></td>
  <td><center><h6>Telefono</h6></center></td>
  <td><center><h6>Direccion</h6></center></td>
  <td><center><h6>Ciudad</h6></center></td>
  <td><center><h6>Fecha_Nac</h6></center></td>
  <td><center><h6>Estado_civil</h6></center></td>
  <td><center><h6>Sexo</h6></center></td>
  <td><center><h6>Correo</h6></center></td>
  <td colspan='2'><center><h6>Opciones</h6></center></td>
</tr>\n";
$contador=1;
while ($row = mysql_fetch_row($result)){
  if(isset($min)){
    if(($contador>=$min)&&($contador<=$max)){
      echo "<tr>
        <td>$row[0]</td>
        <td>$row[1]</td>
        <td>$row[2]</td>
        <td>$row[3]</td>
        <td>$row[4]</td>
        <td>$row[5]</td>
        <td>$row[11]</td>
```



Fdi manual del sistema

tabla FDILOGS (tabla que contiene todos los movimientos que realicen los usuarios en la aplicación) después cualquier movimiento que afecta a la Base de Datos, esta tabla contiene los siguientes campos:

Logusuario
Logfecha
Loghora
Logtransacción
Logtabla

A través de esta tabla el administrador sabrá que hizo el usuario X, el día Y, a la hora Z, el moviendo que realizó en la Base de Datos y sobre que tabla lo efectuó

5.2. MODIFICACIÓN DE DOCENTES



En la elaboración de este formulario se hizo uso de HTML, PHP, SQL y JAVASCRIPT que nos ayudaron a darle una buena apariencia al diseño y a permitir manejar datos almacenados en una Base de Datos.

En este formulario de hicieron validaciones con JAVASCRIPT para que el usuario confirme algún movimiento como cuando va a borrar un determinado registro o va a digitar algo en los campos del formulario y no los deje vacío por ejemplo en el campo cedula aquí el usuario sólo puede ingresar números por que de lo contrario le saldrá un mensaje de error hecho en JAVASCRIPT que le dice que digite números solamente. Unas de las funciones que aparece a continuación son usadas para validar los campos numéricos.

```
// tiene solo numeros
function isNumeric (s,p)
{ var i;
  for (i = 0; i < s.length; i++)
  {
    // Verifica que el actual caracter sea numero.
    var c = s.charAt(i);
```



Fdi manual del sistema

```
    if (!isDigit(c)){
        alert("El campo solo permite numeros.")
        document.forms[0].elements[p].focus()
        return false;
    }
}
return true;
}

function isDigit (c)
{
    var Digits = "0123456789"

    return( ( Digits.indexOf( c ) != -1 ) )
}
}
```

Esta forma contiene 9 campos que son : cedula, nombre, apellidos teléfono, dirección, título, especialización, fecha_nac, ciudad. todos estos campos son llenados por el usuario final pero el campo ciudad hacemos uso de SQL porque los datos en ese campo se tienen que cargar automáticamente de acuerdo con las ciudades existentes en FDICIUDADES (tabla que contiene todas la ciudades registradas por el coordinador académico).

En esta parte de la aplicación se manejan inserciones, borrados, actualizaciones, carga académica y consulta de los datos a través de SQL, cuando se va a realizar movimientos como:

- **Insertar:** El usuario final digita todos los datos correspondiente al docente a ingresar y presiona el botón **Insertar** para que se almacenen todos los datos en la Base de Datos, pero antes de esto se realiza una verificación que nos dice si el docente ya existe, si este no se encuentra en la Base de Datos se almacena sin ningún problema si no observará un mensaje de advertencia comunicándole que dicho docente ya existe, el código en SQL que nos permite verificar si un docente existe o no es el siguiente:

```
$query="select* from FDIdocentes";
$ptr = mysql_db_query("fdi", $query);
while ($row = @mysql_fetch_row($ptr)){
    if ($row[0] == $cedula) {
        $sw=1;
    }
}
}
```

- **Borrar:** Con este link coordinador borra el registro de un profesor seleccionado no solamente de la tabla FDIDOCENTES (tabla que contiene todos los datos de los docentes) sino también de la tabla FDIUSUARIOS (tabla que contiene todos los usuarios que hacen uso de la aplicación) y FDICARGAACADEMICA (tabla que contiene todas las materias que dicta un profesor).
- **Consultar:** Con este botón se puede consultar a todos los docentes o a uno en específico por ejemplo si se quiere ver todos los profesores que sean abogados sólo tiene que colocar el nombre de la carrera en el campo título y por medio del siguiente código podrá ver la consulta que requirió:

```
if(!empty ($titulo)){
```



Fdi manual del sistema

```
if(empty($sql)){
    $sql="SELECT FDIdocentes.*
        FDIciudades.ciunombre
        FROM FDIdocentes,
        FDIciudades
        where FDIdocentes.ciucod=FDIciudades.ciucod and
        FDIdocentes.doctitulo='$titulo';
    }else{
        $sql=$sql." and FDIdocentes.doctitulo='$titulo';
    }
}
```

Para que se vean los datos consultados más organizados se hizo uso del siguiente código que nos permite paginar los datos y la vez los introduce dentro de una tabla:

```
$result = mysql_db_query("fdi", $sql);
$num_record=mysql_num_rows($result);
$num_page=ceil($num_record/8);

echo"<center><h4>CONSULTA DE DOCENTES</h4></center>";
echo "<table align='center' border=2 bgcolor='white' bordercolor='gray'> \n";
echo "<tr>
    <td><center><h6>Cedula</h6></center></td>
    <td><center><h6>Nombre</h6></center></td>
    <td><center><h6>Apellidos</h6></center></td>
    <td><center><h6>Telefono</h6></center></td>
    <td><center><h6>Direccion</h6></center></td>
    <td><center><h6>Titulo</h6></center></td>
    <td><center><h6>Especializacion</h6></center></td>
    <td><center><h6>Fecha_Nac</h6></center></td>
    <td><center><h6>Ciudad</h6></center></td>
    <td colspan='2'><center><h6>Opciones</h6></center></td>
</tr>\n";
$contador=1;
while ($row = @mysql_fetch_row($result)){
    if(isset($min)){

        if(($contador>=$min)&&($contador<=$max)){

            echo "<tr>
                <td>$row[0]</td>
                <td>$row[1]</td>
                <td>$row[2]</td>
                <td>$row[3]</td>
                <td>$row[4]</td>
                <td>$row[5]</td>
                <td>$row[6]</td>
                <td>$row[7]</td>
                <td>$row[9]</td>
```



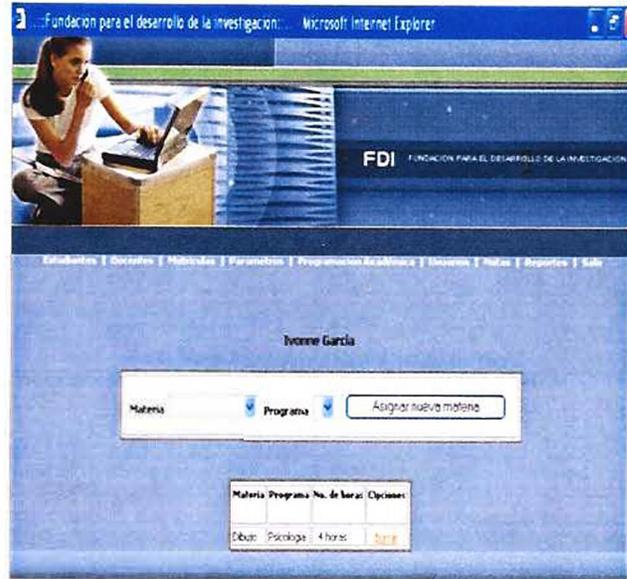
```

<td><a href='docentes.php?cedula=$row[0]&
        nombre=$row[1]&
        apellidos=$row[2]&
        telefono=$row[3]&
        direccion=$row[4]&
        titulo=$row[5]&
        especializacion=$row[6]&
        fechana=$row[7]&
        ciudad=$row[8]&
        boton=Editar'>Editar</a></td>
<td><a href='docentes.php?boton=Borrar&cedula=$row[0]'
onClic="return valeliminar();" <?php echo " >Borrar</a></td></tr> \n";
    }
} $contador++;
}
echo "</table> \n";
echo "<center>";
for($i=1;$i<=$num_page;$i++){
    if($i==1){
        $min=1;
        $max=8;
    }
    else{
        $min=$min+8;
        $max=$max+8;
    }
    echo
    "|<a
href=docentes.php?boton=$boton&cedula=$cedula&nombre=$nombre&apellido
s=$apellidos&telefono=$telefono&direccion=$direccion&titulo=$titulo&especializ
acion=$especializacion&fechana=$fechana&ciudad=$ciudad&min=$min&max=
$max>$i</a>|";
}

```

Con este botón no sólo podemos visualizar los datos sino que podemos cargar la información de un determinado docente al formulario para que se le realicen las actualizaciones, carga de materia o seleccionar al profesor que se va a eliminar.

- **Actualizar:** Aquí con sólo darle click al botón **Actualizar** se hace una renovación de los datos almacenados en la Base de Datos a través de la función update de SQL que es la encargada de remplazar todos aquellos datos viejos que tenía el usuario por los nuevos que acaba de digitar.
- **Carga Académica:** Presionando este botón nos transporta a una nueva parte de la aplicación (en la parte inferior se encuentra una imagen de la parte de carga académica) en donde se le asignan todas la materias a los profesores existentes en la institución a través de un formulario, a igual que en las otras formas se realiza validaciones con JAVASCRIPT para que el usuario confirme algún movimiento como cuando va a borrar un determinado registro o deja el campo de selección de materias o programas vacío e intenta realizar la asignación de una materia.



Este formulario cuenta con dos campos de selección que son materia y programas, lo cual quiere decir que el coordinador deberá escoger los datos que se visualicen en la lista desplegable que le aparece a la hora de hacer un asignación, la información que se observa en dicha lista es captura de la tablas `FDIMATERIAS` (tabla donde se encuentra todas las materias de los diverso programas) y `FDIPROGRAMAS` (tabla donde se encuentran las diferentes carreras), estas tablas deben tener datos porque de lo contrario no se podrá hacer ninguna selección, la anteriormente mencionado se realiza a través del siguiente código:

```

&nbsp;&nbsp; <b>Materia</b>&nbsp;&nbsp;&nbsp;";
<select name='materia' onchange="cambia_programa();">
<?
echo "<option></option>";
$query = mysql_query("SELECT * FROM FDImaterias", $conexion);
while ($row = mysql_fetch_row($query)){
if($row[0]==$materia) echo "<option value=" . $row[0].
selected>". $row[1]. "</OPTION>";
else
    echo "<option value=" . $row[0]. ">". $row[1]. "</OPTION>";
}
echo "</select>

```

```

&nbsp;&nbsp;&nbsp;<b>Programa</b>&nbsp;&nbsp;&nbsp;
<select name='programa'>
<option></option> ";
$query = mysql_query("SELECT FDIprogramas.procodigo,
FDIprogramas.pronombre FROM FDIpensumateria, FDIprogramas,
FDIpensums where FDIpensumateria.codmateria=$materia and

```



Fdi manual del sistema

```
FDIpensumateria.codpensum=FDIpensums.pencodigo and
FDIpensums.penprograma=FDIprogramas.procodigo", $conexion);
    while ($row = mysql_fetch_row($query)){

        echo "<option value=".$row[0].">".$row[1]."</OPTION>";
    }
echo "</select>";
```

?>

Para la asignación de una materia a un profesor primero se verifica que dicha asignatura no la este dictando dos veces en mismo semestres de un programa y después la inserta, esto se realiza por medio del siguiente código:

```
$query = "select * from FDIcargaacademica where carmateria='$materia' and
cardocente=$cedula and carprograma=$programa";
$ptr = mysql_db_query("fdi", $query);
$filas= mysql_num_rows($ptr);
if ($filas != NULL)
    echo "<script language='javascript'>alert('La materia seleccionada ya
existe');</script>";
else
    {
    $sql = "insert into FDIcargaacademica (carmateria, cardocente, carprograma)
        values ('$materia', '$cedula', '$programa)";
    $result = mysql_db_query("fdi", $sql);
```

Las materias asignadas son mostradas en un tabla en la parte inferior del formulario.

En este formulario también se borrar las materias asignadas a un docente que se encuentra almacenada en la tabla FDICARGAACADEMICA a través de la función delete de SQL,

Para el administrador de una Base de Datos es muy importante saber que hacen los usuarios con la información que es usada en la aplicación y es por eso que se realizan inserciones en la tabla FDILOGS (tabla que contiene todos los movimientos que realicen los usuarios en la aplicación) después cualquier movimiento que afecta a la Base de Datos, esta tabla contiene los siguientes campos:

- Logusuario
- Logfecha
- Loghora
- Logtransacción
- Logtabla

A través de esta tabla el administrador sabrá que hizo el usuario X, el día Y, a la hora Z, el moviendo que realizó en la Base de Datos y sobre que tabla lo efectuó

5.3. MODIFICACIÓN DE MATRICULAS



Para hacer este formulario se hizo uso de HTML, PHP, SQL y JAVASCRIPT que nos ayudaron a darle una buena apariencia al diseño y a permitir manejar datos almacenados en una Base de Datos.

En este formulario se hicieron validaciones con JAVASCRIPT para que el usuario confirme algún movimiento como cuando va a borrar un determinado registro o va a digitar los valores correspondiente a cada campo del formulario y no los deje vacío por ejemplo en el campo cod.estudiante sólo se puede digitar número porque si coloca letras le saldrá un mensaje de error hecho en JAVASCRIPT sugiriéndole que coloque números para poder avanzar. Unas de las funciones que aparece a continuación son usadas para validar los campos numéricos:

// s tiene solo numeros

```
function isNumeric (s,p)
```

```
{ var i;
```

```
  for (i = 0; i < s.length; i++)
```

```
  {
```

```
    // Verifica que el actual caracter sea numero.
```

```
    var c = s.charAt(i);
```

```
    if (!isDigit(c)){
```

```
      alert("El campo solo permite numeros.")
```

```
      document.forms[0].elements[p].focus()
```

```
      return false;
```

```
    }
```

```
  }
```

```
  return true;
```

```
}
```

```
function isDigit (c)
```



Fdi manual del sistema

```
{  
    var Digits = "0123456789"  
  
    return( ( Digits.indexOf( c ) != -1 ) )  
  
}
```

Este formulario contiene 5 campos que son: cod. estudiante, programa , curso, período y año, los campos cod.estudiante y año son los únicos en donde el coordinador tiene que tipiar la información pero el campo años tiene algo en particular que le toma por default el año actual sin embargo si el usuario requiere de otro año allí si tiene que digitarlo, el resto de los campos son de selección en donde la información se extrae de las tablas FDIPROGRAMAS (tabla donde se encuentran las diferentes carreras) y FDI CURSOS (tabla en que almacena todos los cursos de los diversos programas) por medio del siguiente código:

Para programa.

```
$result = mysql_query("SELECT * FROM FDIprogramas", $conexion);  
while ($row = mysql_fetch_row($result)){  
if($row[0]==$programa) echo "<option value=".$row[0]."  
selected>".$row[1]."</OPTION>";  
else  
    echo "<option value=".$row[0].">".$row[1]."</OPTION>";  
}
```

Para curso.

```
$result = mysql_query("SELECT * FROM FDIcursos where curprograma=$programa",  
$conexion);  
while ($row = mysql_fetch_row($result)){  
if($row[0]==$curso) echo "<option value=".$row[0]." selected>".$row[0]."</OPTION>";  
else  
    echo "<option value=".$row[0].">".$row[0]."</OPTION>";  
}
```

El campo período actúa estáticamente por dicha razón no se hace ninguna consulta a la Base de Datos. En este formulario el campo código de matrícula no aparece en el formulario porque que es generado automáticamente por la aplicación a través de las siguientes funciones de SQL:

```
$consulta="select consecutivo2 from FDIconsecutivo " ;  
$resultado = mysql_db_query("fdi",$consulta);  
$row = mysql_fetch_row($resultado) ;  
$codigo=$row[0]+1;  
  
$query="update FDIconsecutivo set consecutivo2=consecutivo2+1";  
mysql_db_query ("fdi",$query);
```

Como podemos ver en las funciones anteriores el consecutivo es manejado por una tabla diferente a FDI MATRICULAS (tabla que contiene todos los datos de las matrículas), porque de esta manera se tiene una mayor organización y efectividad en cuanto a la secuencia correspondiente al código de la matrícula

En esta parte de la aplicación se manejan inserciones, borrados, actualizaciones y consulta de los datos a través de SQL, cuando se va a realizar movimientos como:



Fdi manual del sistema

- **Insertar:** El usuario final digita todos los datos correspondiente a la matricula que se va a ingresar y presiona el botón **Insertar** para que se realice la inserción de los datos a la Base de Datos y se genere automáticamente el código de la matricula por medio de las funciones SQL (este funciones se encuentra en el párrafo anterior). Antes de hacer la inserción hay que tener presente que se realizan validaciones en donde se verifica que:
 - si un estudiante está matriculado en un misma carrera, programa, año , período y curso, esto se comprueba a través de este código:

```
//Verifico si se repite el registro
$query="select* from FDImatriculas where matprograma='$programa'and
matestudiantes='$codigo'and matperiodo='$periodo'and
matcurso='$curso'and matano='$ano'";
$ptr = mysql_db_query("fdi",$query);
$filas= @mysql_num_rows($ptr);
```

- Para poder matricular un estudiante previamente este debe encontrarse registrado en la taba FDIESTUDIATES esto se hace por medio del siguiente código:

```
//Verifico si el estudiante existe
$query="select estcodigo from FDIestudiantes where
estcodigo='$codigo'";
$ptr = mysql_db_query("fdi",$query);
$filas= mysql_num_rows($ptr);
```

Si el usuario no existe le saldrá un mensaje de advertencia diciéndole que el estudiante no existe.

Al momento de matricular a un estudiante automáticamente se esta registrando en la tabla FDINOTAS (tabla donde se encuentra todas las notas con sus respectivas materias) todas aquellas materias que el estudiante perdió y las va a ver en el semestre en el que se matriculo.

- **Borrar:** Con este link coordinador borra el registro de la matricula seleccionada de la tabla FDI MATRICULAS (tabla que contiene toda la información concernientes a matricula).
- **Consultar:** Con este botón se puede consultar a todos las matriculas o a una en específico por ejemplo si se quiere ver todos las matriculas del año 2005 sólo tiene que colocar el año en el campo correspondiente y por medio del siguiente código podrá ver la consulta que requirió:

```
if(!empty ($ano)){
  if(empty($sql)){
    $sql="SELECT FDImatriculas.*,
              FDIprogramas.pronombre
            FROM FDImatriculas,
              FDIprogramas
            where FDImatriculas.matprograma=FDIprogramas.procodigo and
              FDImatriculas.matano='$ano'";
  }else{
```



Fdi manual del sistema

```

    $sql=$sql." and FDImatriculas.matano='$ano'";
}

```

Para que se vean los datos consultados más organizados se hizo uso del siguiente código que nos permite paginar los datos y la vez los introduce dentro de una tabla:

```

$result = mysql_db_query("fdi",$sql);
$num_record=mysql_num_rows($result);
$num_page=ceil($num_record/8);

echo "<center><h4>CONSULTA DE MATRICULAS</h4></center>";
echo "<table align='center' border=2 bgcolor='white' bordercolor='gray'> \n";
echo "<tr>
    <td><center><h6>Numero</h6></center></td>
    <td><center><h6>Codigo</h6></center></td>
    <td><center><h6>Programa</h6></center></td>
    <td><center><h6>Curso</h6></center></td>
    <td><center><h6>Periodo</h6></center></td>
    <td><center><h6>Año</h6></center></td>
    <td colspan='2'><center><h6>Opciones</h6></center></td>

    </tr>\n";
$contador=1;
while ($row = @mysql_fetch_row($result)){
    if(isset($min)){
        if(($contador>=$min)&&($contador<=$max)){
            echo "<tr>
                <td>$row[0]</td>
                <td>$row[1]</td>
                <td>$row[6]</td>
                <td>$row[4]</td>
                <td>$row[3]</td>
                <td>$row[5]</td>

                <td><a href='matriculas.php?numero=$row[0]&
                    codigo=$row[1]&
                    programa=$row[2]&
                    periodo=$row[3]&
                    curso=$row[4]&
                    ano=$row[5]'>Editar</a></td>

                <td><a href='matriculas.php?boton=Borrar&numero=$row[0]' ";
            ?> onClick="return valeliminar();" <?php echo " >Borrar</a></td></tr> \n";
        }
    }
    $contador++;
}
}

```





```
$sql=$sql." and FDImatriculas.matano='$ano'";
}
```

Para que se vean los datos consultados más organizados se hizo uso del siguiente código que nos permite paginar los datos y la vez los introduce dentro de una tabla:

```
$result = mysql_db_query("fdi",$sql);
$num_record=mysql_num_rows($result);
$num_page=ceil($num_record/8);

echo"<center><h4>CONSULTA DE MATRICULAS</h4></center>";
echo "<table align='center' border=2 bgcolor='white' bordercolor='gray'> \n";
echo "<tr>
    <td><center><h6>Numero</h6></center></td>
    <td><center><h6>Codigo</h6></center></td>
    <td><center><h6>Programa</h6></center></td>
    <td><center><h6>Curso</h6></center></td>
    <td><center><h6>Periodo</h6></center></td>
    <td><center><h6>Año</h6></center></td>
    <td colspan='2'><center><h6>Opciones</h6></center></td>

    </tr>\n";
$contador=1;
while ($row = @mysql_fetch_row($result)){
    if(isset($min)){
        if(($contador>=$min)&&($contador<=$max)){
            echo "<tr>
                <td>$row[0]</td>
                <td>$row[1]</td>
                <td>$row[6]</td>
                <td>$row[4]</td>
                <td>$row[3]</td>
                <td>$row[5]</td>

                <td><a href='matriculas.php?numero=$row[0]&
                    codigo=$row[1]&
                    programa=$row[2]&
                    periodo=$row[3]&
                    curso=$row[4]&
                    ano=$row[5]'>Editar</a></td>

                <td><a href='matriculas.php?boton=Borrar&numero=$row[0]' ";
?> onClick="return valeliminar();" <?php echo" >Borrar</a></td></tr> \n";
            }
        }
        $contador++;
    }
}
```



Fdi manual del sistema

```
echo "</table> \n";
for($i=1;$i<=$num_page;$i++){
    if($i==1){
        $min=1;
        $max=8;
    }
    else{
        $min=$min+8;
        $max=$max+8;
    }
    echo "|\<a
href=matriculas.php?boton=$boton&numero=$numero&codigo=$codigo&progra
ma=$programa&periodo=$periodo&curso=$curso&ano=$ano&min=$min&max=
$max>$i</a>|";
}
echo "<br><br><br><br><br>";
?>
```

Con este botón no sólo podemos visualizar los datos sino que podemos cargar la información de una determinada matricula al formulario para que se le realicen las actualizaciones que el usuario final le desee hacer y seleccionar la matricula que se va a eliminar.

- **Actualizar:** Aquí con sólo darle click al botón **Actualizar** se hace una renovación de los datos almacenados en la Base de Datos a través de la función update de SQL que es la encargada de remplazar todos aquellos datos viejos que tenia el usuario por los nuevos que acaba de digitar.

Para el administrador de una Base de Datos es muy importante saber que hacen los usuarios con la información que es usada en la aplicación y es por eso que se realizan inserciones en la tabla FDILOGS (tabla que contiene todos los movimientos que realicen los usuarios en la aplicación) después cualquier movimiento que afecta a la Base de Datos, esta tabla contiene los siguientes campos:

Logusuario
Logfecha
Loghora
Logtransacción
Logtabla

A través de esta tabla el administrador sabrá que hizo el usuario X, el día Y, a la hora Z, el moviendo que realizó en la Base de Datos y sobre que tabla lo efectuó

5.4. MODIFICACIÓN DE PAISES



En la elaboración de este formulario se hizo uso de HTML, PHP, SQL y JAVASCRIPT que nos ayudaron a darle una buena apariencia al diseño y a permitir manejar datos almacenados en una Base de Datos.

En este formulario se hicieron validaciones con JAVASCRIPT para que el usuario confirme algún movimiento como cuando va a borrar un determinado registro o va a digitar los valores correspondiente a cada campo del formulario y no los deje vacío por ejemplo en el campo nombre sólo se puede digitar letras porque si coloca números le saldrá un mensaje de error hecho en JAVASCRIPT sugiriéndole que coloque letras para poder avanzar. Unas de las funciones que aparece a continuación son usadas para validar los campos alfabético:

```
function isAlphabetic (s,p)
{
  var i;
  for (i = 0; i < s.length; i++)
  {
    // Verifica que el actual caracter sea letra.
    var c = s.charAt(i);

    if (!isLetter(c)){
      alert("El campo solo permite letras")
      document.forms[0].elements[p].focus()
      return false;
    }
  }
  return true;
}

function isLetter (c)
```



Fdi manual del sistema

```
{  
  var uppercaseLetters = "ABCDEFGHIJKLMNOPQRSTUVWXYZÁÉÍÓÚÑ .-,,"  
  var lowercaseLetters = "abcdefghijklmnopqrstuvwxyzáéíóúñü"  
  
  return( ( uppercaseLetters.indexOf( c ) != -1 ) ||  
          ( lowercaseLetters.indexOf( c ) != -1 ) )  
}
```

Este formulario solo cuenta con 2 campos de texto que son código y nombre los cuales son utilizados para manejar toda la información de los países.

En esta parte de la aplicación se manejan inserciones, borrados, actualizaciones y consulta de los datos a través de SQL, cuando se va a realizar movimientos como:

- **Insertar:** El usuario final digita todos los datos correspondiente al país a ingresar y presiona el botón **Insertar** para que se realice la inserción de los datos a la tabla FDIPAISES siempre y cuando el país a ingresar no exista, esta validación se realiza con este código:

```
$query="select* from FDIPAISES";  
$ptr = mysql_db_query("fdi",$query);  
while ($row = @mysql_fetch_row($ptr)){  
  if ($row[0] == $codigo) {  
    $sw=1;  
  }  
}
```

Si el país ya existe el usuario final visualizará un mensaje de advertencia comunicándole que ya ese país esta registrado

- **Borrar:** Con este link coordinador borra el registro del país que halla seleccionado de la tabla FDIPAISES (tabla que contiene todos los países de procedencia de estudiante y profesores).
- **Consultar:** Con este botón se puede consultar a todos los países o a uno en específico por ejemplo si se quiere ver Colombia sólo tiene que colocar este nombre en el campo correspondiente y por medio del siguiente código podrá ver la consulta que requirió:

```
if(!empty($nombre)){  
  if(empty($sql)){  
    $sql="SELECT * FROM FDIPAISES where painombre='$nombre'";  
  }else{  
    $sql=$sql." and painombre='$nombre'";  
  }  
}
```

Para que se vean los datos consultados más organizados se hizo uso del siguiente código que nos permite paginar los datos y la vez los introduce dentro de una tabla:

```
$result = mysql_db_query("fdi",$sql);  
$num_record=mysql_num_rows($result);  
$num_page=ceil($num_record/8);
```

```
echo"<center><h4>CONSULTA DE PAISES</h4></center>";
```



Fdi manual del sistema

```

echo "<center>";
echo "<table border=2 bgcolor='white' bordercolor='gray'> \n";
echo "<tr><td><h6>Codigo</h6></td>
      <td><h6>Nombre</h6></td>
      <td colspan='2'><h6><center>Opciones</center></h6></td></tr>\n";

$contador=1;
while ($row = @mysql_fetch_row($result)){
  if(isset($min)){
    if(($contador>=$min)&&($contador<=$max)){
      echo "<tr><td>$row[0]</td>
            <td>$row[1]</td>
            <td><a href='países.php?codigo=$row[0]&
                    nombre=$row[1]&
                    boton=Editar'>Editar</a></td>
            <td><a href='países.php?boton=Borrar&codigo=$row[0]' "; ?> onClick="return
valediminar();" <?php echo " >Borrar</a></td></tr> \n";
    }
  }
  $contador++;
}
echo "</table> \n ";
for($i=1;$i<=$num_page;$i++){
  if($i==1){
    $min=1;
    $max=8;
  }
  else{
    $min=$min+8;
    $max=$max+8;
  }
  echo
  href=países.php?boton=$boton&codigo=$codigo&nombre=$nombre&min=$min&ma
x=$max>$i</a>|";
}

```

```

echo "</center><br><br><br><br><br>";
?>

```

Con este botón no sólo podemos visualizar los datos sino que podemos cargar la información de un determinado país al formulario para que se le realicen las actualizaciones que el usuario final le desee hacer y seleccionar al país que se va a eliminar.

- **Actualizar:** Aquí con sólo darle click al botón **Actualizar** se hace una renovación de los datos almacenados en la Base de Datos a través de la función update de SQL que es la encargada de remplazar todos aquellos datos viejos que tenía el usuario por los nuevos que acaba de digitar.

Para el administrador de una Base de Datos es muy importante saber que hacen los usuarios con la información que es usada en la aplicación y es por eso que se realizan inserciones en la tabla FDILOGS (tabla que contiene todos los movimientos que realicen los usuarios en la

Fdi manual del sistema

aplicación) después cualquier movimiento que afecta a la Base de Datos, esta tabla contiene los siguientes campos:

Logusuario
Logfecha
Loghora
Logtransacción
Logtabla

A través de esta tabla el administrador sabrá que hizo el usuario X, el día Y, a la hora Z, el moviendo que realizó en la Base de Datos y sobre que tabla lo efectuó

5.5. MODIFICACIÓN DE DEPARTAMENTOS



En la construcción de este formulario se hizo uso de HTML, PHP, SQL y JAVASCRIPT que nos ayudaron a darle una buena apariencia al diseño y a permitir manejar datos almacenados en una Base de Datos.

En este formulario se hicieron validaciones con JAVASCRIPT para que el usuario confirme algún movimiento como cuando va a borrar un determinado registro o va a digitar los valores correspondiente a cada campo del formulario y no los deje vacío por ejemplo en el campo nombre sólo se puede digitar letras porque si coloca números le saldrá un mensaje de error hecho en JAVASCRIPT sugiriéndole que coloque letras para poder avanzar. Unas de las funciones que aparece a continuación son usadas para validar los campos alfabético:

```
function isAlphabetic (s,p)
{
  var i;
  for (i = 0; i < s.length; i++)
  {
    // Verifica que el actual caracter sea letra.
    var c = s.charAt(i);

    if (!isLetter(c)){
      alert("El campo solo permite letras")
    }
  }
}
```



Fdi manual del sistema

```
        document.forms[0].elements[p].focus()
        return false;
    }
}
return true;
}

function isLetter (c)
{
    var uppercaseLetters = "ABCDEFGHIJKLMNOPQRSTUVWXYZÁÉÍÓÚÑ .-,"
    var lowercaseLetters = "abcdefghijklmnopqrstuvwxyzáéíóúñü"

    return( ( uppercaseLetters.indexOf( c ) != -1 ) ||
            ( lowercaseLetters.indexOf( c ) != -1 ) )
}
}
```

Este formulario contiene 3 campos que son código, nombre y país en los 2 primeros el coordinador tiene que digitar la información pero en el campo de selección país tiene que escoger el dato de un lista desplegable que extrae esa información de la tabla FDIPAISES (tabla que contiene todos los países de procedencia de profesores y estudiantes), para extraer la información de la tabla el procedimiento que se realizó fue este:

```
<?php
$result = mysql_query("SELECT * FROM FDIpaíses", $conexion);
while ($row = mysql_fetch_row($result)){
if($row[0]==$pais) echo "<option value=".$row[0]." selected>".$row[1]."</OPTION>";
else
        echo "<option value=".$row[0].">".$row[1]."</OPTION>";
}
?>
```

En esta parte de la aplicación se manejan inserciones, borrados, actualizaciones y consulta de los datos a través de SQL, cuando se va a realizar movimientos como:

- **Insertar:** El usuario final digita todos los datos correspondiente al departamento a ingresar y presiona el botón **Insertar** para que se realice la inserción de los datos a la tabla FDIDEPARTAMENTOS siempre y cuando el departamento a ingresar no exista, esta validación se realiza con este código:

```
$query="select* from FDIdepartamentos";
$ptr = mysql_db_query("fdi",$query);
while ($row = @mysql_fetch_row($ptr)){
    if ($row[0] == $codigo) {
        $sw=1;
    }
}
}
```

Si el departamento ya existe el usuario final visualizará un mensaje de advertencia comunicándole que ya ese departamento esta registrado



Fdi manual del sistema

- **Borrar:** Con este link coordinador borra el registro del departamento que halla seleccionado de la taba FDIDEPARTAMENTO(tabla que contiene todos los departamentos de procedencia de estudiante y profesores).
- **Consultar:** Con este botón se puede consultar a todos los departamentos o a uno en específico por ejemplo si se quiere ver el departamento de Atlántico sólo tiene que colocar Atlántico en el campo nombre y por medio del siguiente código podrá ver la consulta que requirió:

```
if(!empty ($nombre)){
  if(empty($sql)){
    $sql="SELECT FDIdepartamentos.*, FDIpaíses.painombre
      FROM FDIdepartamentos, FDIpaíses
      where FDIdepartamentos.deppais=FDIpaíses.paicodigo and
      FDIdepartamentos.depnombre='$nombre'";
  }else{
    $sql=$sql." and FDIdepartamentos.depnombre='$nombre'";
  }
}
```

Para que se vean los datos consultados más organizados se hizo uso del siguiente código que nos permite paginar los datos y la vez los introduce dentro de una tabla:

```
$result = mysql_db_query("fdi",$sql);
$num_record=mysql_num_rows($result);
$num_page=ceil($num_record/8);

echo"<center><h4>CONSULTA DE DEPARTAMENTOS</h4></center>";
echo "<center>";
echo "<table border=2 bgcolor='white' bordercolor='gray'> \n";
echo "<tr><td><h6><center>Codigo</center></h6></td>
  <td><h6><center>Nombre</center></h6></td>
  <td><h6><center>Pais</center></h6></td>
  <td colspan='2'><h6><center>Opciones</center></h6></td></tr>\n";
$contador=1;
while ($row = @mysql_fetch_row($result)){
  if(isset($min)){
    if(($contador>=$min) && ($contador<=$max)){

      echo "<tr><td>$row[0]</td>
        <td>$row[1]</td>
        <td>$row[3]</td>
        <td><a href='departamentos.php?codigo=$row[0]&
          nombre=$row[1]&
          pais=$row[2]&
          boton=Editar'>Editar</a></td>
        <td><a href='departamentos.php?boton=Borrar&codigo=$row[0]'
          onClic="return valeliminar();" <?php echo" >Borrar</a></td></tr> \n";
    }
  }
  $contador++;
}
```



Fdi manual del sistema

```
}
echo "</table> \n";
for($i=1;$i<=$num_page;$i++){
    if($i==1){
        $min=1;
        $max=8;
    }
    else{
        $min=$min+8;
        $max=$max+8;
    }
    echo
    href=departamentos.php?boton=$boton&codigo=$codigo&nombre=$nombre&pais=$pais&min=$min&max=$max>$i</a>|";
}
echo "</center><br><br><br><br><br>";
?>
```

Con este botón no sólo podemos visualizar los datos sino que podemos cargar la información de un determinado departamento al formulario para que se le realicen las actualizaciones que el usuario final le desee hacer y seleccionar al departamento que se va a eliminar.

- **Actualizar:** Aquí con sólo darle click al botón **Actualizar** se hace una renovación de los datos almacenados en la Base de Datos a través de la función update de SQL que es la encargada de remplazar todos aquellos datos viejos que tenía el usuario por los nuevos que acaba de digitar.

Para el administrador de una Base de Datos es muy importante saber que hacen los usuarios con la información que es usada en la aplicación y es por eso que se realizan inserciones en la tabla FDILOGS (tabla que contiene todos los movimientos que realicen los usuarios en la aplicación) después cualquier movimiento que afecta a la Base de Datos, esta tabla contiene los siguientes campos:

Logusuario
Logfecha
Loghora
Logtransacción
Logtabla

A través de esta tabla el administrador sabrá que hizo el usuario X, el día Y, a la hora Z, el moviendo que realizó en la Base de Datos y sobre que tabla lo efectuó

5.6. MODIFICACIÓN DE CIUDADES



Para hacer este formulario se recurrió al uso de HTML, PHP, SQL y JAVASCRIPT que nos ayudaron a darle una buena apariencia al diseño y a permitir manejar datos almacenados en una Base de Datos.

En este formulario se hicieron validaciones con JAVASCRIPT para que el usuario confirme algún movimiento como cuando va a borrar un determinado registro o va a digitar los valores correspondiente a cada campo del formulario y no los deje vacío por ejemplo en el campo nombre sólo se puede digitar letras porque si coloca números le saldrá un mensaje de error hecho en JAVASCRIPT sugiriéndole que coloque letras para poder avanzar. Unas de las funciones que aparece a continuación son usadas para validar los campos alfabético:

```
function isAlphabetic (s,p)
{
  var i;
  for (i = 0; i < s.length; i++)
  {
    // Verifica que el actual caracter sea letra.
    var c = s.charAt(i);

    if (!isLetter(c)){
      alert("El campo solo permite letras")
      document.foms[0].elements[p].focus()
      return false;
    }
  }
  return true;
}

function isLetter (c)
{
```





Fdi manual del sistema

```
var uppercaseLetters = "ABCDEFGHIJKLMNOPQRSTUVWXYZÁÉÍÓÚÑ .-,"
var lowercaseLetters = "abcdefghijklmnopqrstuvwxyzáéíóúñü"
```

```
return( ( uppercaseLetters.indexOf( c ) != -1 ) ||
        ( lowercaseLetters.indexOf( c ) != -1 ) )
```

}
Este formulario contiene 2 campos de texto en donde el coordinador tiene que digitar la información y estos son código y nombre y los otros dos campos país y departamento son de selección en cual el usuario tiene que escoger la información disponible en la lista desplegable que se les presenta a la hora de utilizar este formulario, los datos que visualiza en las lista son extraídos de las tabla FDIPAISES y (tabla que contiene todos los países de procedencia de estudiante y profesores) y FDIDEPARTAMENTOS(tabla que contiene todos los departamentos de procedencia de estudiante y profesores), para extraer la información de la tabla el procedimiento que se realizó fue este:

Para países

```
$result = mysql_query("SELECT * FROM FDIPAISES", $conexion);
while ($row = mysql_fetch_row($result)){
if($row[0]==$pais) echo "<option value=\".$row[0].\" selected>".$row[1]."</OPTION>";

else
        echo "<option value=\".$row[0].>".$row[1]."</OPTION>";
}
?>
```

Para departamentos

```
<?
$result = mysql_query("SELECT * FROM FDIDEPARTAMENTOS where deppais=$pais",
$conexion);
while ($row = mysql_fetch_row($result)){
if($row[0]==$departamento) echo "<option value=\".$row[0].\"
selected>".$row[1]."</OPTION>";
else
        echo "<option value=\".$row[0].>".$row[1]."</OPTION>";
}
?>
```

En esta parte de la aplicación se manejan inserciones, borrados, actualizaciones y consulta de los datos a través de SQL, cuando se va a realizar movimientos como:

- **Insertar:** El usuario final digita todos los datos correspondiente a la ciudad a ingresar y presiona el botón **Insertar** para que se realice la inserción de los datos a la tabla FDICIUDADES (tabla que contiene todas las ciudades de procedencia de estudiante y profesores) siempre y cuando la ciudad a ingresar no exista, esta validación se realiza con este código:

```
$query="select* from FDICIUDADES";
$ptr = mysql_db_query("fdi",$query);
while ($row = @mysql_fetch_row($ptr)){
if ($row[0] == $codigo) {
$sw=1;
}
}
```



Fdi manual del sistema

```
}
```

Si la ciudad ya existe el usuario final visualizará un mensaje de advertencia comunicándole que ya esa ciudad esta registrada.

- **Borrar:** Con este link coordinador borra el registro de la ciudad seleccionada no solamente de la tabla FDICIUDADES (tabla que contiene todas las ciudades de procedencia de estudiante y profesores) sino también de la tabla FDIDEPARTAMENTOS (tabla que contiene todos los departamentos de procedencia de estudiante y profesores).
- **Consultar:** Con este botón se puede consultar a todas las ciudades o a una en específico por ejemplo si se quiere ver Barranquilla sólo tiene que colocar este nombre en el campo correspondiente al nombre de la ciudad y por medio del siguiente código podrá ver la consulta que requirió:

```
if(!empty ($nombre)){
  if(empty($sql)){
    $sql="SELECT FDlciudades.*,
      FDlpaises.painombre,
      FDldepartamentos.depnombre
    FROM FDlciudades,
      FDldepartamentos,
      FDlpaises
    where FDlciudades.ciupais=FDlpaises.paicodigo and
      FDlciudades.ciudepartamento=FDldepartamentos.depcodigo and
      FDlciudades.ciunombre='$nombre';
  }else{
    $sql=$sql." and FDlciudades.ciunombre='$nombre';
  }
}
```

Para que se vean los datos consultados más organizados se hizo uso del siguiente código que nos permite paginar los datos y la vez los introduce dentro de una tabla:

```
$result = mysql_db_query("fdi",$sql);
$num_record=mysql_num_rows($result);
$num_page=ceil($num_record/8);

echo"<center><h4>CONSULTA DE CIUDADES</h4></center>";
echo "<center>";
echo "<table border=2 bgcolor='white' bordercolor='gray' > \n";
echo "<tr><td><h5>Codigo</h5></td>
  <td><h5>Nombre</h5></td>
  <td><h5>Pais</h5></td>
  <td><h5>Departamento</h5></td>
  <td colspan='2'><h5>Opciones</h5></td></tr>\n";
$contador=1;
while ($row = @mysql_fetch_row($result)){
  if(isset($min)){
    if(($contador>=$min)&&($contador<=$max)){
```



Fdi manual del sistema

```

echo "<tr><td>$row[0]</td>
      <td>$row[1]</td>
      <td>$row[4]</td>
      <td>$row[5]</td>
      <td><a href='ciudades.php?codigo=$row[0]&
              nombre=$row[1]&
              pais=$row[2]&
              departamento=$row[3]&
              boton=Editar'>Editar</a></td>
      <td><a href='ciudades.php?boton=Borrar&codigo=$row[0]'
onClick='return valeliminar();' <?php echo " >Borrar</a></td></tr> \n";
    }
  } $contador++;
}
echo "</table> \n";
for($i=1;$i<=$num_page;$i++){
  if($i==1){
    $min=1;
    $max=8;
  }
  else{
    $min=$min+8;
    $max=$max+8;
  }
  echo
href=ciudades.php?boton=$boton&codigo=$codigo&nombre=$nombre&pais=$pa
is&departamento=$departamento&min=$min&max=$max>$i</a>|";
}

echo "</center><br><br><br><br><br>";
?>

```

Con este botón no sólo podemos visualizar los datos sino que podemos cargar la información de una determinada ciudad al formulario para que se le realicen las actualizaciones que el usuario final le desee hacer y seleccionar a la ciudad que se va a eliminar.

- **Actualizar:** Aquí con sólo darle click al botón **Actualizar** se hace una renovación de los datos almacenados en la Base de Datos a través de la función update de SQL que es la encargada de remplazar todos aquellos datos viejos que tenía el usuario por los nuevos que acaba de digitar.

Para el administrador de una Base de Datos es muy importante saber que hacen los usuarios con la información que es usada en la aplicación y es por eso que se realizan inserciones en la tabla FDILOGS (tabla que contiene todos los movimientos que realicen los usuarios en la aplicación) después cualquier movimiento que afecta a la Base de Datos, esta tabla contiene los siguientes campos:

- Logusuario
- Logfecha
- Loghora

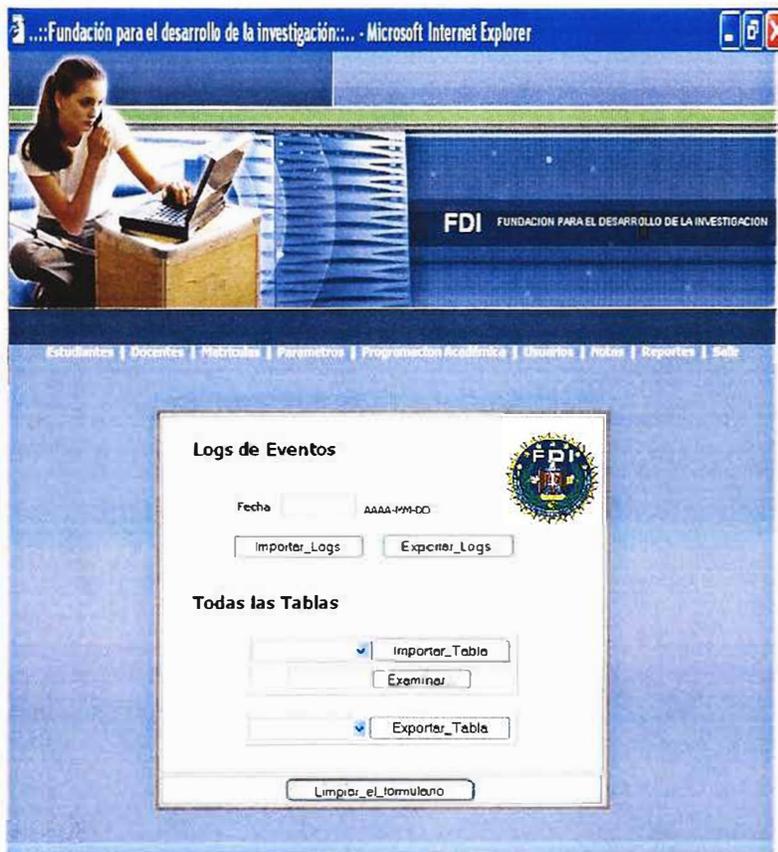
Fdi manual del sistema

Logtransacción

Logtabla

A través de esta tabla el administrador sabrá que hizo el usuario X, el día Y, a la hora Z, el moviendo que realizó en la Base de Datos y sobre que tabla lo efectuó

5.7. AUDITORÍA (LOGS DE EVENTOS)



En la construcción de este formulario se hizo uso de HTML, PHP, SQL y JAVASCRIPT que nos ayudaron a darle una buena apariencia al diseño y a permitir manejar datos almacenados en una Base de Datos, en esta forma encontramos dos parte que son **Logs de eventos** y **Todas las tabla**, la primera parte maneja la tabla FDILOGS (tabla donde se almacena todos los movimientos que hacen los usuarios en la aplicación) y la segunda manipula todos los datos que son utilizados en la aplicación.

En este formulario de hicieron validaciones con JAVASCRIPT para el ingreso correcto de la fecha, cuando los datos son exportados o el usuario intenta importar un tabla y no ha especificado cual.

En la parte de **Logs de eventos** encontramos un campo que es fecha en cual el usuario ingresa la fecha del archivo plano que quiere importar a la Base de datos, además de esto encontramos 2 botones esenciales para el funcionamiento de logs de eventos que son:

- **Importar_logs:** Con este botón se abre y se lee los datos de un archivo plano que tiene por nombre la fecha que usuario digito en el campo fecha, este archivo esta ubicado en la ruta c:\FDI\Logs y después de leída la información de este documento se coloca en una



Fdi manual del sistema

tabla en la aplicación para que el coordinador visualice la información correspondiente a los movimientos que han hecho los usuario que usan la aplicación en una determinada fecha , este procedimiento se realiza a través de este código:

```

$fp = @fopen("c:\FDI\Logs\".$aud_fecha.".doc", "r");
$content = @fread ($fp, 10000000);
$arreglo = split( "/", $content, 1000005 );
@fclose ($fp);
echo "<center><h4>Registro de ".$aud_fecha."</h4></center>";
echo "<center>";
echo "<table border=2 bgcolor='white' bordercolor='gray' \n";
echo "<tr><td><h6><center>Usuario</center></h6></td>";
    <td><h6><center>Fecha</center></h6></td>";
    <td><h6><center>Hora</center></h6></td>";
    <td><h6><center>Transaccion</center></h6></td>";
    <td><h6><center>Tabla</center></h6></td>";
</tr>\n";
$cont=0;
$longitud=count($arreglo);
echo "<tr>";
for($i = 0; $i <= $longitud; $i++) {
    echo "<td><center>$arreglo[$i]</center></td>";
    $cont=$cont+1;
    if($cont==5){
        $cont=0;
        echo "</tr><tr>";
    }
}
echo "</tr></table> \n Fin de los registros <br><br><br><br><br>";
}

```

- **Exportar_logs:** Con este botón se escribe en un archivo plano toda la información almacenada en la tabla FDILOGS (tabla donde se almacena todos los movimientos que hacen los usuarios en la aplicación) , la anteriormente mencionado se realiza por medio de este código:

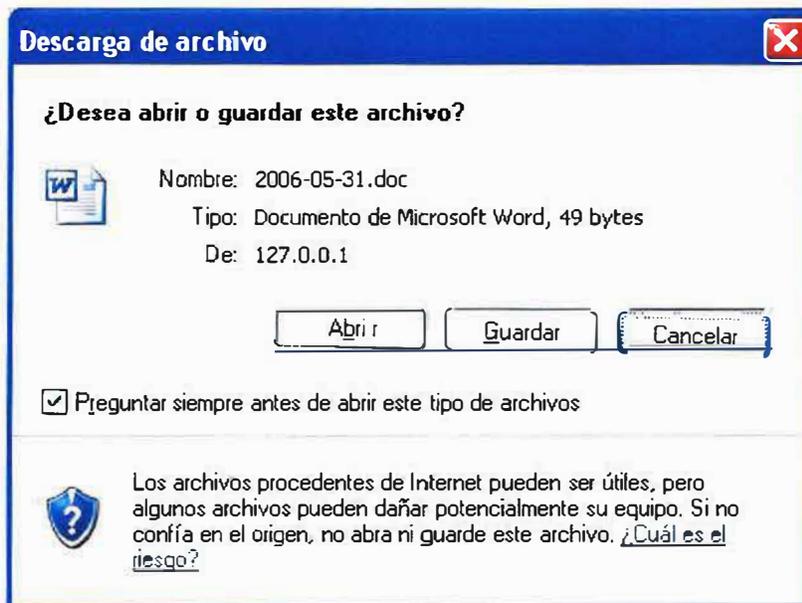
```

$sql="SELECT *
FROM fdilogs";
$result = mysql_db_query("fdi", $sql);
while ($row = mysql_fetch_row($result)){
    $registro=$row[0]."/".$row[1]."/".$row[2]."/".$row[3]."/".$row[4]."/";
    $fp = fopen("c:\FDI\Logs\".$fecha.".doc", "a");
    fwrite($fp, "\n");
    fwrite($fp, $registro, 1000);
}
@fclose($fp);
$query="delete from fdilogs";
mysql_db_query ("fdi", $query);

```

Fdi manual del sistema

Si el usuario quiere ver el archivo plano solo basta con darle click a links que aparece en la debajo de los botones **Importar_logs** y **Exportar_logs** para que les aparezca la siguiente ventana (se encuentra debajo) y abran el archivo que desean ver .



En la parte de **Todas las tabla** encontramos 3 tres campos que se encuentran acompañados cada uno con un botón los campos con sus respectivos botones son:

- **Importar_tabla:** Este botón a su derecha tiene una lista desplegable en cual el coordinador visualiza el nombre de todas las tabla que contiene la Base de Datos, al seleccionar una de ellas debe presionar el botón **Examinar** para que se abra un ventana como la anterior y el usuario seleccione el archivo plano que contiene la información que quiere retornar a la tabla escogida , puesto que en le momento de presionar el botón de **Importat_tabla** se borran todos los datos almacenado en dicha tabla , para pasar los datos del archivo plano a la tabla se pasa primero esa información a un array y de allí se ingresan de nuevo los datos a la tabla .
- **Exportar_tabla:** Este botón a su derecha tiene un lista desplegable en cual el coordinador visualiza el nombre de todas las tabla que contiene la Base de Datos, al seleccionar una de ellas y presionar este botón pasará toda la información que contiene la tabla seleccionada a un archivo plano ubicado en la ruta que seleccione el coordinador a través de una ventana de descarga de archivo.

5.8. MODIFICACIÓN DE PROGRAMAS



En la construcción de este formulario se hizo uso de HTML, PHP, SQL y JAVASCRIPT que nos ayudaron a darle una buena apariencia al diseño y a permitir manejar datos almacenados en una Base de Datos.

En este formulario de hicieron validaciones con JAVASCRIPT para que el usuario confirme algún movimiento como cuando va a borrar un determinado registro o va a digitar los valores correspondiente a cada campo del formulario y no los deje vacío por ejemplo en el campo nombre sólo se puede digitar letras porque si coloca números le saldrá un mensaje de error hecho en JAVASCRIPT sugiriéndole que coloque letras para poder avanzar. Unas de las funciones que aparece a continuación son usadas para validar los campos alfabético:

```
function isAlphabetic (s,p)
{
  var i;
  for (i = 0; i < s.length; i++)
  {
    // Verifica que el actual caracter sea letra.
    var c = s.charAt(i);

    if (!isLetter(c)){
      alert("El campo solo permite letras")
      document.forms[0].elements[p].focus()
      return false;
    }
  }
  return true;
}
```

```
function isLetter (c)
```



Fdi manual del sistema

```

{
var uppercaseLetters = "ABCDEFGHIJKLMNOPQRSTUVWXYZÁÉÍÓÚÑ .-,"
var lowercaseLetters = "abcdefghijklmnopqrstuvwxyzáéíóúñü"

return( ( uppercaseLetters.indexOf( c ) != -1 ) ||
        ( lowercaseLetters.indexOf( c ) != -1 ) )
}

```

Este formulario contiene 5 campos que son código, nombre, código icfes, fecha de creación y duración este último campo es de selección pero su información no proviene de la Base de Datos sino que es estática.

En esta forma se manejan inserciones, borrados, actualizaciones y consulta de los datos a través de SQL, cuando se va a realizar movimientos como:

- **Insertar:** El usuario final digita todos los datos correspondiente al programa a ingresar y presiona el botón **Insertar** para que se realice la inserción a la tabla FDIPROGRMAS (tabla que contiene todos los programas que dista actualmente el FDI) siempre y cuando el programa a ingresar no exista, esta validación se realiza con este código:

```

$query="select* from FDIprogramas";
$ptr = mysql_db_query("fdi",$query);
while ($row = @mysql_fetch_row($ptr)){
if ($row[0] == $codigo) {
$sw=1;
}
}
}

```

- **Borrar:** Con este link coordinador borra el registro del programa seleccionado y sus respectivas materias de la tabla FDIPENSUMMATERIA (tabla en donde se encuentran todos los pensums relacionados con una materia.).
- **Consultar:** Con este botón se puede consultar a todos los programas o uno en específico por ejemplo si se quiere ver el programa de criminalistica sólo tiene que colocar este nombre de la carrera en el campo correspondiente al nombre de programa y por medio del siguiente código podrá ver la consulta que requirió:

```

if(!empty ($nombre)){
if(empty($sql)){
$sql="SELECT * FROM FDIprogramas where pronombre='$nombre'";
}else{
$sql=$sql." and pronombre='$nombre'";
}
}
}

```

Para que se vean los datos consultados más organizados se hizo uso del siguiente código que nos permite paginar los datos y la vez los introduce dentro de una tabla:

```

$result = mysql_db_query("fdi",$sql);
$num_record=mysql_num_rows($result);
$num_page=ceil($num_record/8);

echo"<center><h4>CONSULTA DE PROGRAMAS</h4></center>";

```



Fdi manual del sistema

```

echo "<center>";
echo "<table border=2 bgcolor='white' bordercolor='gray'> \n";
echo "<tr><td><h5>Codigo</h5></td>
      <td><h5>Nombre</h5></td>
      <td><h5>Cod_icfes</h5></td>
      <td><h5>Creacion</h5></td>
      <td><h5>Duracion</h5></td>
      <td colspan='2'><center><h4>Opciones</h4></center></td></tr>\n";
$contador=1;
while ($row = @mysql_fetch_row($result)){
  if(isset($min)){
    if(($contador>=$min)&&($contador<=$max)){
      echo "<tr><td>$row[0]</td>
            <td>$row[1]</td>
            <td>$row[2]</td>
            <td>$row[3]</td>
            <td>$row[4]</td>
            <td><a href='programas.php?codigo=$row[0]&
                    nombre=$row[1]&
                    icfes=$row[2]&
                    creacion=$row[3]&
                    duracion=$row[4]&
                    boton=Editar'>Editar</a></td>
            <td><a href='programas.php?boton=Borrar&codigo=$row[0]'
onClick='return valeliminar();' <?php echo" >Borrar</a></td></tr> \n";
    }
  }
  $contador++;
}
echo "</table> \n";
for($i=1;$i<=$num_page;$i++){
  if($i==1){
    $min=1;
    $max=8;
  }
  else{
    $min=$min+8;
    $max=$max+8;
  }
  echo
href=programas.php?boton=$boton&codigo=$codigo&nombre=$nombre&icfes=$i
cfes&creacion=$creacion&duracion=$duracion&min=$min&max=$max>$i</a>|";
}

echo "</center><br><br><br><br><br>";
?>

```

Con este botón no sólo podemos visualizar los datos sino que podemos cargar la información de un determinado programa al formulario para que se le realicen las

Fdi manual del sistema

actualizaciones que el usuario final le desee hacer o seleccionar un programa para eliminar.

- **Actualizar:** Aquí con sólo darle click al botón **Actualizar** se hace una renovación de los datos almacenados en la Base de Datos a través de la función update de SQL que es la encargada de remplazar todos aquellos datos viejos que tenía el usuario por los nuevos que acaba de digitar.

Para el administrador de una Base de Datos es muy importante saber que hacen los usuarios con la información que es usada en la aplicación y es por eso que se realizan inserciones en la tabla FDILOGS (tabla que contiene todos los movimientos que realicen los usuarios en la aplicación) después cualquier movimiento que afecta a la Base de Datos, esta tabla contiene los siguientes campos:

Logusuario
Logfecha
Loghora
Logtransacción
Logtabla

A través de esta tabla el administrador sabrá que hizo el usuario X, el día Y, a la hora Z, el moviendo que realizó en la Base de Datos y sobre que tabla lo efectuó

5.9. MODIFICACIÓN DE PENSUMS



Este formulario tiene dos campos de selección que son programas y semestre en el cual la información del primero proviene de la tabla FDIPROGRMAS (tabla que contiene todos los programas que dista actualmente el FDI) y el segundo es estático porque la información que este campo contiene no se saca de la Base de Datos.

Al igual que los demás formularios este contiene los siguientes botones:

- **Insertar:** Con este botón el usuario final selecciona el programa y el semestre al cual le quiere crear el pensums y la almacena en la tabla FDIPENSUMS (tabla en donde se encuentran todos los programas relacionados con una materia), hay que tener en cuenta





Fdi manual del sistema

que al momento de ingresar un pensum este no exista, esta validación se hace a través del siguiente código:

```
$query="select* from FDIpensums";
$ptr = mysql_db_query("fdi",$query);
while ($row = @mysql_fetch_row($ptr)){
    if ($row[1] == $codigo){
        $sw=1;
    }
}
```

Si el pensum a ingresar ya existe le saldrá un mensaje de JAVASCRIPT comunicándole que ese pensum y existe

- **Borrar:** Con este link coordinador borra el registro del pensum seleccionado de la tabla FDIPENSUMS (tabla en donde se encuentran todos los programas relacionados con una materia).
- **Consultar:** Con este botón se puede consultar a todos los pensum o a uno en específico por ejemplo si se quiere ver los pensums de balística sólo tiene que seleccionar dicha carrera y eso la hace por medio de este código.

```
$sql="";
if(!empty ($programa)){
    $sql="SELECT FDIpensums.*,
        FDIprogramas.pronombre
        FROM FDIpensums,
        FDIprogramas
        where FDIprogramas.procodigo=FDIpensums.penprograma and
        FDIpensums.penprograma='$programa'";
}
```

Para que se vean los datos consultados más organizados se hizo uso del siguiente código que nos permite paginar los datos y la vez los introduce dentro de una tabla:

```
$result = mysql_db_query("fdi",$sql);
$num_record=mysql_num_rows($result);
$num_page=ceil($num_record/6);

echo"<center><h4>CONSULTA DE PENSUMS</h4></center>";
echo "<table align='center' border=2 bgcolor='white' bordercolor='gray'> \n";
echo "<tr>
    <td><center><h6>Programa</h6></center></td>
    <td><center><h6>Semestre</h6></center></td>
    <td colspan='3'><center><h6>Opciones</h6></center></td>
</tr>\n";
$contador=1;
while ($row = mysql_fetch_row($result)){
    if(isset($min)){
        if(($contador>=$min)&&($contador<=$max)){
            echo "<tr>
                <td>$row[3]</td>
```

Fdi manual del sistema

```

        <td>$row[2]</td>
        <td><a
href='manejodematerias.php?pensum=$row[1]'>Materias</a></td>
        <td><a href='pensums.php?boton=Borrar&codigo=$row[1]'
onClick="return valeliminar();" <?php echo " >Borrar</a></td></tr> \n";
    }
} $contador++;
}
echo "</table> \n";
for($i=1;$i<=$num_page;$i++){
    if($i==1){
        $min=1;
        $max=6;
    }
    else{
        $min=$min+6;
        $max=$max+6;
    }
    echo
href=pensums.php?boton=$boton&programa=$programa&semestre=$semestre&
min=$min&max=$max>$i</a>|";
}
echo "<br><br><br><br><br>";
?>

```

Con este botón no sólo podemos visualizar los datos correspondientes a los pensum sino que podemos hacer la asignación de las materias presionando el link de materia que nos transporta al siguiente formulario.



Este formulario contiene un campo de selección llamado materia y uno texto cuyo nombre es hora en el primer campo la información que se observa en la lista de selección proviene de la tabla FDIMATERIA (tabla que contiene toda la información correspondiente a las asignaturas que se dictan en la institución), la cual se carga a la lista por medio del siguiente código:

```
$query = mysql_query("SELECT * FROM FDImaterias", $conexion);  
while ($row = mysql_fetch_row($query)){  
  
    echo "<option value=".$row[0].">".$row[1]."</OPTION>";  
  
}
```

En el segundo campo el coordinador tiene que digitar el número de horas que quiere asignar a dicha materia, a parte de esos 2 campos también encontramos el botón **Insertar nueva materia** que al presionarlo verifica que la materia seleccionada no este ya asignada a ese pensum por medio del siguiente código:

```
$query = "select * from FDIpensummateria where codmateria='$materia' and  
codpensum=$pensum";  
$ptr = mysql_db_query("fdi",$query);  
$filas= mysql_num_rows ($ptr);
```

Después de hacer esa verificación se procede ha hacer la inserción de la materia escogida a la tabla FDIPENSUMMATERIA (tabla que contien toda la información correspondiente a las asignaturas que se dictan en la institución).

Borrar: Con este link el coordinador borra una materia seleccionada de la tabla FDIPENSUMMATERIA (tabla que contiene toda la información correspondiente a las asignaturas que se dictan en la institución).

5.10. MODIFICACIÓN DE CURSOS



En la construcción de este formulario se hizo uso de HTML, PHP, SQL y JAVASCRIPT que nos ayudaron a darle una buena apariencia al diseño y a permitir manejar datos almacenados en una Base de Datos.

En este formulario se hicieron validaciones con JAVASCRIPT para que el usuario confirme algún movimiento como cuando va a borrar un determinado registro o intenta hacer alguna



Fdi manual del sistema

inserción sin haber llenado todos los campos, por ejemplo si todos los campos del formulario están llenos a excepción del campo nombre y se presiona el botón de insertar se visualizará un mensaje de advertencia en JAVASCRIPT comunicándole al coordinador que debe llenar dicho campo, el código de dicho mensaje es el siguiente:

```
function validar() {  
    if (document.cursos.nombre.value==""){  
        alert('Debe ingresar el nombre del curso');  
        return false;  
    }  
    if (document.cursos.programa.value==""){  
        alert('Debe seleccionar un programa de la lista');  
        return false;  
    }  
    if (document.cursos.pensum.value==""){  
        alert('Debe seleccionar un pensum de la lista ');  
        return false;  
    }  
}
```

El anterior código no solo permite validar si el campo nombre no fue llenado sino todos los campos que contiene el formulario

Este formulario contiene 3 campos que son nombre, programa y pensum, el primer campo es de texto en cual el usuario tiene que digitar la información que desee ingresar en cambio los otros dos campos son de selección en el cual los datos se extrae de las tablas FDIPROGRAMAS y FDIPENSUM, estos datos se cargan en los campos de selección a través del siguiente código:

```
<?php  
$result = mysql_query("SELECT * FROM FDIprogramas", $conexion);  
while ($row = mysql_fetch_row($result)){  
if($row[0]==$programa) echo "<option value=".$row[0]."  
selected>".$row[1]."</OPTION>";  
  
else  
        echo "<option value=".$row[0].">".$row[1]."</OPTION>";  
}  
?>  
<?  
$result = mysql_query("SELECT * FROM FDIpensums where  
penprograma=$programa", $conexion);  
while ($row = mysql_fetch_row($result)){  
if($row[1]==$pensum) echo "<option value=".$row[1]."  
selected>".$row[2]."</OPTION>";  
else  
        echo "<option value=".$row[1].">".$row[2]."</OPTION>";  
}  
?>
```

En este formulario encontramos 4 botones que nos permite manipular la información que se encuentra en la Base de Datos y estos botones son:

- **Insertar:** Con este botón el usuario final ingresa toda la información del curso a la tabla FDICURSOS (tabla en donde se encuentran todos los cursos de las diversos programa



Fdi manual del sistema

del FDI), pero antes de hacer la inserción se verifica si el registro que se va a ingresar ya existe en la Base de Datos y esto se hace a través de siguiente código:

```

$query="select* from FDIcursos";
$ptr = mysql_db_query("fdi",$query);
while ($row = @mysql_fetch_row($ptr)){
  if ($row[0] == $nombre) {
    if ($row[1] == $programa){
      $sw=1;
    }
  }
}

```

Si el curso a ingresar ya existe le saldrá un mensaje de JAVASCRIPT comunicándole que ese curso y existe

- **Borrar:** Con este link coordinador borra el registro del curso seleccionado de la tabla FDICURSOS (tabla en donde se encuentran todos los cursos de las diversos programa del FDI).
- **Consultar:** Con este botón se puede consultar a todos los cursos o a uno en específico por ejemplo si se quiere ver todos los cursos de 1 a sólo tiene que colocar 1-a en el campo nombre, esto lo hace por medio del siguiente código

```

if(!empty($nombre)){
  $sql="SELECT FDIcursos.*,
    FDIprogramas.pronombre,
    FDIpensums.pensemestre
  FROM FDIcursos,
    FDIprogramas,
    FDIpensums
  where FDIcursos.curprograma=FDIprogramas.procodigo and
    FDIcursos.curpensum=FDIpensums.pencodigo and
    FDIcursos.curnombre='$nombre'";
}

```

Para que se vean los datos consultados más organizados se hizo uso del siguiente código que nos permite paginar los datos y la vez los introduce dentro de una tabla:

```

$result = mysql_db_query("fdi",$sql);
$num_record=mysql_num_rows($result);
$num_page=ceil($num_record/8);

echo "<center><h4>CONSULTA DE CURSOS</h4></center>";
echo "<center>";
echo "<table border=2 bgcolor='white' bordercolor='gray'> \n";
echo "<tr><td><h6><center>Nombre</center></h6></td>
  <td><h6><center>Programa</center></h6></td>
  <td><h6><center>Pensum</center></h6></td>
  <td colspan='2'><h6><center>Opciones</center></h6></td></tr>\n";
$contador=1;
while ($row = @mysql_fetch_row($result)){
  if(isset($min)){

```



Fdi manual del sistema

```

if(($contador>=$min)&&($contador<=$max)){
    echo "<tr><td>$row[0]</td>
        <td>$row[3]</td>
        <td>$row[4]</td>
        <td><a href='cursos.php?nombre=$row[0]&
            programa=$row[1]&
            pensum=$row[2]&
            boton=Editar'>Editar</a></td>
        <td><a href='cursos.php?boton=Borrar&nombre=$row[0]'
            onClick='return valeliminar();' <?php echo " >Borrar</a></td></tr> \n";
    }
} $contador++;
}
echo "</table> \n";
for($i=1;$i<=$num_page;$i++){
    if($i==1){
        $min=1;
        $max=8;
    }
    else{
        $min=$min+8;
        $max=$max+8;
    }
    echo
    href=cursos.php?boton=$boton&nombre=$nombre&programa=$programa&pensum=$pensum&min=$min&max=$max>$i</a>|";
}

echo "</center><br><br><br><br><br>";

?>

```

Con este botón no sólo podemos visualizar los datos sino que podemos cargar la información de un determinado curso al formulario para que se le realicen las actualizaciones que el usuario final le desee hacer o seleccionar un curso para eliminar.

- **Actualizar:** Aquí con sólo darle click al botón **Actualizar** se hace una renovación de los datos almacenados en la Base de Datos a través de la función update de SQL que es la encargada de remplazar todos aquellos datos viejos que tenía el usuario por los nuevos que acaba de digitar.

Para el administrador de una Base de Datos es muy importante saber que hacen los usuarios con la información que es usada en la aplicación y es por eso que se realizan inserciones en la tabla FDILOGS (tabla que contiene todos los movimientos que realicen los usuarios en la aplicación) después cualquier movimiento que afecta a la Base de Datos, esta tabla contiene los siguientes campos:

Logusuario
Logfecha
Loghora
Logtransacción

Fdi manual del sistema

Logtabla

A través de esta tabla el administrador sabrá que hizo el usuario X, el día Y, a la hora Z, el moviendo que realizó en la Base de Datos y sobre que tabla lo efectuó

5.11. MODIFICACIÓN DE MATERIAS



En la construcción de este formulario se hizo uso de HTML, PHP, SQL y JAVASCRIPT que nos ayudaron a darle una buena apariencia al diseño y a permitir manejar datos almacenados en una Base de Datos.

En este formulario se hicieron validaciones con JAVASCRIPT para que el usuario confirme algún movimiento como cuando va a borrar un determinado registro o va a digitar los valores correspondiente a cada campo del formulario y no los deje vacío por ejemplo en el campo nombre sólo se puede digitar letras porque si coloca números le saldrá un mensaje de error hecho en JAVASCRIPT sugiriéndole que coloque letras para poder avanzar. Unas de las funciones que aparece a continuación son usadas para validar los campos alfabético:

```
function isAlphabetic (s,p)
{
  var i;
  for (i = 0; i < s.length; i++)
  {
    // Verifica que el actual caracter sea letra.
    var c = s.charAt(i);

    if (!isLetter(c)){
      alert("El campo solo permite letras")
      document.forms[0].elements[p].focus()
      return false;
    }
  }
  return true;
}
```



Fdi manual del sistema

```
function isLetter (c)
{
  var uppercaseLetters = "ABCDEFGHIJKLMNOPQRSTUVWXYZÁÉÍÓÚÑ .-, "
  var lowercaseLetters = "abcdefghijklmnopqrstuvwxyzáéíóúñü"

  return( ( uppercaseLetters.indexOf( c ) != -1 ) ||
          ( lowercaseLetters.indexOf( c ) != -1 ) )
}
}
```

Este formulario contiene dos campos de texto en el cual el coordinador tiene que digitar toda la información que requiera ya sea para ingresar, consultar o actualizar, aparte de esto contiene los siguientes botones:

- **Insertar:** Con este botón el usuario final ingresa toda la información de la materia a la tabla `FDIMATERIAS` (tabla en donde se encuentran todas las materias de los diversos programas del FDI). pero antes de hacer la inserción se verifica si el registro que se va a ingresar ya existe en la Base de Datos y esto se hace a través de siguiente código:

```
$query="select* from FDImaterias";
$ptr = mysql_db_query("fdi",$query);
while ($row = @mysql_fetch_row($ptr)){
  if ($row[0] == $codigo) {
    $sw=1;
  }
}
}
```

Si la materia a ingresar ya existe le saldrá un mensaje de JAVASCRIPT comunicándole que esa materia ya existe

- **Borrar:** Con este link coordinador borra el registro de la materia seleccionada de la tabla `FDIMATERIAS` (tabla en donde se encuentran todos las materias de los diversos programas del FDI).
- **Consultar:** Con este botón se puede consultar todas las materias o a una en específico esto lo hace por medio del siguiente código:

```
if(!empty($nombre)){
  $sql="SELECT FDIcursos.*,
            FDIprogramas.pronombre,
            FDIpensums.pensemestre
        FROM FDIcursos,
            FDIprogramas,
            FDIpensums
        where FDIcursos.curprograma=FDIprogramas.procodigo and
            FDIcursos.curpensum=FDIpensums.pencodigo and
            FDIcursos.cumombre='$nombre'";
}
```

Para que se vean los datos consultados más organizados se hizo uso del siguiente código que nos permite paginar los datos y la vez los introduce dentro de una tabla:



Fdi manual del sistema

```

$result = mysql_db_query("fdi", $sql);
$num_record=mysql_num_rows($result);
$num_page=ceil($num_record/8);

echo"<center><h4>CONSULTA DE MATERIAS</h4></center>";
echo "<center>";
echo "<table border=2 bgcolor='white' bordercolor='gray'> \n";
echo "<tr><td><h6>Codigo</h6></td>
      <td><h6>Nombre</h6></td>
      <td colspan='2'><center><h6>Opciones</h6></center></td></tr>\n";
$contador=1;
while ($row = @mysql_fetch_row($result)){
  if(isset($min)){
    if(($contador>=$min)&&($contador<=$max)){
      echo "<tr><td>$row[0]</td>
            <td>$row[1]</td>
            <td><a href='materias.php?codigo=$row[0]&
                    nombre=$row[1]&
                    boton=Editar'>Editar</a></td>
            <td><a href='materias.php?boton=Borrar&codigo=$row[0]'
onClick='return valeliminar();'<input type='text'>
Borrar</a></td></tr> \n";
    }
  }
  $contador++;
}
echo "</table> \n";
for($i=1;$i<=$num_page;$i++){
  if($i==1){
    $min=1;
    $max=8;
  }
  else{
    $min=$min+8;
    $max=$max+8;
  }
  echo
href='materias.php?boton=$boton&codigo=$codigo&nombre=$nombre&min=$min
&max=$max>$i</a>|";
}

echo "</center><br><br><br><br><br>";
?>

```

Con este botón no sólo podemos visualizar los datos sino que podemos cargar la información de una determinada materia al formulario para que se le realicen las actualizaciones que el usuario final le desee hacer o seleccionar una materia para eliminar.

Fdi manual del sistema

- **Actualizar:** Aquí con sólo darle click al botón **Actualizar** se hace una renovación de los datos almacenados en la Base de Datos a través de la función update de SQL que es la encargada de remplazar todos aquellos datos viejos que tenía el usuario por los nuevos que acaba de digitar.

Para el administrador de una Base de Datos es muy importante saber que hacen los usuarios con la información que es usada en la aplicación y es por eso que se realizan inserciones en la tabla FDILOGS (tabla que contiene todos los movimientos que realicen los usuarios en la aplicación) después cualquier movimiento que afecta a la Base de Datos, esta tabla contiene los siguientes campos:

Logusuario
Logfecha
Loghora
Logtransacción
Logtabla

A través de esta tabla el administrador sabrá que hizo el usuario X, el día Y, a la hora Z, el moviendo que realizó en la Base de Datos y sobre que tabla lo efectuó

5.12. MODIFICACIÓN DE USUARIOS



En la construcción de este formulario se hizo uso de HTML, PHP, SQL y JAVASCRIPT que nos ayudaron a darle una buena apariencia al diseño y a permitir manejar datos almacenados en una Base de Datos.

Este formulario contiene 3 campos que son código, password y rool, los dos primeros campos son de texto lo cual quiere decir que el coordinador tiene que digitar la información y en el campo rool el usuario tiene que seleccionar el tipo de usuario, esta información que contiene la lista desplegable es estática puesto que los datos que se carga no proviene de la Base de Datos

- **Insertar:** Con este botón el usuario final ingresa toda la información del usuario a la tabla FDIUSUARIOS (tabla en donde se encuentran todos los usuarios que van hacer uso de la aplicación). pero antes de hacer la inserción se verifica si el registro que se va a ingresar ya existe en la Base de Datos y esto se hace a través de siguiente código:





Fdi manual del sistema

```

$query="select* from FDIusuarios";
  $ptr = mysql_db_query("fdi",$query);
  while ($row = @mysql_fetch_row($ptr)){
    if ($row[0] == $codigo) {
      $sw=1;
    }
  }
}

```

Si el usuario a ingresar ya existe le saldrá un mensaje de JAVASCRIPT comunicándole que ese usuario ya existe

- **Borrar:** Con este link coordinador borra el registro del usuario seleccionado de la tabla FDIUSUARIOS (tabla en donde se encuentran todos los usuarios que van hacer uso de la aplicacion)
- **Consultar:** Con este botón se puede consultar a todos los usuario o uno en específico por ejemplo el usuario codigo 001 esto lo hace por medio del siguiente codigo:

```

if(!empty ($codigo)){
  $sql="SELECT * FROM FDIusuarios where usucodigo='$codigo'";
}

```

Para que se vean los datos consultados más organizados se hizo uso del siguiente código que nos permite paginar los datos y la vez los introduce dentro de una tabla:

```

$result = mysql_db_query("fdi",$sql);
$num_record=mysql_num_rows($result);
$num_page=ceil($num_record/8);

```

```

echo"<center><h4>CONSULTA DE USUARIOS</h4></center>";
echo "<center>";
echo "<table border=2 bgcolor='white' bordercolor='gray'> \n";
echo "<tr><td><h6><center>Codigo</center></h6></td>
      <td><h6><center>Password</center></h6></td>
      <td><h6><center>Rooll</center></h6></td>
      <td colspan='2'><h6><center>Opciones</center></h6></td></tr>\n";

```

```

$contador=1;
while ($row = @mysql_fetch_row($result)){
  if(isset($min)){
    if(($contador>=$min)&&($contador<=$max)){
      //INVERSO DE MD5 PARA MOSTRAR EN FORMULARIO
      echo "<tr><td><center>$row[0]</center></td>
            <td><center>$row[1]</center></td>
            <td><center>$row[2]</center></td>
            <td><a href='usuarios.php?codigo=$row[0]&
                    password=$row[1]&
                    rooll=$row[2]&
                    boton=Editar'>Editar</a></td>
            <td><a href='usuarios.php?boton=Borrar&codigo=$row[0]'
onClick='return valeliminar();' <?php echo " >Borrar</a></td></tr> \n";
    }
}

```



Fdi manual del sistema

```
    } $contador++;  
  }  
  echo "</table> \n";  
  for($i=1;$i<=$num_page;$i++){  
    if($i==1){  
      $min=1;  
      $max=8;  
    }  
    else{  
      $min=$min+8;  
      $max=$max+8;  
    }  
    echo  
    href=usuarios.php?boton=$boton&codigo=$codigo&password=$password&root=  
$root&min=$min&max=$max>$i</a>|";  
  }  
  
  echo "</center><br><br><br><br><br>";  
?>
```

Con este botón no sólo podemos visualizar los datos sino que podemos cargar la información de un determinado usuario al formulario para que se le realicen las actualizaciones que el usuario final le desee hacer o seleccionar un usuario para eliminar.

- **Actualizar:** Aquí con sólo darle click al botón **Actualizar** se hace una renovación de los datos almacenados en la Base de Datos a través de la función update de SQL que es la encargada de remplazar todos aquellos datos viejos que tenía el usuario por los nuevos que acaba de digitar.

Para el administrador de una Base de Datos es muy importante saber que hacen los usuarios con la información que es usada en la aplicación y es por eso que se realizan inserciones en la tabla FDILOGS (tabla que contiene todos los movimientos que realicen los usuarios en la aplicación) después cualquier movimiento que afecta a la Base de Datos, esta tabla contiene los siguientes campos:

Logusuario
Logfecha
Loghora
Logtransacción
Logtabla

A través de esta tabla el administrador sabrá que hizo el usuario X, el día Y, a la **hora Z**, el moviendo que realizó en la Base de Datos y sobre que tabla lo efectuó

5.13. MODIFICACIÓN DE NOTAS POR CURSO



En la construcción de este formulario se hizo uso de HTML, PHP, SQL y JAVASCRIPT que nos ayudaron a darle una buena apariencia al diseño y a permitir manejar datos almacenados en una Base de Datos.

En este formulario se hicieron validaciones con JAVASCRIPT una de esas validaciones es cuando el usuario intenta consultar una nota y no vigencia todos los campos de formulario, cuando esto pasa el coordinador observará un mensaje de advertencia sugiriéndole que por favor llene todos los campos, el código por medio del cual se realiza este moviendo es el siguiente:

```
function validar() {
    if (document.notas.ano.value==""){
        alert('Debe ingresar el año de consulta');
        return false;
    }
    if (document.notas.periodo.value==""){
        alert('Debe seleccionar un periodo');
        return false;
    }
    if (document.notas.programa.value==""){
        alert('Debe seleccionar un programa de la lista');
        return false;
    }
    if (document.notas.curso.value==""){
        alert('Debe seleccionar un curso de la lista');
        return false;
    }
}
```



Fdi manual del sistema

```

}
    if (document.notas.materia.value==""){
        alert("Debe seleccionar una materia de la lista");
        return false;
    }
    if (document.notas.nota.value==""){
        alert("Debe seleccionar la nota a modificar");
        return false;
    }
}
}

```

Este formulario contiene 6 campos de los cuales 5 son de selección que son: período, programa , curso, materia y nota, el campo período es de selección pero estático porque no se tiene que consultar la Base de Datos para poder observar los datos en la lista desplegable del campo, en cambio en los demás se tiene que hacer una consulta a las en las siguientes tablas : FDIPROGRAMAS, FDICURSOS, FDI MATERIA y FDI NOTAS. El campo año toma automáticamente el año actual por default pero si el usuario requiere otro diferente tendrá que digitarlo.

Este formulario contiene los dos siguientes botones que son:

- **Consultar:** Este botón permite que el coordinador visualice una tabla en la parte inferior del formulario con todos los estudiantes que ven una determinada materia, en un programa, semestre, año, período, curso y nota en específico, aparte de esto el usuario final podrá hacer las respectivas asignaciones de nota (ya se la nota 1, 2 ó 3) con solo digitar la nota correspondiente a cada alumno y presionar el botón de **Guardar**, esta parte del almacenamiento de la notas se realiza por medio del siguiente código:

```

foreach($valor as $codigo => $dato){

$query = "select * from FDInotas where notestudiante='$codigo' and
        notmateria='$materia' and
        notano='$ano' and
        notperiodo='$periodo'";

$ptr = mysql_db_query("fdi",$query);
$filas= mysql_num_rows($ptr);
if ($filas != NULL){
    if($nota=='1'){
        $consulta = "update FDInotas set notprimera='$dato' where
notestudiante='$codigo' and
notmateria='$materia' and
notano='$ano' and
notperiodo='$periodo'";
        mysql_db_query ("fdi",$consulta);
    }
    if($nota=='2'){
        $consulta = "update FDInotas set notsegunda='$dato' where
notestudiante='$codigo' and

```



Fdi manual del sistema

```
notmateria='$materia' and
```

```
notano='$ano' and
```

```
notperiodo='$periodo';
```

```
mysql_db_query ("fdi", $consulta);  
}  
if($nota=='3'){  
$consulta = "update FDInotas set nottercera='$dato' where  
notestudiante='$codigo' and
```

```
notmateria='$materia' and
```

```
notano='$ano' and
```

```
notperiodo='$periodo';
```

```
mysql_db_query ("fdi", $consulta);  
}
```

Si el usuario necesita hacer alguna actualización sólo basta con realizar la consulta que requiera, hace los cambios y presiona el botón **Guardar** para almacenar los datos que fueron modificados.

- **Cancelar:** Este botón sirve para limpiar los campos de formulario y hacer desaparecer la tabla que observamos cuando presionamos el botón **Consultar**.

Para el administrador de una Base de Datos es muy importante saber que hacen los usuarios con la información que es usada en la aplicación y es por eso que se realizan inserciones en la tabla FDILOGS (tabla que contiene todos los movimientos que realicen los usuarios en la aplicación) después cualquier movimiento que afecta a la Base de Datos, esta tabla contiene los siguientes campos:

Logusuario

Logfecha

Loghora

Logtransacción

Logtabla

A través de esta tabla el administrador sabrá que hizo el usuario X, el día Y, a la hora Z, el movimiento que realizó en la Base de Datos y sobre que tabla lo efectuó

5.14. MODIFICACIÓN DE NOTAS POR ESTUDIANTES



En la elaboración de este formulario se hizo uso de HTML, PHP, SQL y JAVASCRIPT que nos ayudaron a darle una buena apariencia al diseño y a permitir manejar datos almacenados en una Base de Datos.

En este formulario se hicieron validaciones con JAVASCRIPT para que el usuario no cometa errores como por ejemplo se quiere consultar que notas tiene un estudiante en una determinada materia y este digita el código del estudiante pero no la materia, que pasa que este observará un mensaje de advertencia diciéndole que por favor ingrese la materia, esto se hace a través de la siguiente función:

```
function validar() {  
    if (document.notas.codigo.value=="") {  
        alert('Debe seleccionar el codigo a consultar');  
        return false;  
    }  
    if (document.notas.materia.value=="") {  
        alert('Debe seleccionar una materia de la lista');  
        return false;  
    }  
}
```

Este formulario contiene 2 campos de selección dinámica que son código y materia, los cuales tienen que cargar la información a mostrar en el campo de las tablas FDIESTUDIANTES, FDI MATERIAS, y FDI NOTAS

En esta parte de la aplicación encontramos 2 botones muy fundamentales para el manejo de los datos que contiene la aplicación estos botones son:

- **Asignar nueva materia:** Después que el coordinador halla seleccionado el código y la materia debe presionar este botón para que pueda visualizar el formulario correspondiente a la asignación de materia que es el siguiente:



Para poderle asignar una materia a un estudiante primero se debe seleccionar la materia que se le va a otorgar al estudiante del campo de selección materia, el cual es dinámico puesto que la información que muestra es cargada de la tabla `FDIMATERIAS`, este procedimiento se hace a través del siguiente código:

```
$query = mysql_query("SELECT * FROM FDImaterias", $conexion);
```

Después de haber seleccionado la materia se procede a presionar el botón **Asignar nueva materia** para que se ingrese la materia al estudiante siempre y cuando esta materia no la tenga ya asignada, lo anterior se hace a través de siguiente código :

```
$query = "select *
          from FDInotas
          where notestudiante='$codigo' and
                notmateria='$materia' and
                notano='$ano' and
                notperiodo='$periodo'";
$ptr = mysql_db_query("fdi",$query);
$filas= mysql_num_rows($ptr);
if ($filas != NULL)
  echo "<script language='javascript'>alert('El registro ya existe');</script>";
else
  {
$sql = "insert into FDInotas (notestudiante, notmateria, notano, notperiodo)
        values ('$codigo', '$materia', '$ano',
'$periodo')";
```

Si el usuario desea **Borrar** alguna materia lo puede hacer con sólo presionar el link de borrar que se encuentra a lado de cada asignatura que tiene el estudiante asignadas para ese semestre, este procedimiento elimina el registro seleccionado de la tabla `FDINOTAS`.



Fdi manual del sistema

- **Consultar** : Con este botón el coordinador ve la notas de un estudiante en particular con el siguiente código:

```
$result = mysql_query("SELECT FDImaterias.matnombre,  
FDInotas.notprimera,  
FDInotas.notsegunda,  
FDInotas.nottercera,  
FDInotas.notano,  
FDInotas.notperiodo  
FROM FDImaterias, FDInotas  
where FDInotas.notestudiante='$codigo' and  
FDInotas.notmateria=$materia and  
FDInotas.notmateria=FDImaterias.matcodigo",
```

```
$conexion);
```

```
mysql_db_query ("fdi",$result);
```

Al presionar este botón se despliega un tabla en la parte inferior del formulario que contiene los siguientes campos : materia, nota1, nota2, nota3, año, período y el botón **Guardar** que le permite al coordinador guardar todas aquellas modificaciones con respecto a las notas que se le hallan realizado al estudiante que selección, esto se hace a través de este código:

```
$consulta = "update FDInotas set notprimera='$nota1'  
where notestudiante='$codigo' and  
notmateria='$materia' and  
notano='$ano' and  
notperiodo='$periodo'";  
mysql_db_query ("fdi",$consulta);
```

```
$consulta = "update FDInotas set notsegunda='$nota2'  
where notestudiante='$codigo' and  
notmateria='$materia' and  
notano='$ano' and  
notperiodo='$periodo'";  
mysql_db_query ("fdi",$consulta);
```

```
$consulta = "update FDInotas set nottercera='$nota3'  
where notestudiante='$codigo' and  
notmateria='$materia' and  
notano='$ano' and  
notperiodo='$periodo'";  
mysql_db_query ("fdi",$consulta);
```

5.15. REPORTE LISTA DE ASISTENCIA



En la construcción de este formulario se hizo uso de HTML, PHP, SQL y JAVASCRIPT que nos ayudaron a darle una buena apariencia al diseño y a permitir manejar datos almacenados en una Base de Datos.

Este formulario contiene 4 campos de los cuales 3 son de selección y son los siguientes: programa, curso y período, los dos primeros campos son dinámicos porque sus datos son extraídos de las tablas FDICURSOS y FDIPROGRAMAS, en cambio período es estático porque su información no proviene de la Base de Datos y por último el campo año que toma automáticamente al año actual por default pero se requiere de otro año lo tiene que digitar.

Este reporte no es más que visualizar de una manera agradable los datos del siguiente Query en una tabla construida en HTML dentro la aplicación:

```
$SQL="SELECT FDlestudiantes.estcodigo as CODIGO,  
        FDlestudiantes.estapellidos as APELLIDOS,  
        FDlestudiantes.estnombre as NOMBRE,  
        FDlestudiantes.estmail as MAIL  
FROM FDlestudiantes,  
        FDlmatriculas  
WHERE FDlmatriculas.matcurso='$curso' and  
        FDlmatriculas.matprograma='$programa' and  
        FDlmatriculas.matestudiantes=FDlestudiantes.estcodigo and  
        FDlmatriculas.matano='$ano' and  
        FDlmatriculas.matperiodo='$periodo'";
```

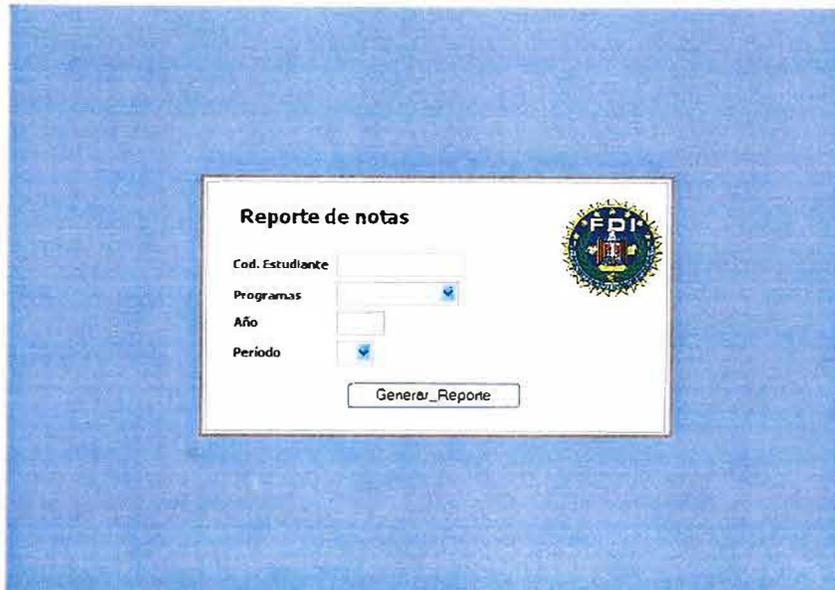
```
$result = mysql_query($SQL);
```

```
$query="Select FDlprogramas.pronombre  
        From FDlprogramas  
        Where FDlprogramas.procodigo='$programa'";
```

```
$resultado = mysql_query($query);
```

Los movimientos realizados en este formulario no se almacenan en la tabla FDILOGS porque no se hace ninguna alteración en la información almacenada en la Base de Datos

5.16. REPORTE DE NOTAS



En la construcción de este formulario se hizo uso de HTML, PHP, SQL y JAVASCRIPT que nos ayudaron a darle una buena apariencia al diseño y a permitir manejar datos almacenados en una Base de Datos.

Este formulario contiene 4 campos de los cuales 2 son de selección y son los siguientes programas y período en el cual el primero es dinámico por que su información viene de la tabla FDIPROGRAMAS y el segundo es estático por sus datos no viene de ninguna tabla pertenecientes a la Base de Datos, con respecto a los otros dos campos que cod. estudiante y año el usuario tiene que digitar la información que requiera pero si va a generar un reporte de año presente no tiene que colocar el año por la aplicación se lo coloca de lo contrario lo tiene que escribir.

Este reporte no es más que visualizar de una manera agradable los datos del siguiente Query en una tabla construida en HTML dentro la aplicación:

```
$sql = "SELECT FDInotas.notprimera,  
          FDInotas.notsegunda,  
          FDInotas.nottercera,  
          FDImaterias.matnombre  
FROM FDInotas,  
          FDImaterias  
where FDInotas.notestudiante='$estudiante' and  
       FDInotas.notano='$ano' and  
       FDInotas.notperiodo='$periodo' and  
       FDInotas.notmateria=FDImaterias.matcodigo";
```

```
$result = mysql_db_query("fdi", $sql);
```

```
//datos del individuo
```



Fdi manual del sistema

```
$query="Select FDlestudiantes.estnombre,  
          FDlestudiantes.estapellidos  
        From FDlestudiantes  
        Where FDlestudiantes.estcodigo='$estudiante';"  
$resultado = mysql_query($query);
```

5.17. REPORTE DINAMICOS



En esta parte de la aplicación el coordinador arma la consulta que requiera seleccionado las tablas y los campos necesarios para llevar a cabo la consulta que necesite en esos momentos, además le brinda la oportunidad de digitar su propio Query y a la hora de generar los reporte le puede colocar un alias a los campo de las tablas seleccionade según sea su gusto
Los movimientos realizados en este formulario no se almacenan en la tabla FDILOGS porque no se hace ninguna alteración en la información almacenada en la Base de Datos

**Desarrollo de un Software en Entorno Web
para el Manejo del Departamento Académico
en la Fundación para el Desarrollo de la Investigación
(FDI)**

Carlos Julio Nova Peñaranda

Adriana Patricia Arzayus Ecobar

Alberto Porto Danado

Universidad Simón Bolívar

Facultad

Ingeniería de Sistemas

Barranquilla





TABLA DE CONTENIDO

| | PAG |
|---|-----|
| 1. INTRODUCCION | 3 |
| 2. OBJETIVOS | 4 |
| 2.1 OBJETIVOS GENERALES | 4 |
| 2.2 OBJETIVOS ESPECIFICOS | 4 |
| 3. ENTRADA AL SISTEMA | 5 |
| 3.1 INICIO DE SESION(INDEX) | 5 |
| 3.2 SESION ESTUDIANTES | 6 |
| 3.3 SESION ADMINISTRADOR | 6 |
| 3.3.1 MODIFICACION DE ESTUDIANTES | 7 |
| 3.3.2 MODIFICACION DE DOCENTES | 11 |
| 3.3.2.1 CARGA ACADEMICA | 14 |
| 3.3.3 MODIFICACION DE MATRICULAS | 16 |
| 3.3.4 MODIFICACION DE PARAMETROS | 18 |
| 3.3.4.1 MODIFICACION DE PAISES | 19 |
| 3.3.4.2 MODIFICACION DE DEPARTAMENTOS | 21 |
| 3.3.4.3 MODIFICACION DE CIUDADES | 24 |
| 3.3.4.4 MODIFICACION DE LOGS | 27 |
| 3.3.5 MODIFICACION DE PROGRAMACION ACADEMICA | 32 |
| 3.3.5.1 MOD. DE PROGRAMACION ACADEMICA/PROGRAMAS | 32 |
| 3.3.5.2 MOD. DE PROGRAMACION ACADEMICA/PENSUMS | 35 |
| 3.3.5.3 MODIFICACION DE PROGRAMACION ACADEMICA/CURSOS | 39 |
| 3.3.5.4 MODIFICACION DE PROGRAMACION ACADEMICA/MATERIAS | 42 |
| 3.3.6 MODIFICACION DE USUARIOS | 45 |
| 3.3.7 NOTAS | 47 |
| 3.3.7.1 NOTAS, CONSULTA POR CURSOS | 48 |
| 3.3.7.2 NOTAS, CONSULTAS POR ESTUDIANTES | 50 |
| 3.3.8 REPORTE | 53 |
| 3.3.8.1 REPORTE, LISTA DE ASISTENCIA | 53 |
| 3.3.8.2 REPORTE DE NOTAS | 54 |
| 3.3.8.3 REPORTE DINAMICOS | 56 |



TABLA DE IMAGENES

| | PAG. |
|--------------------|------|
| FIGURA 1,2,3 | 5 |
| FIGURA 4 | 6 |
| FIGURA 5 | 7 |
| FIGURA 6,7 | 8 |
| FIGURA 8,9 | 9 |
| FIGURA 10,11 | 10 |
| FIGURA 12,13,14 | 11 |
| FIGURA 15,16 | 12 |
| FIGURA 17,18 | 13 |
| FIGURA 19,20,21 | 14 |
| FIGURA 22,23 | 15 |
| FIGURA 24,25 | 16 |
| FIGURA 26,27,28 | 17 |
| FIGURA 29,30 | 18 |
| FIGURA 31,32,33 | 19 |
| FIGURA 34,35 | 20 |
| FIGURA 36,37 | 21 |
| FIGURA 38,39,40,41 | 22 |
| FIGURA 42,43 | 23 |
| FIGURA 44,45 | 24 |
| FIGURA 46,47,48 | 25 |
| FIGURA 49,50 | 26 |
| FIGURA 51,52 | 27 |
| FIGURA 53,54,55 | 28 |
| FIGURA 56 | 29 |
| FIGURA 57,58,59 | 30 |
| FIGURA 60,61 | 31 |
| FIGURA 62,63 | 32 |
| FIGURA 64,65,66 | 33 |
| FIGURA 67 | 34 |
| FIGURA 68,69 | 35 |
| FIGURA 70,71 | 36 |
| FIGURA 72,73,74 | 37 |
| FIGURA 75 | 38 |
| FIGURA 76,77 | 39 |



INTRODUCCIÓN

La fundación para el desarrollo de la investigación FDI quiere ser la mejor en todos y cada uno de los aspectos, es por esto que se quiere superar todas aquellas partes donde se encuentran fallas como es el departamento académico que no tiene debidamente organizada la información y no cuenta con una aplicación que satisfaga todas sus necesidades.

Es por esto que se procede a desarrollar un software capaz de cumplir con todas aquellas necesidades que presentan en estos momentos la fundación para que con esta herramienta se trabaje de la mejor forma y no se les presente ningún contratiempo. Además, cuando se requiera un reporte de notas, etc, se realice de una manera rápida y eficiente.

Este proyecto pretende que el FDI supere todas aquellas fallas que presenta, hacer que la fundación reduzca sus costos lo que para nosotros significará aplicar todos los conocimientos adquiridos durante el transcurso de nuestra carrera profesional.



1 OBJETIVOS

1.1 OBJETIVO GENERAL.

Diseñar e implementar una aplicación en entorno web, para que el departamento académico permita manejar de una forma eficiente y confiable la información de los estudiantes en el FDI

1.2 OBTIVOS ESPECIFICOS

- Identificar los controles que se llevan en el registro de las notas, datos personales de los estudiantes, entre otros.
- Suministrar una herramienta que pueda hacer un mejor uso de los recursos que tiene disponible el FDI, para la realización de sus funciones.

2. ENTRADA AL SISTEMA

2.1 INICIO DE SESIÓN (INDEX)

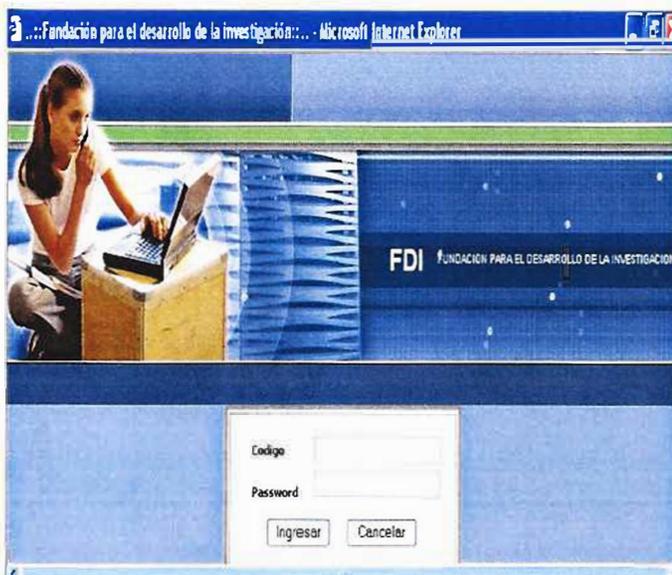


FIGURA 1

A través del anterior formulario podemos ingresar al sistema, a través del **Código** y el **Password** del estudiante o administrador, si el código o el password son incorrectos nos muestra los siguientes mensajes:



FIGURA 2



FIGURA 3

2.2 SESION ESTUDIANTES

Cuando el estudiante ingrese su código visualizará la siguiente pantalla, en donde podrá ver su nombre, código, número de cedula, materias que esta viendo en el período académico en curso, primer parcial, segundo parcial, parcial final y la nota definitiva.

FIGURA 4

Datos del estudiante

Materias y Notas del estudiante

| Materia | Parcial1 | Parcial2 | Final | Definitiva |
|--------------|----------|----------|-------|------------|
| Computadores | 5 | 0 | 0 | 1.5 |
| Gerencia | 5 | 4.5 | 0 | 2.85 |

Al transcurrir un lapso tiempo de 10 segundos este formulario desaparecerá y retornaremos al formulario inicial o index.

2.3 SESION ADMINSTRADOR

Cuando el administrador(coordinador académico) ingrese a esta parte de la aplicación encontrará la imagen que observamos en la parte inferior en donde le muestra un menú que contiene las siguientes enlaces: Estudiante, Docente, Matriculas, Parámetros(Países,

Departamento, Ciudades y Auditoría), Programación académica (Programa, Pensum, Curso y Materias) , Usuarios, Notas, Reportes que contiene Lista de Asistencia y Reportes de Notas y salir que regresa al administrador al Index.

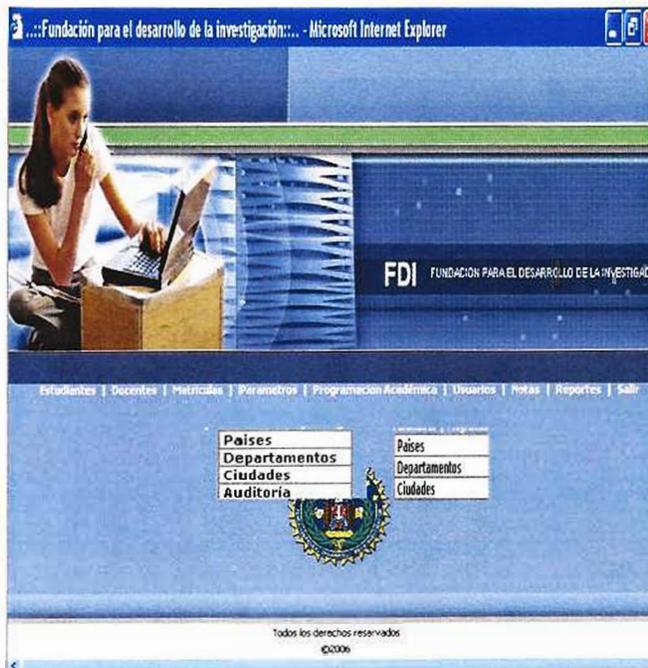
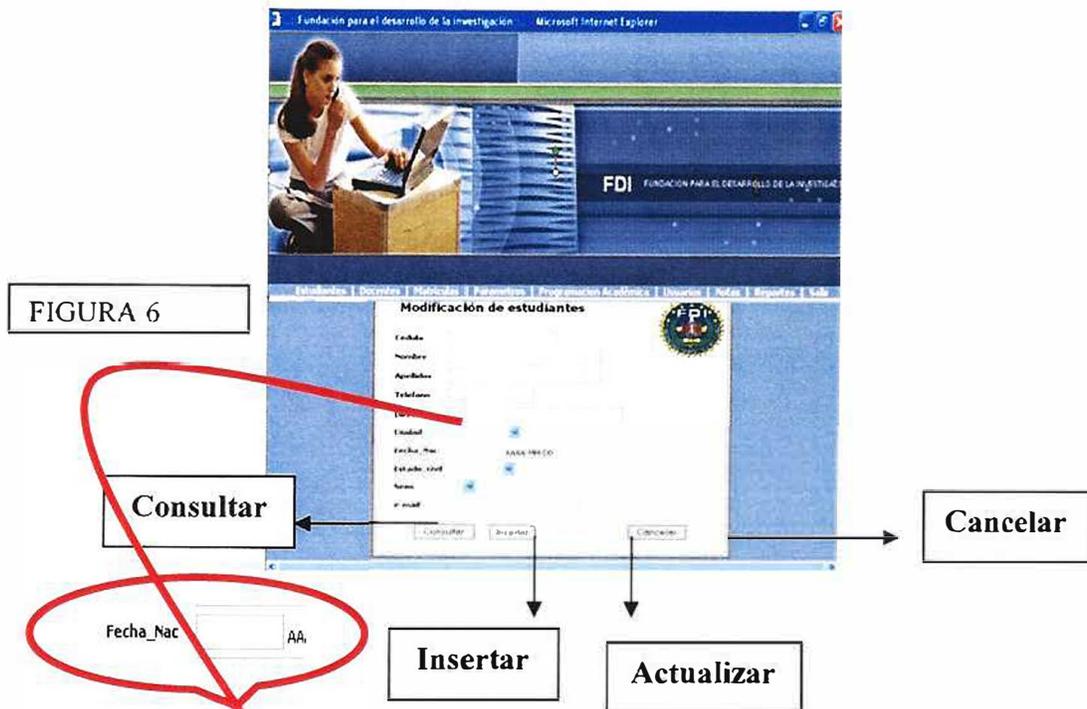


FIGURA 5

2.3.1 MODIFICACIÓN DE ESTUDIANTES

En esta sesión el coordinador podrá consultar, insertar, actualizar y cancelar los diferentes datos de los estudiantes matriculados en la fundación. Los datos de dichos alumnos se captarán a través de siguiente formulario.



La sección resaltada hace referencia al campo **Fecha de creación** en el cual sólo se pueden colocar fecha con el formato AAAA-MM-DD porque si el usuario coloca una fecha que no cumpla con este formato visualizará el mensaje que se encuentra en la parte inferior, si el usuario insiste en colocar una fecha inadecuada no podrá avanzar.



- **Consultar:** Dando click a este botón el coordinador podrá visualizar los datos personales correspondientes a todos los estudiantes matriculados en la fundación si la consulta requiere ser más específica se debe escribir parte de los datos a buscar en uno o varios campos perteneciente al formulario anteriormente mostrado ya sea el número cedula, nombres, apellidos, teléfono, dirección, ciudad, estado civil, sexo, etc y como resultado nos presentará lo siguiente:



- **Num_Pág:** Este número permite desplazarnos en la diversa paginas generadas por la consulta.
- **Editar:** Esté enlace le permitirá enviar los datos de un alumno en específico al formulario **Modificación de Estudiantes** para que se hagan las respectivas actualizaciones, cuando estos datos son transportados automáticamente el botón **Actualizar** se activa para que el coordinador pueda modificar la información , si el usuario no pasa los datos al formulario el botón **Actualizar** nunca se activará y por lo tanto no realizará ninguna modificación.
- **Borrar:** Con este enlace el coordinador puede borrar a cualquier estudiante matriculado. Cuando el usuario de click en este link el registro del estudiante que selecciono se borrará siempre y cuando le de aceptar en el mensaje de confirmación que se muestra a continuación:



Si la opción seleccionada fue aceptar borraré inmediatamente el registro del estudiante seleccionado.

- **Ingresar:** Con este botón él coordinador ingresará todos los datos de los alumnos inscritos en la institución, la información que se digitalice tiene que ir de acuerdo al tipo de dato, por ejemplo en el campo cedula no se puede colocar ninguna letra porque si sucede esto nos mostrará un mensaje como el que se encuentra en la parte inferior, además si el usuario insiste en colocar letras la inserción de los datos no se podrá realizar debido a que no le permitirá avanzar hasta que coloque sólo números.



FIGURA 10

Si el usuario en los campos alfabéticos intenta colocar algún número le saldrá el siguiente mensaje:



FIGURA 11

Si el usuario ingresa un registro ya existente le mostrara el siguiente mensaje





FIGURA 12

Si la dirección de correo que digito el usuario no es correcta visualizará este mensaje:



FIGURA 13

Dependiendo al campo que quede vacío en el formulario al momento de ingresar a un estudiante se mostrará este mensaje con el nombre del campo que debe llenar.



FIGURA 14

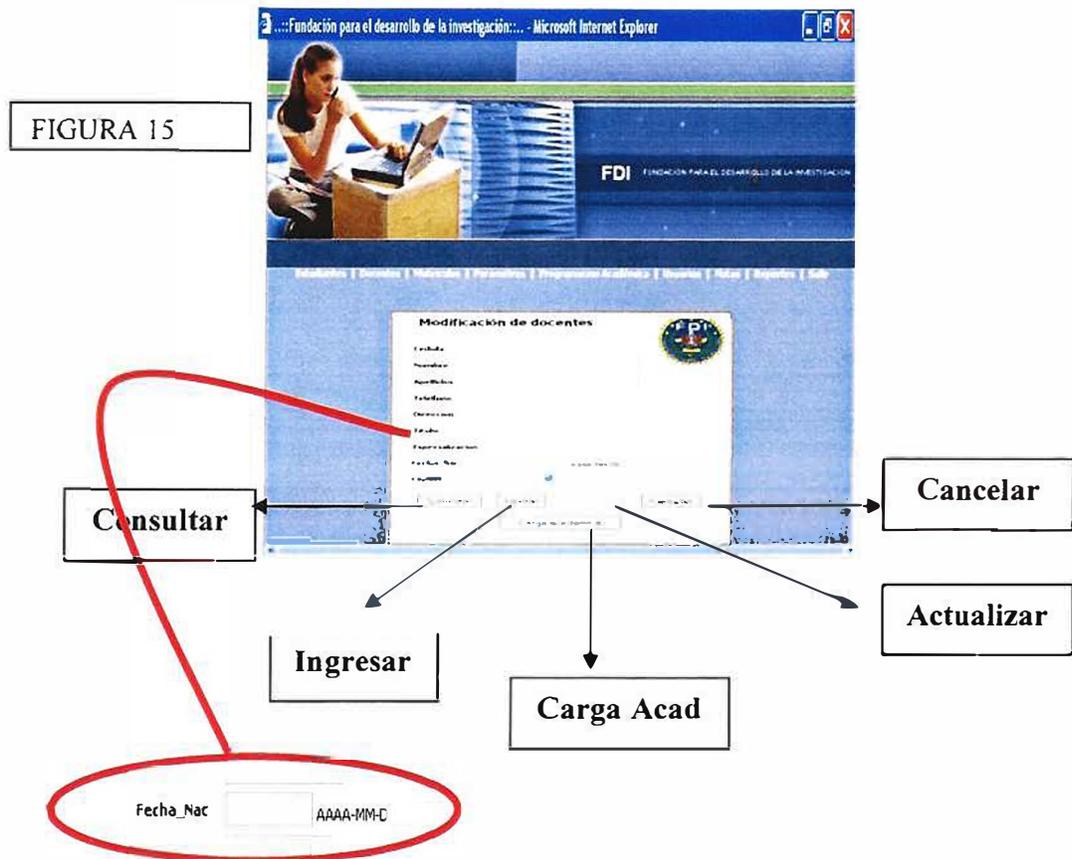
Campos

- Nombre
- Apellidos
- Direccion
- Ciudad
- Fecha_nac
- Estado_civil
- Sexo

- **Cancelar:** Con este botón el usuario limpiará todos los campos de formulario ya se para realizar un nueva búsqueda, ingresar los datos de otros estudiante ,etc.

2.3.2 MODIFICACIÓN DE DOCENTES

En esta sesión el coordinador manejará todos los datos personales y materias que dictan los diferentes profesores en la institución, lo anteriormente mencionado se hará a través del siguiente formulario.



La sección resaltada hace referencia al campo **Fecha de creación** en el cual sólo se pueden colocar fecha con el formato AAAA-MM-DD porque si el usuario coloca una fecha que no cumpla con este formato visualizará el mensaje que se encuentra en la parte inferior, si el usuario insiste en colocar una fecha inadecuada no podrá avanzar.



- **Consultar:** Dando click a este botón el coordinador podrá visualizar los datos personales correspondientes a todos los profesores que dictan clases en la fundación ,si la consulta requiere ser más específica se debe escribir parte de los datos a buscar en uno o varios campos pertenecientes al formulario anteriormente mostrado ya sea el número cedula, nombres, apellidos, teléfono, dirección, ciudad, estado civil, sexo, etc y como resultado nos presentará lo siguiente:

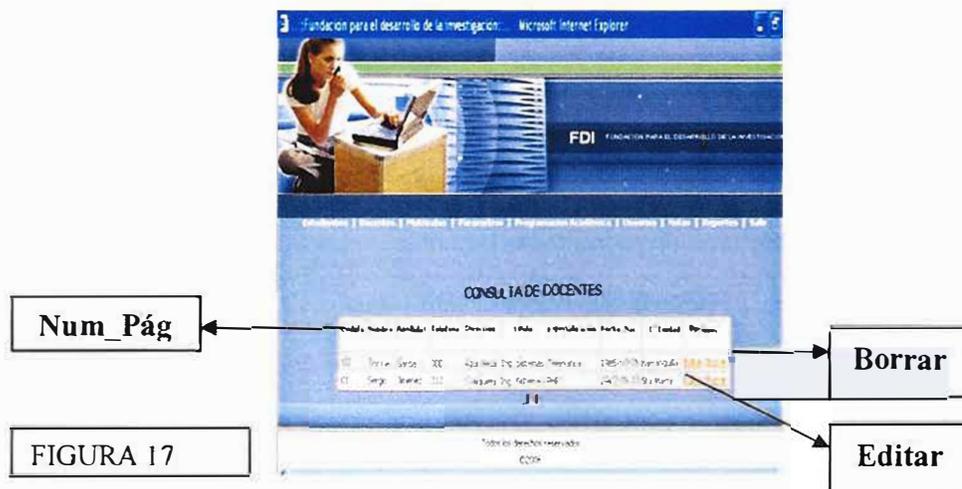


FIGURA 17

- **Num_Pág:** Este número permite desplazarnos en la diversa paginas generadas por la consulta.
- **Editar:** Esté enlace le permitirá enviar los datos de un docente en específico al formulario **Modificación de Docentes** para que se hagan las respectivas actualizaciones, cuando estos datos son transportados automáticamente el botón **Actualizar** se activa para que el coordinador pueda modificar la información , si el usuario no pasa los datos al formulario el botón **Actualizar** nunca se activará y por lo tanto no realizará ninguna modificación.
- **Borrar:** Con este enlace el coordinador podrá borrar a cualquier docente. Cuando el usuario de click en este link el registro del docente que selecciono se borrará siempre y cuando le de aceptar en el mensaje de confirmación que se muestra a continuación:



FIGURA 18

Si la opción seleccionada fue aceptar borrará inmediatamente el registro del docente seleccionado.

- **Ingresar:** Con este botón él coordinador ingresará todos los datos de los profesores en la institución, la información que se digitalice tiene que ir de acuerdo al tipo de dato, por ejemplo en el campo cedula no se puede colocar ninguna letra porque si sucede esto nos mostrará un mensaje como el que se encuentra en la parte inferior, además si el usuario insiste en colocar letras la inserción de los datos no se podrá realizar debido a que no le permitirá avanzar hasta que coloque sólo números.

FIGURA 19



Si el usuario en los campos alfabéticos intenta colocar algún número le saldrá el siguiente mensaje:

FIGURA 20



Si el usuario ingresa un registro ya existente le mostrara el siguiente mensaje.

FIGURA 21



Si algún cuadro de texto queda vacío en el formulario al momento de ingresar a un profesor se mostrará este mensaje con el nombre del campo que debe llenar. Dependiendo al campo que quede vacío se mostrará este mensaje con el nombre del campo que debe llenar.

FIGURA 22



Campos

- Nombre
- Apellidos
- Telefono
- Cedula
- Direccion
- Ciudad
- Titulo
- Especialización
- Fecha_nac.

- **Cancelar:** Con este botón el usuario limpiará todos los campos de formulario ya se para realizar un nueva búsqueda, ingresar los datos de otros docentes ,etc.

2.3.2.1 CARGA ACADÉMICA

En este formulario el coordinador asignará a cada uno de los docentes las materias que dictará durante el semestre y su respectivo número de horas. Para poder realizar esta

asignación, el usuario tendrá que digitar el número de cedula del docente o ir a la parte de consulta y dar click en el enlace **Editar** para que de esta manera se cargue los datos del docente en el formulario, para que así pueda hacer la asignación de las materias a través del botón **Carga Académica** que lo llevará a esta pantalla, si el campo cedula se encuentra vacío a la hora de asignarle materia a un profesor le saldrán el siguiente mensajes:

FIGURA 23



En cambio si el usuario digita algún número de cedula al momento de realizar la carga académica visualizará la siguiente imagen:

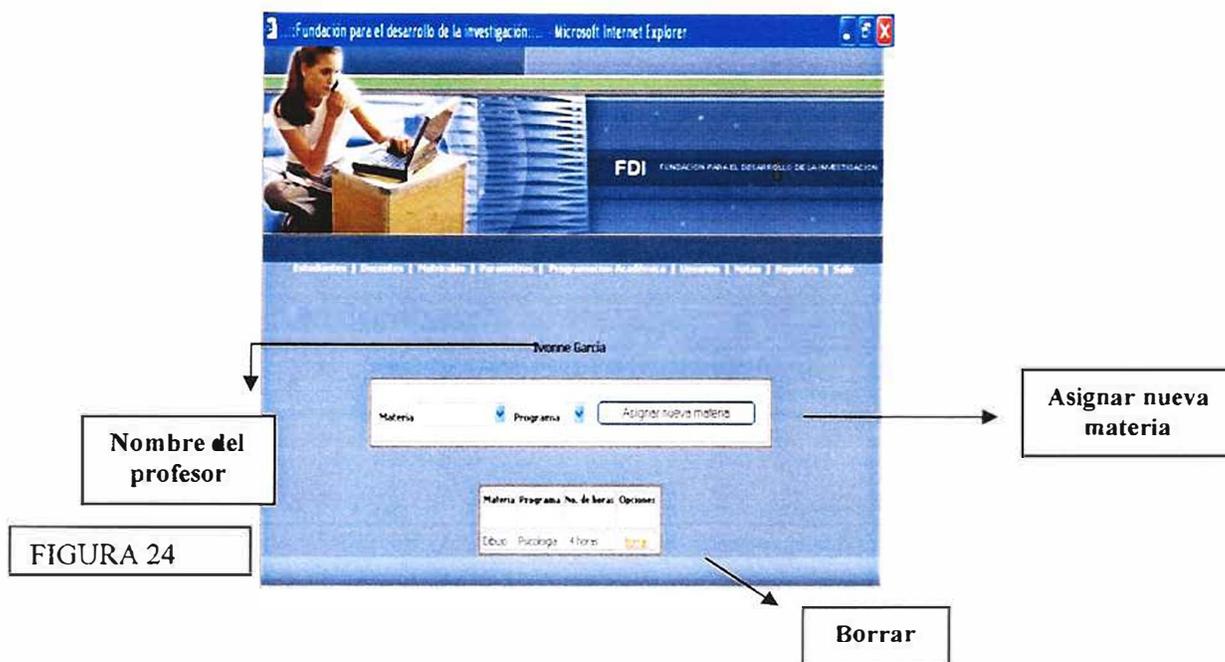


FIGURA 24

En la imagen anterior se observan dos campos que son **Materia** y **Programas** antes de hacer cualquiera asignación el usuario debe verificar que halla materias y programas registrados, esto lo podrá consultar en el enlace **Parámetros** y aquí se desplegará submenú en donde seleccionará el formulario **Materias o Programas** dependiendo de su necesidad, después de haber verificado que hallan materias y programas registrados se procederá a seleccionar la materia y el programa al cual pertenece dicha asignatura. Si al momento de asignarle una nueva materia a un profesor y no se selecciona la materia el usuario observará el siguiente mensaje:



FIGURA 25

- **Asignar nueva materia:** Con este botón el usuario asignará una determinada asignatura a un profesor siempre y cuando halla seleccionado la materia y el programa a cual pertenece, si el coordinador intenta ingresar una materia X correspondiente a un programa Z y esta asignación ya existe se visualizará el siguiente mensaje.



FIGURA 26

- **Borrar:** Con este enlace el coordinador podrá borrar cualquiera materia. Cuando el usuario de click en este link el registro de la materia que selecciono se borrará siempre y cuando le de aceptar en el mensaje de confirmación que se muestra a continuación:



FIGURA 27

2.3.3 MODIFICACIÓN DE MATRICULAS

En esta parte el coordinador registrará a todos los estudiantes matriculados en la fundación. En los campos **Programa** y **Curso** se debe tener en cuenta que previamente se hallan registrado programas y cursos en sus respectivos formularios porque de lo contrario no aparecerá nada en la lista desplegable, en el campo año el usuario no tendrá que digitar nada si va ha hacer referencia al año presente de lo contrario tendrá que tipiar el año.

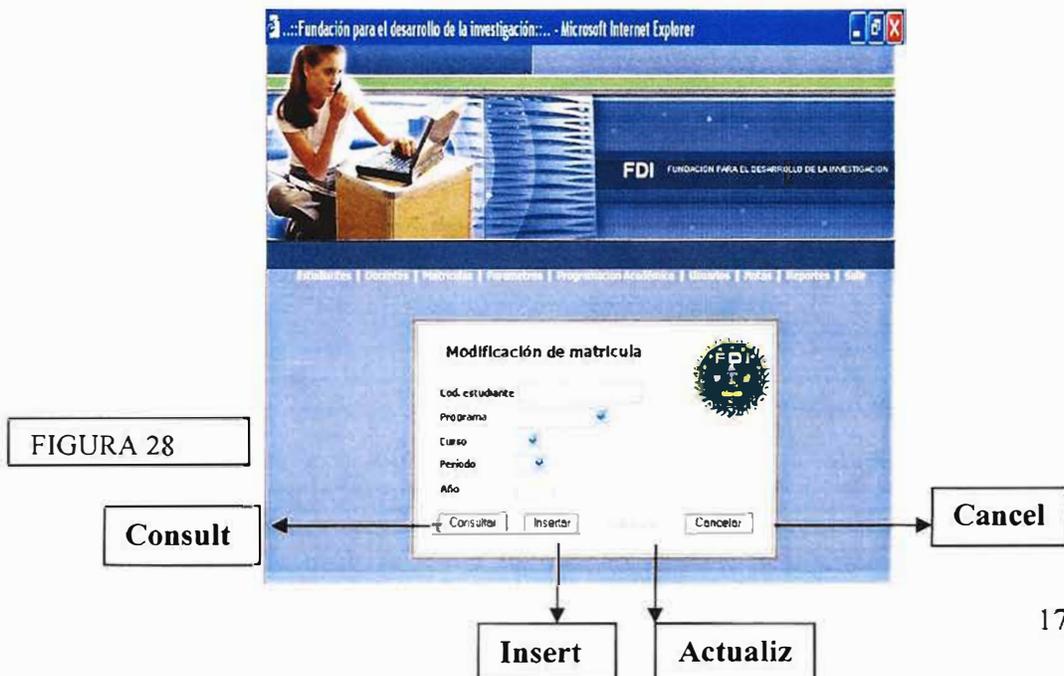


FIGURA 28

- **Consultar:** Dando click a este botón el coordinador podrá visualizar los datos de las matriculadas de la institución, si la consulta requiere ser más específica se debe escribir parte de los datos a buscar en uno o varios campos pertenecientes al formulario anteriormente mostrado ya sea el código del estudiante, programa, curso, período, etc y como resultado nos presentará lo siguiente:



- **Editar:** Este enlace le permitirá enviar los datos de una matricula en específico al formulario **Modificación de Matriculas** para que se hagan las respectivas actualizaciones con respecto a una determinada matricula.
- **Borrar:** Con este enlace el coordinador podrá borrar cualquier matricula. Cuando el usuario de click en este link el registro de la matricula que selecciono se borrará siempre y cuando le de aceptar en el mensaje de confirmación que se muestra a continuación:



Si la opción seleccionada fue aceptar borrará inmediatamente el registro de la matricula seleccionada.

- **Ingresar:** Con este botón él coordinador ingresará todos los datos correspondiente a la matricula, la información que se digitalice tiene que ir de acuerdo al tipo de dato, por ejemplo en el campo cod estudiante no se puede colocar ninguna letra porque si sucede esto nos mostrará un mensaje como el que se encuentra en la parte inferior, además si el usuario insiste en colocar letras la inserción de los datos no se podrá realizar debido a que no le permitirá avanzar hasta que coloque sólo números. Si el estudiante a matricular no se encuentra registrado como estudiante el coordinador observará el mensaje que viene a continuación.



FIGURA 31



FIGURA 32

Si algún cuadro de texto queda vacío en el formulario al momento de ingresar una matricula se mostrará este mensaje con el nombre del campo que debe llenar. Dependiendo al campo que quede vacío se mostrará este mensaje con el nombre del campo que debe llenar.



FIGURA 33

- **Cancelar:** Con este botón el usuario limpiará todos los campos de formulario ya se para realizar un nueva búsqueda o ingresar datos.

2.3.4 MODIFICACIÓN DE PARÁMETROS

En está parte el coordinador podrá seleccionar del enlace **Parámetro** cualquiera de las opciones que contiene el submenú que se despliega de dicho link que son: **Países, Departamentos y Ciudades**, en la cual realizara operaciones de insertar, actualizar, borrar o consultar, según sea su necesidad.



FIGURA 34

2.3.4.1 MODIFICACIÓN DEL PARÁMETRO PAÍSES

En esta parte de la aplicación el usuario registrará los países de procedencia de los estudiantes y profesores pertenecientes a la institución. El formulario **Modificación de Países** contiene dos campos que son **Código** que es donde se le coloca un número de identificación al país y el otro es **Nombre**.



FIGURA 35



- **Consultar:** presionando este botón el coordinador podrá visualizar los países de procedencia de los estudiantes y profesores de la institución, si la consulta requiere ser más específica se debe escribir parte de los datos a buscar en los campos pertenecientes al formulario anteriormente mostrado ya sea el código o el nombre del país y como resultado nos presentará lo siguiente:



FIGURA 36

- **Editar:** Este enlace le permitirá enviar los datos de un país en específico al formulario **Modificación de Países** para que se hagan las respectivas actualizaciones, cuando estos datos son transportados automáticamente el botón **Actualizar** se activa para que el coordinador pueda modificar la información, si el usuario no pasa los datos al formulario el botón **Actualizar** nunca se activará y por lo tanto no realizará ninguna modificación.
- **Borrar:** Con este enlace el coordinador podrá borrar cualquier país. Cuando el usuario de click en este link el registro del país que selecciono se borrará siempre y cuando le de aceptar en el mensaje de confirmación que se muestra a continuación:



FIGURA 37

Si la opción seleccionada fue aceptar borrará inmediatamente el registro del país seleccionado.



- **Ingresar:** Con este botón él coordinador ingresará todos los datos correspondiente a el país, la información que se digitalice tiene que ir de acuerdo al tipo de dato, por ejemplo en el campo código no se puede colocar ninguna letra porque si sucede esto nos mostrará un mensaje como el que se encuentra en la parte inferior, además si el usuario insiste en colocar letras la inserción de los datos no se podrá realizar debido a que no le permitirá avanzar hasta que coloque sólo números.

FIGURA 38



Si en el campo Nombre el usuario coloca números le saldrá este mensaje

FIGURA 39



Si el usuario ingresa un registro ya existente le mostrara el siguiente mensaje.

FIGURA 40



Si algún cuadro de texto queda vacío en el formulario al momento de ingresar un país se mostrará este mensaje con el nombre del campo que debe llenar. Dependiendo al campo que quede vacío se mostrará este mensaje con el nombre del campo que debe diligenciar

FIGURA 41



Campos

- Código
- Nombre

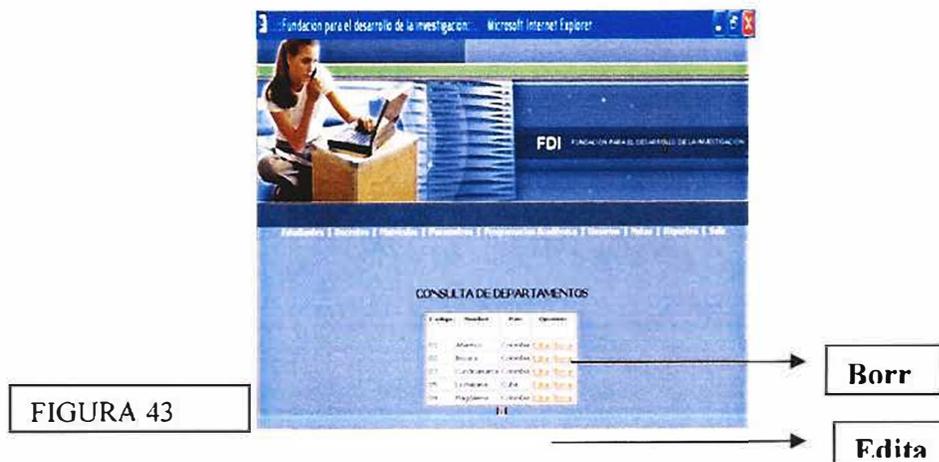
Cancelar: Con este botón el usuario limpiará todos los campos de formulario ya se para realizar un nueva búsqueda o ingresar un dato.

2.3.4.2 MODIFICACIÓN DEL PARÁMETRO DEPARTAMENTOS

En esta parte de la aplicación el usuario registrará los departamentos de procedencia de los estudiantes y profesores pertenecientes a la institución. El formulario **Modificación de Departamentos** contiene tres campos que son **Código** que es donde se le coloca un número de identificación del departamento, el **Nombre** y el **País** al cual pertenece dicho departamento.



- **Consultar:** presionando este botón el coordinador podrá visualizar los departamentos de procedencia de los estudiantes y profesores de la institución, si la consulta requiere ser más específica se debe escribir parte de los datos a buscar en los campos pertenecientes al formulario anteriormente mostrado ya sea el código o el nombre del departamento y como resultado nos presentará lo siguiente:



Editar: Este enlace le permitirá enviar los datos de un departamento en específico al formulario **Modificación de Departamentos** para que se hagan las respectivas actualizaciones, cuando estos datos son transportados automáticamente el botón **Actualizar** se activa para que el coordinador pueda modificar la información, si el usuario no pasa los datos al formulario el botón **Actualizar** nunca se activará y por lo tanto no realizará ninguna modificación.

- **Borrar:** Con este enlace el coordinador podrá borrar cualquier departamento. Cuando el usuario de click en este link el registro del departamento que selecciono se borrará siempre y cuando le de aceptar en el mensaje de confirmación que se muestra a continuación:



Si la opción seleccionada fue aceptar borrará inmediatamente el registro del departamento seleccionado.

- **Ingresar:** Con este botón el coordinador ingresará todos los datos correspondiente a el departamento, la información que se digitalice tiene que ir de acuerdo al tipo de dato, por ejemplo en el campo código no se puede colocar ninguna letra porque si sucede esto nos mostrará un mensaje como el que se encuentra en la parte inferior, además si el usuario insiste en colocar letras la inserción de los datos no se podrá realizar debido a que no le permitirá avanzar hasta que coloque sólo números.



Si el usuario en los campos alfabéticos intenta colocar algún número le saldrá el siguiente mensaje:



FIGURA 46

Si el usuario ingresa un registro ya existente le mostrara el siguiente mensaje.



FIGURA 47

Si algún cuadro de texto queda vacío en el formulario al momento de ingresar un departamento se mostrará este mensaje con el nombre del campo que debe llenar. Dependiendo al campo que quede vacío se mostrará este mensaje con el nombre del campo que debe diligenciar



FIGURA 48

Campos

- Codigo
- Nombre
- Pais

- **Cancelar:** Con este botón el usuario limpiará todos los campos del formulario ya sea para realizar un nueva búsqueda o ingresar un dato.

2.3.4.3 MODIFICACIÓN DEL PARÁMETRO CIUDADES

En esta parte de la aplicación el usuario registrará las ciudades con sus respectivos departamentos y países de procedencia de los estudiantes y profesores pertenecientes a la institución. El formulario **Modificación de Ciudades** contiene cuatro campos que son **Código** que es donde se le coloca un número de

identificación de la ciudad , **Nombre** correspondiente al nombre de la ciudad, **País** en donde se selecciona el nombre del país al cual pertenece dicha ciudad y dependiendo de este campo se despliega los departamento que conforman al país anteriormente seleccionado en el campo **Departamento**.



FIGURA 49

- **Consultar:** presionando este botón el coordinador podrá visualizar las ciudades de procedencia de los estudiantes y profesores de la institución, si la consulta requiere ser más específica se debe escribir parte de los datos a buscar en los campos pertenecientes al formulario anteriormente mostrado ya sea el código o el nombre de la ciudad y como resultado nos presentará lo siguiente:

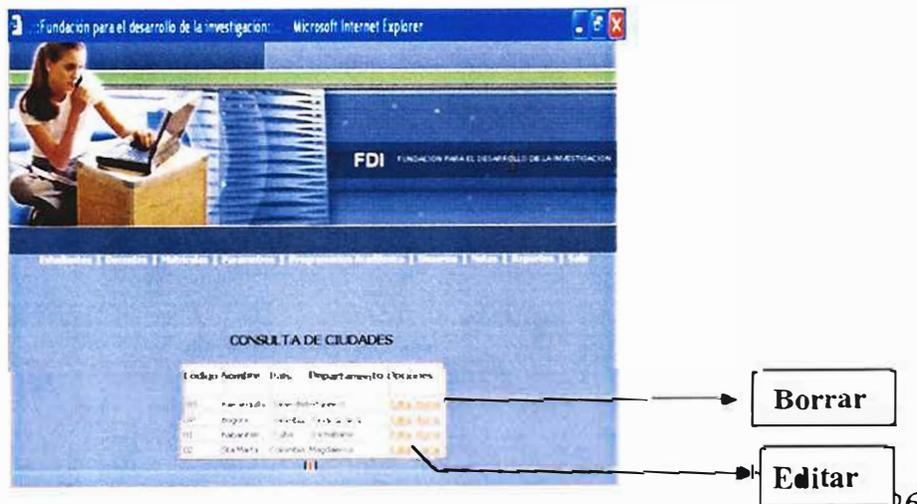


FIGURA 50



- **Editar:** Este enlace le permitirá enviar los datos de una ciudad en específico al formulario **Modificación de Ciudades** para que se hagan las respectivas actualizaciones, cuando estos datos son transportados automáticamente el botón **Actualizar** se activa para que el coordinador pueda modificar la información, si el usuario no pasa los datos al formulario el botón **Actualizar** nunca se activará y por lo tanto no realizará ninguna modificación.
- **Borrar:** Con este enlace el coordinador podrá borrar cualquiera ciudad. Cuando el usuario de click en este link el registro de la ciudad que selecciono se borrará siempre y cuando le de aceptar en el mensaje de confirmación que se muestra a continuación:



Si la opción seleccionada fue aceptar borrará inmediatamente el registro de la ciudad seleccionada.

- **Ingresar:** Con este botón él coordinador ingresará todos los datos correspondiente a la ciudad, la información que se digitalice tiene que ir de acuerdo al tipo de dato, por ejemplo en el campo código no se puede colocar ninguna letra porque si sucede esto nos mostrará un mensaje como el que se encuentra en la parte inferior, además si el usuario insiste en colocar letras la inserción de los datos no se podrá realizar debido a que no le permitirá avanzar hasta que coloque sólo números.



Si el usuario en los campos alfabéticos intenta colocar algún número le saldrá el siguiente mensaje:



FIGURA 53



Si el usuario ingresa un registro ya existente le mostrara el siguiente mensaje.

FIGURA 54



Si algún cuadro de texto queda vacío en el formulario al momento de ingresar una ciudad se mostrará este mensaje con el nombre del campo que debe llenar. Dependiendo al campo que quede vacío se mostrará este mensaje con el nombre del campo que debe diligenciar

FIGURA 55

**Campos**

- Código
- Nombre
- País
- Departamento

- **Cancelar:** Con este botón el usuario limpiará todos los campos de formulario ya se para realizar un nueva búsqueda o ingresar un dato.

2.3.4.4 MODIFICACIÓN DEL PARÁMETRO LOGS

Con el formulario **Logs de Eventos** el coordinador académico podrá verificar cuales son los cambios que se han realizado con respecto a los datos que se manejan en la aplicación, dicho formulario cuenta con tres campos muy importantes que son: **Importar**, **Exportar** y **Cancelar** que nos ayuda a llevar un mayor control sobre la información del FDI , por ejemplo si el usuario ingresa a un alumno a través de formulario estudiante en el Logs de evento aparecerá en registro detallado de los movimientos que se realizaron, aparte de logs de evento también se puede **Importar** y **Exportar** todos los datos pertenecientes a **Estudiantes, Docentes ,Matriculas, País, Departamentos, Ciudad, Programas, Materias y Notas**. A continuación se muestra la pantalla por la cual se maneja el Logs de evento.

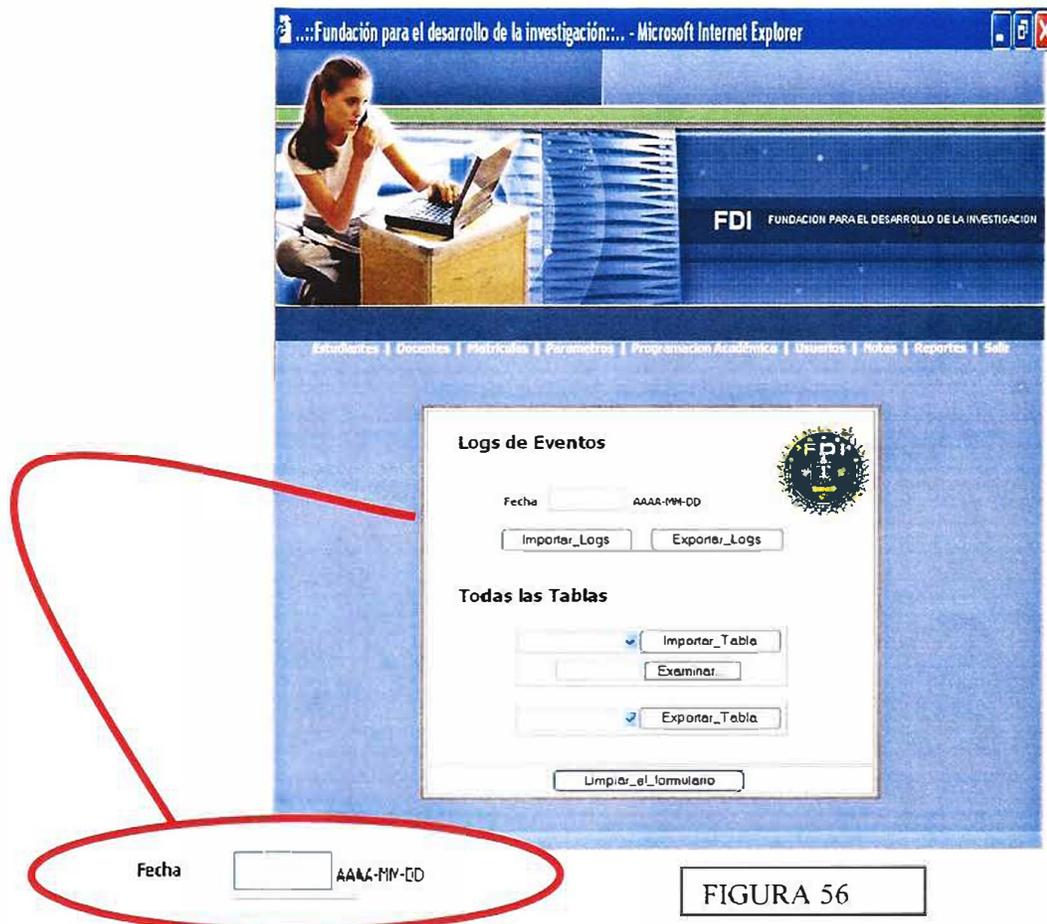


FIGURA 56

La sección resaltada hace referencia al campo **Fecha** en el cual sólo se pueden colocar fecha con el formato AAAA-MM-DD porque si el usuario coloca una fecha que no cumpla con este formato visualizará el mensaje que se encuentra en la parte inferior, si el usuario insiste en colocar una fecha inadecuada no podrá avanzar, este campo sólo se utiliza cuando se va a presionar el botón **Importar** pertenecientes a la parte de logs de eventos.

FIGURA 57



- **Exportar (Logs de Eventos):** Dando click a este botón el usuario podrá enviar a un archivo plano todos los movimientos que se realizaron en un día con sus respectivas hora y fecha, este archivo tendrá como nombre la fecha del día que se presiono el botón para realizar la exportación . Cuando los datos son transportados se observara el siguiente mensaje.

FIGURA 58



Después de haber presionado el botón de **Exportar** se visualizará el mismo formulario de **Logs de Evento** pero con un links en la parte inferior de los botones **Exportar** e **Importar** que le permitirá al usuario ver el archivo de texto que contiene los datos exportado, a continuación se muestra una imagen que contiene dicho formulario

FIGURA 59



Cuando el usuario presione el enlace de [Descarga de la Exportación de Logs](#) observará la siguiente ventana de descarga que le permitirá abrir, guardar o cancelar según sea su necesidad.

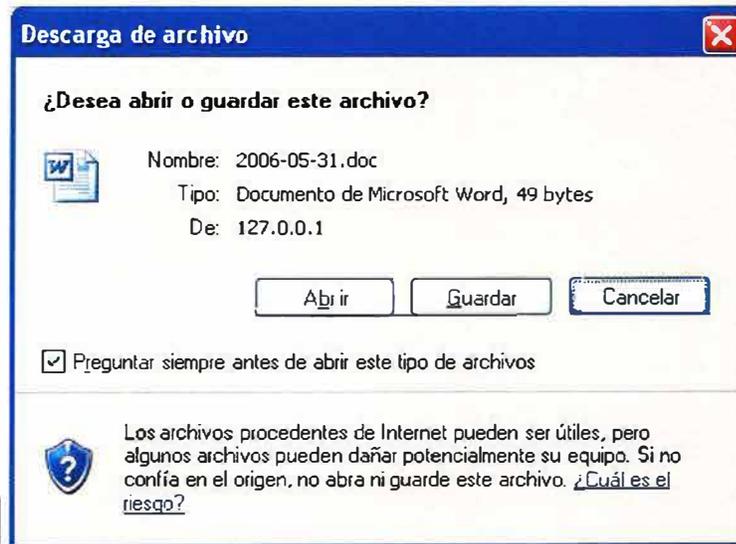


FIGURA 60

- **Importar (Logs de Eventos):** Llenando el campo **Fecha** con el dato correcto el usuario podrá visualizar en la tabla ubicada en la parte inferior toda la información almacenada en el archivos plano con la fecha que digito en el campo mencionado.

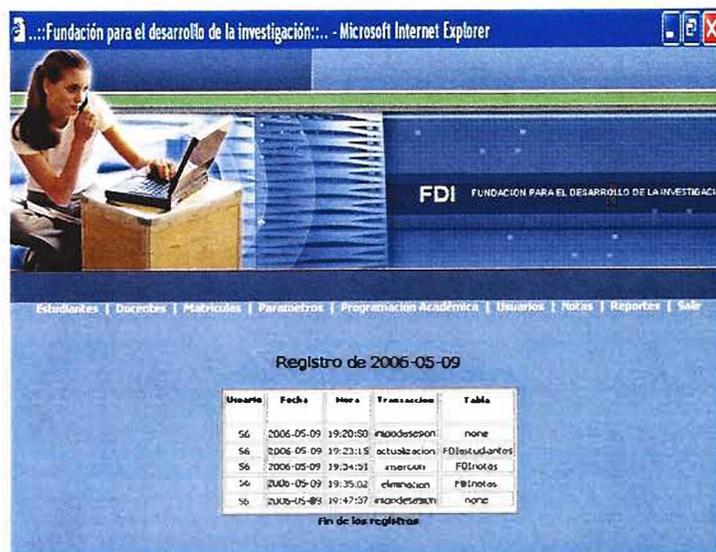


FIGURA 61

En la parte de **Todas las Tablas** encontramos los botones de **Importar_Tabla**, **Examinar**, **Exportar_Tabla**, **Limpiar_el_Formulario** y dos campos desplegables que permiten al usuario seleccionar los datos de las siguientes tablas: **Estudiantes**, **Docentes**, **Matriculas**, **País**, **Departamentos**, **Ciudad**, **Programas**, **Materias** y **Notas**, después de haber escogido la tabla se **Exportar** o **Importar** la información de la tabla elegida.

- **Importar_Tabla:** Con este botón el coordinador seleccionará la tabla a la cual se va a escribir en la base de datos la información almacenada en el archivo plano perteneciente a dicha tabla. Si el usuario presiona este botón sin haber escogido una tabla visualizará el siguiente mensaje.



FIGURA 62

- **Examinar:** Este botón permitirá seleccionar el archivo plano que se quiera importar a la base de datos a través de la siguiente ventana

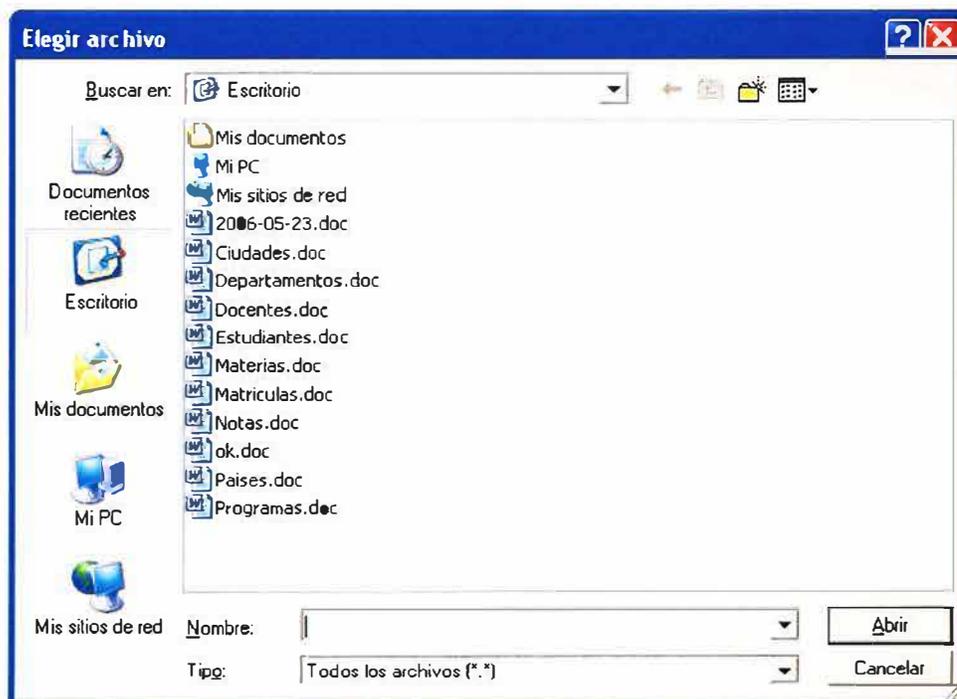


FIGURA 63



Cuando se presiona **Examinar** sin colocar la ruta del archivo a **Importar** se muestra el mensaje que se observa en la parte inferior, el usuario tiene que seleccionar el archivo porque de la contrario no podrá ejecutar la importación.



FIGURA 64

- **Exportar:** Con este botón el usuario enviará a un archivo plano toda la información almacenada en la tabla seleccionada del campo desplegable que se encuentra en la parte izquierda de este botón. Al presionar **Exportar** se activará un link en la parte inferior del formulario que al presionarlo abrirá la ventana que se encuentra en la parte inferior, en la que se pueda abrir, guardar o cerrar el archivo correspondiente a la tabla que se selecciona.

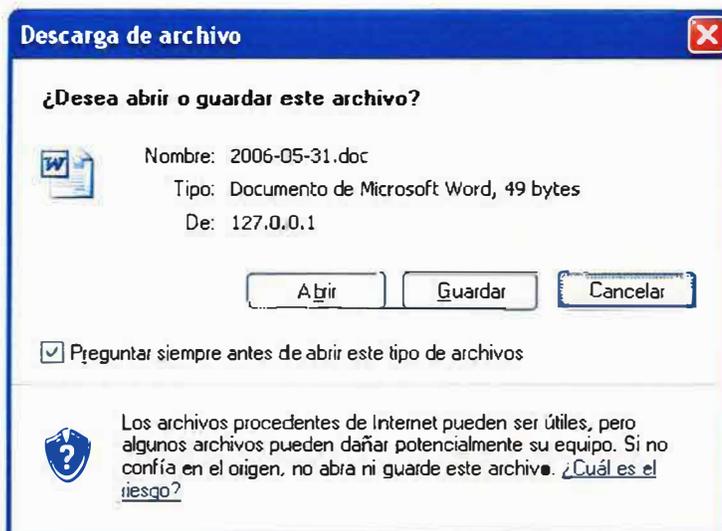


FIGURA 65

Cuando se presiona exportar sin seleccionar la tabla se visualiza este mensaje



FIGURA 66

Cancelar: Con este botón el usuario limpiará todos los campos de formulario ya se para realizar un nueva búsqueda o ingresar un dato.

2.3.5 MODIFICACIÓN DE PROGRAMACIÓN ACADÉMICA

En esta parte el coordinador podrá seleccionar del enlace **Programación Académica** que se visualiza en la parte inferior un submenú que contiene las opciones de **Programas**, **Pensums**, **Cursos** y **Materias** en la cual realizara operaciones de insertar, actualizar, borrar o consultar, según sea su necesidad.

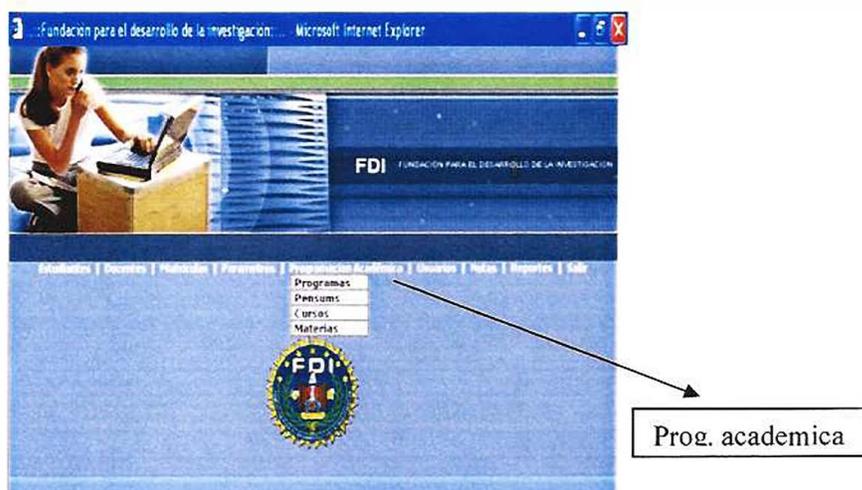


FIGURA 67

2.3.51 MODIFICACIÓN DE PROGRAMACIÓN ACADÉMICA/PROGRAMAS

En esta parte de la aplicación el coordinador académico registrará todos los programas que se vayan a dictar en el FDI. En este formulario contiene cinco campos que son: **Código** que es en donde se le coloca el número por el cual se va a identificar el programa, **Nombre** perteneciente al nombre del programa académico como tal, **Código Icfes** correspondiente al número de aprobación que otorga el Icfes a una carrera, **Fecha de creación** esta es la fecha en la que se creó la carrera y **Duración** que hace referencia al tiempo que va a durar esta, la imagen que encontramos en la parte inferior corresponde al formulario anteriormente mencionado.



FIGURA 68

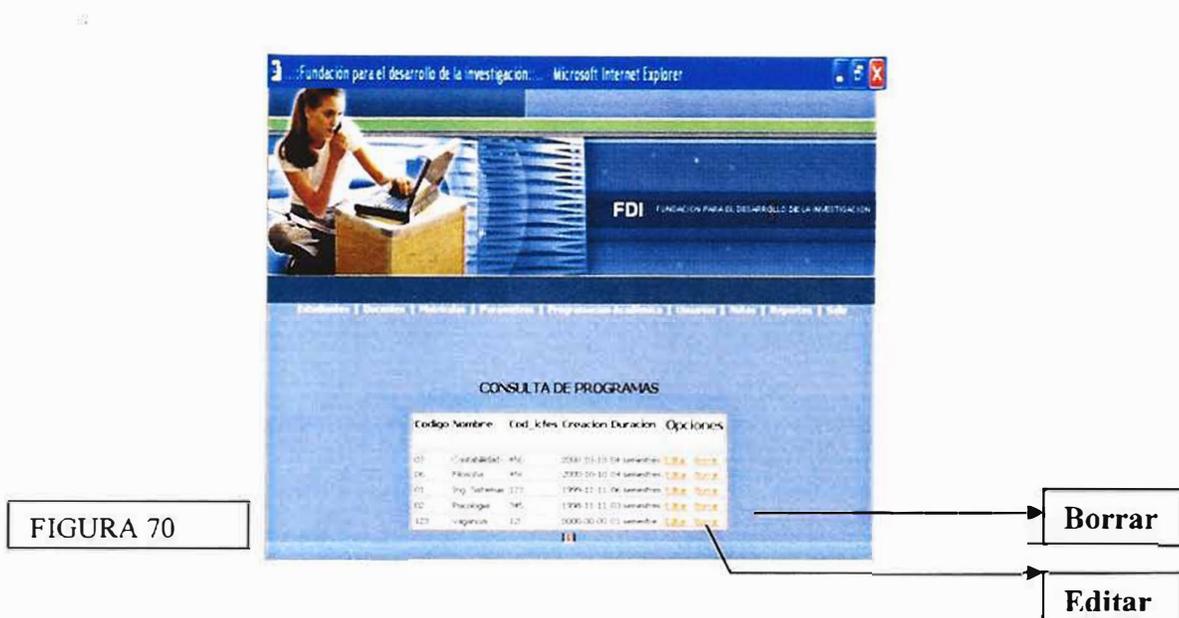
La sección resaltada hace referencia al campo **Fecha de creación** en el cual sólo se pueden colocar fecha con el formato AAAA-MM-DD porque si el usuario coloca una fecha que no cumpla con este formato visualizará el mensaje que se encuentra en la parte inferior, si el usuario insiste en colocar una fecha inadecuada no podrá avanzar.



FIGURA 69

- **Consultar:** presionando este botón el coordinador podrá visualizar los diversos programas que dicta el FDI actualmente, si la consulta requiere ser más específica se debe escribir parte de los datos a buscar en los campos pertenecientes al formulario

anteriormente mostrado ya sea el código o el nombre del programa y como resultado nos presentará lo siguiente:



- **Editar:** Este enlace le permitirá enviar los datos de un programa en específico al formulario **Modificación de Programas** para que se hagan las respectivas actualizaciones, cuando estos datos son transportados automáticamente el botón **Actualizar** se activa para que el coordinador pueda modificar la información, si el usuario no pasa los datos al formulario el botón **Actualizar** nunca se activará y por lo tanto no realizará ninguna modificación.
- **Borrar:** Con este enlace el coordinador podrá borrar cualquiera programa. Cuando el usuario de click en este link el registro del programa que selecciono se borrará siempre y cuando le de aceptar en el mensaje de confirmación que se muestra a continuación:



Si la opción seleccionada fue aceptar borrará inmediatamente el registro del programa seleccionado.



- **Ingresar:** Con este botón él coordinador ingresará todos los datos correspondiente a el programa académico, la información que se digitalice tiene que ir de acuerdo al tipo de dato, por ejemplo en el campo código no se puede colocar ninguna letra porque si sucede esto nos mostrará un mensaje como el que se encuentra en la parte inferior, además si el usuario insiste en colocar letras la inserción de los datos no se podrá realizar debido a que no le permitirá avanzar hasta que coloque sólo números.

FIGURA 72



Si el usuario ingresa un registro ya existente le mostrara el siguiente mensaje.

FIGURA 73



Si algún cuadro de texto queda vacío en el formulario al momento de ingresar un programa se mostrará este mensaje con el nombre del campo que debe llenar. Dependiendo al campo que quede vacío se mostrará este mensaje con el nombre del campo que debe diligenciar

FIGURA 74



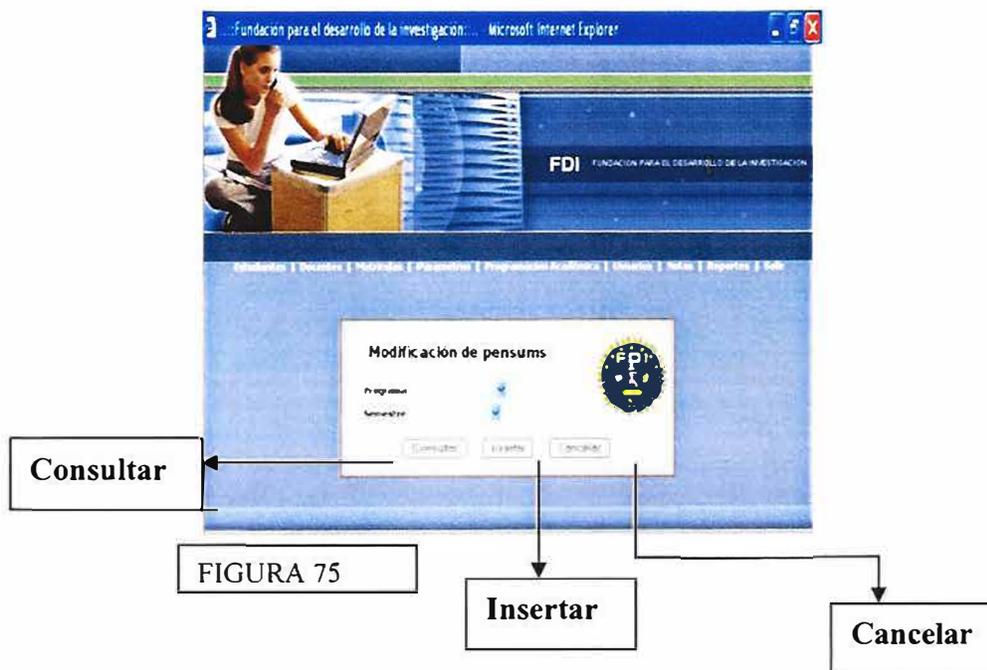
Campos

- Código
- Nombre
- Codigo icfes
- Fecha de creación
- Duracion

- **Cancelar:** Con este botón el usuario limpiará todos los campos de formulario ya se para realizar un nueva búsqueda o ingresar un dato.

2.3.5.2 MODIFICACIÓN DE PROGRAMACIÓN ACADÉMICA/PENSUMS

Con este formulario el coordinador académico podrá crear los diversos pensums con sus respectivas materias, este formulario consta de dos campos desplegables que son **Programa** que es donde se selecciona la carrera al cual se le va crear el pensum y **Semestre** que es el semestre al cual se le va asignar el pensums.



- **Consultar:** presionando este botón el coordinador podrá visualizar los diversos pensums con los que cuenta actualmente FDI, si la consulta requiere ser más específica se debe escribir parte de los datos a buscar en los campos pertenecientes al formulario anteriormente mostrado ya sea **Programa** o **Semestre** como resultado nos presentará lo siguiente:

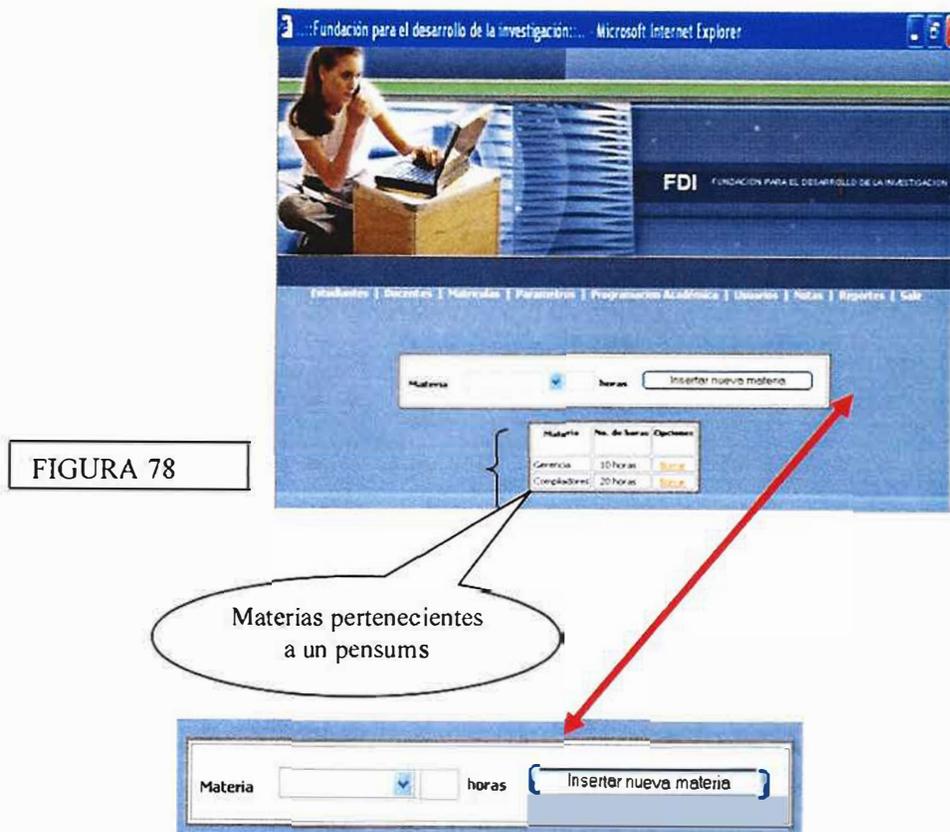


- **Borrar:** Con este enlace el coordinador podrá borrar cualquier pensum perteneciente a un determinado programa. Cuando el usuario de click en este link el registro del pensum que selecciono se borrará siempre y cuando le de aceptar en el mensaje de confirmación que se muestra a continuación:



Si la opción seleccionada fue aceptar borrará inmediatamente el registro del pensum que selecciono.

- **Materias:** con este link el usuario podrá asignar/consultar las diferentes materias perteneciente al programa académico que haya seleccionado, a través del siguiente formulario.



En esta parte de la aplicación cuando el usuario de click en el botón **Insertar nueva materia** podrá registrar las materias que requiera cualquier programa académico a través de campo desplegable **Materia** y asignarle la intensidad horaria por medio del campo **Horas** Si estos campos no son diligenciados el usuario visualizará los siguientes mensajes.



- **Borrar:** Con este enlace el coordinador podrá borrar cualquiera materia perteneciente a un determinado pensum. Cuando el usuario de click en este link el registro de la materia que selecciono se borrará siempre y cuando le de aceptar en el mensaje de confirmación que se muestra a continuación:



FIGURA 81

Si la opción seleccionada fue aceptar borrará inmediatamente el registro de la materia que selecciono.

- **Ingresar:** Con este botón él coordinador ingresará todos los datos correspondiente a un determinado pensum académico.

Si el usuario ingresa un registro ya existente le mostrara el siguiente mensaje.



FIGURA 82

Si algún cuadro de texto queda vacío en el formulario al momento de ingresar un programa se mostrará este mensaje con el nombre del campo que debe llenar. Dependiendo al campo que quede vacío se mostrará este mensaje con el nombre del campo que debe diligenciar

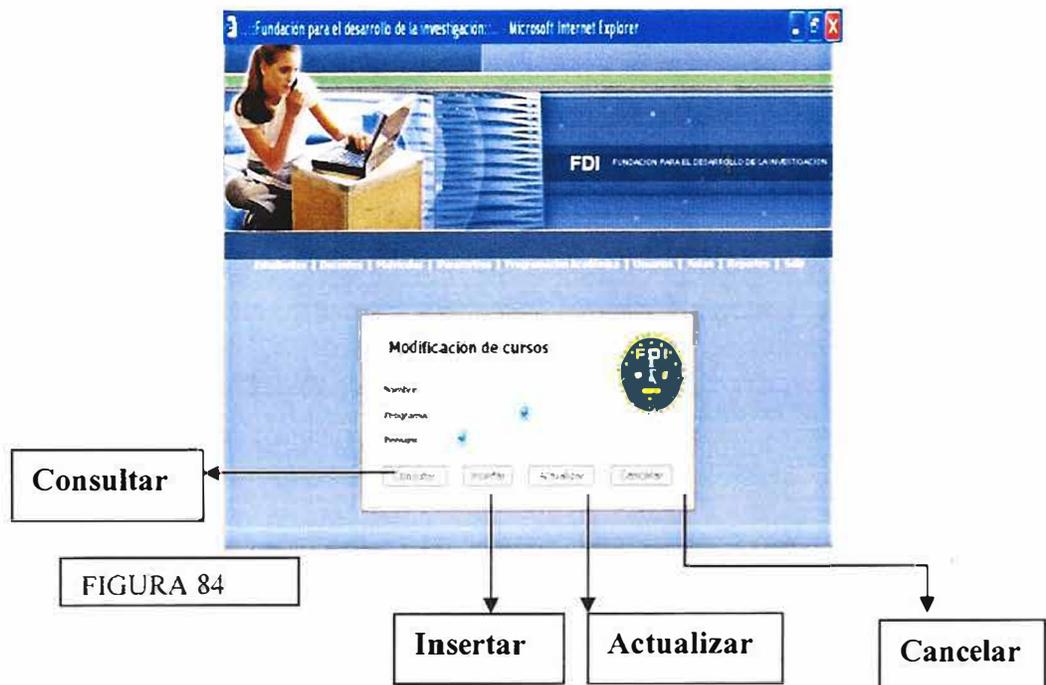


FIGURA 83

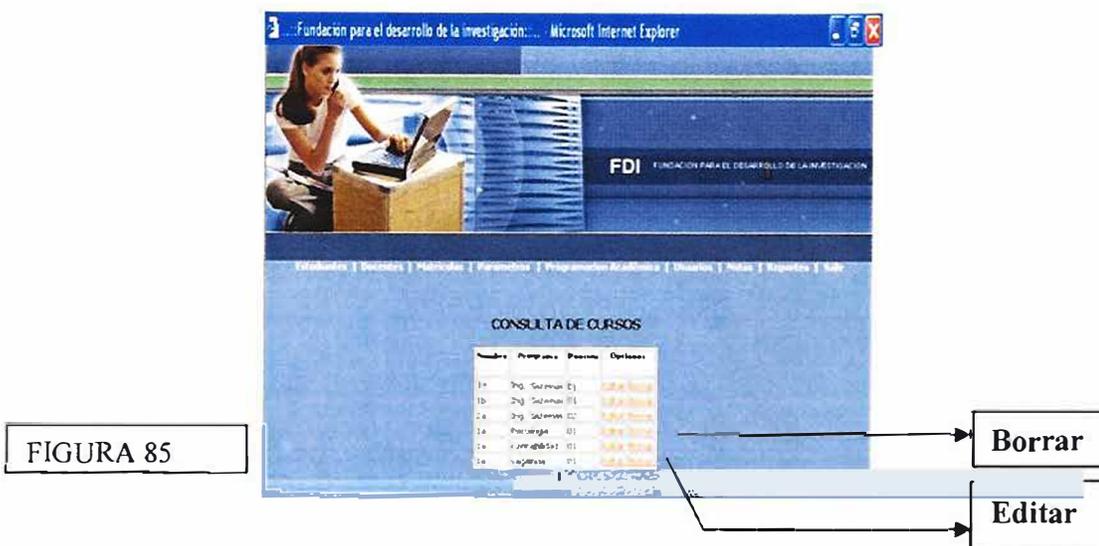
- **Cancelar:** Con este botón el usuario limpiará todos los campos de formulario ya se para realizar un nueva búsqueda o ingresar un dato.

2.3.5.3 MODIFICACIÓN DE PROGRAMACIÓN ACADÉMICA/CURSOS

En esta parte de la aplicación el usuario podrá crear los cursos correspondientes a cada semestre de los programas del FDI, el formulario **Modificación de Cursos** cuenta con tres campos que son **Nombre** que es el lugar en donde se le da el nombre al curso por ejemplo primero A, **Programa** que es el campo en el cual se selecciona el programa académico que se le va a asignar el curso y **Semestre** que es el campo por el cual se selecciona el semestre de la carrera que se le va asignar el curso.



- **Consultar:** presionando este botón el coordinador podrá visualizar los diversos curso con los que cuenta FDI actualmente, si la consulta requiere ser más específica se debe escribir parte de los datos a buscar en los campos pertenecientes al formulario anteriormente mostrado ya sea el nombre o cualquier otro dato y como resultado nos presentará lo siguiente:



- **Editar:** Este enlace le permitirá enviar los datos de un curso en específico al formulario **Modificación de Cursos** para que se hagan las respectivas actualizaciones, cuando estos datos son transportados automáticamente el botón **Actualizar** se activa para que el coordinador pueda modificar la información, si el usuario no pasa los datos al formulario el botón **Actualizar** nunca se activará y por lo tanto no realizará ninguna modificación.
- **Borrar:** Con este enlace el coordinador podrá borrar cualquiera curso. Cuando el usuario de click en este link el registro del curso que selecciono se borrará siempre y cuando le de aceptar en el mensaje de confirmación que se muestra a continuación:



Si la opción seleccionada fue aceptar borrará inmediatamente el registro el curso seleccionado.

- **Ingresar:** Con este botón él coordinador ingresará todos los datos correspondientes a el curso que desea crear.

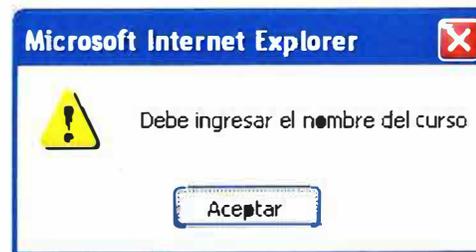
Si el usuario ingresa un registro ya existente le mostrara el siguiente mensaje.

FIGURA 87



Si algún cuadro de texto queda vacío en el formulario al momento de ingresar un curso se mostrará este mensaje con el nombre del campo que debe llenar. Dependiendo al campo que quede vacío se mostrará este mensaje con el nombre del campo que debe diligenciar

FIGURA 88



Campos

- Nombre
- Pregarma
- Pensum

- **Cancelar:** Con este botón el usuario limpiará todos los campos de formulario ya se para realizar un nueva búsqueda o ingresar un dato.

2.3.5.4 MODIFICACIÓN DE PROGRAMACIÓN ACADÉMICA/MATERIAS

En el formulario **Modificación de Materias** el coordinador creará todas las materias que contendrán las carreras del FDI, este formulario cuenta con dos campos que son **Código** que es la identificación de la materia y **Nombre** que es el nombre de la asignatura.



FIGURA 89

- **Consultar:** presionando este botón el coordinador podrá visualizar los diversas materias correspondientes a los diferentes programas, si la consulta requiere ser más específica se debe escribir parte de los datos a buscar en los campos pertenecientes al formulario anteriormente mostrado ya sea el nombre o cualquier otro dato y como resultado nos presentará lo siguiente:



- **Editar:** Este enlace le permitirá enviar los datos de una materia en específico al formulario **Modificación de Materias** para que se hagan las respectivas actualizaciones, cuando estos datos son transportados automáticamente el botón **Actualizar** se activa para que el coordinador pueda modificar la información, si el usuario no pasa los datos al formulario el botón **Actualizar** nunca se activará y por lo tanto no realizará ninguna modificación.
- **Borrar:** Con este enlace el coordinador podrá borrar cualquiera materia. Cuando el usuario de click en este link el registro de la materia que selecciono se borrará siempre y cuando le de aceptar en el mensaje de confirmación que se muestra a continuación:



Si la opción seleccionada fue aceptar borrará inmediatamente el registro de la materia seleccionada.

- **Ingresar:** Con este botón él coordinador ingresará todos los datos correspondientes a la materia que desea crea, la información que se digitalice tiene que ir de acuerdo al tipo de dato, por ejemplo en el campo código no se puede colocar ninguna letra porque si sucede esto nos mostrará un mensaje como el que se encuentra en la parte inferior, además si el usuario insiste en colocar letras la inserción de los datos no se podrá realizar debido a que no le permitirá avanzar hasta que coloque sólo números.

FIGURA 92



Si el usuario ingresa un registro ya existente le mostrara el siguiente mensaje.

FIGURA 93



Si algún cuadro de texto queda vacío en el formulario al momento de ingresar una materia se mostrará este mensaje con el nombre del campo que debe llenar. Dependiendo al campo que quede vacío se mostrará este mensaje con el nombre del campo que debe diligenciar

FIGURA 94



Campos

- Código
- Nombre

- **Cancelar:** Con este botón el usuario limpiará todos los campos de formulario ya se para realizar un nueva búsqueda o ingresar un dato.

2.3.6 MODIFICACIÓN DE USUARIOS

Con este formulario el coordinador le creará a todos los estudiantes y administradores del sistema su respectivo perfil, la forma **Modificación de Usuarios** cuenta con tres campos que son **Código** lugar en donde se coloca la identificación del estudiante/administrador, **Password** es la clave con la cual el estudiante/administrador van a tener acceso al sistema y **Role**, en este campo de selección se determina que persona va ser estudiante/administrador en el sistema.



- **Consultar:** presionando este botón el coordinador podrá visualizar los diversos usuarios que hacen parte del sistema, si la consulta requiere ser más específica se debe escribir parte de los datos a buscar en los campos pertenecientes al formulario anteriormente mostrado ya sea el código o cualquier otro dato y como resultado nos presentará lo siguiente:

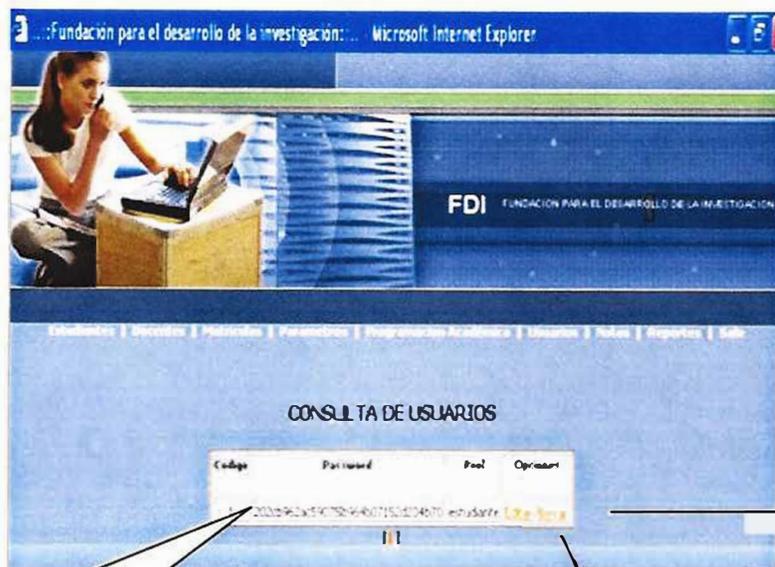


FIGURA 96

Estos valores están encriptado por seguridad.

- **Editar:** Este enlace le permitirá enviar los datos de un usuario en específico al formulario **Modificación de Usuarios** para que se hagan las respectivas actualizaciones, cuando estos datos son transportados automáticamente el botón **Actualizar** se activa para que el coordinador pueda modificar la información, si el usuario no pasa los datos al formulario el botón **Actualizar** nunca se activará y por lo tanto no realizará ninguna modificación.
- **Borrar:** Con este enlace el coordinador podrá borrar cualquier usuario. Cuando el usuario de click en este link el registro del usuario que selecciono se borrará siempre y cuando le de aceptar en el mensaje de confirmación que se muestra a continuación:



FIGURA 97

Si la opción seleccionada fue aceptar borrará inmediatamente el registro del usuario seleccionado.



- **Ingresar:** Con este botón el coordinador ingresará todos los datos correspondientes a los diferentes usuarios existentes en la institución para que puedan acceder al sistema.

Si el usuario ingresa un registro ya existente le mostrara el siguiente mensaje.



Si algún cuadro de texto queda vacío en el formulario al momento de ingresar un usuario se mostrará este mensaje con el nombre del campo que debe llenar. Dependiendo al campo que quede vacío se mostrará este mensaje con el nombre del campo que debe diligenciar



- **Cancelar:** Con este botón el usuario limpiará todos los campos de formulario ya se para realizar un nueva búsqueda o ingresar un dato.

2.3.7 NOTAS

En este enlace el usuario encontrará dos links que son: **Consulta por Curso** y **Consulta por Estudiante** estos enlaces le permitirá al coordinador manejar todo la concerniente a inserciones, actualizaciones, asignación de materias perdidas y borrar registros relacionados con las notas, en la parte inferior se muestra una ventana en donde se observan estos dos links.



FIGURA 100

2.3.7.1 NOTAS/ CONSULTA POR CURSO

En esta parte de la aplicación el coordinador ingresará las notas de todos le estudiante pertenecientes a un determinado curso, este formulario contiene los siguiente campos: **Año, Período, Programa, Curso, Materia** y **Nota**. En la consulta hay que tener en cuenta que en el campo **Nota** hay que seleccionar la nota (1,2,3) que se quiera insertar, actualizar, guardar y editar. A continuación se muestra el formulario de **Modificación de Notas**.



FIGURA 101



Ante de presionar el botón de **Consultar** todos los campos de formulario **Modificación de Notas** tienen que estar llenos porque de lo contrario se observarán el siguiente mensajes de advertencia dependiendo al campo que se encuentre vacío



FIGURA 102

- **Consultar:** presionando este botón el coordinador podrá consultar, insertar, guardar y editar las diversas notas de los estudiantes pertenecientes a un curso en específico, a continuación se observa la imagen de la consulta de notas.
- **Cancelar:** Con este botón el usuario limpiará todos los campos de formulario ya se para realizar un nueva búsqueda o ingresar un dato.

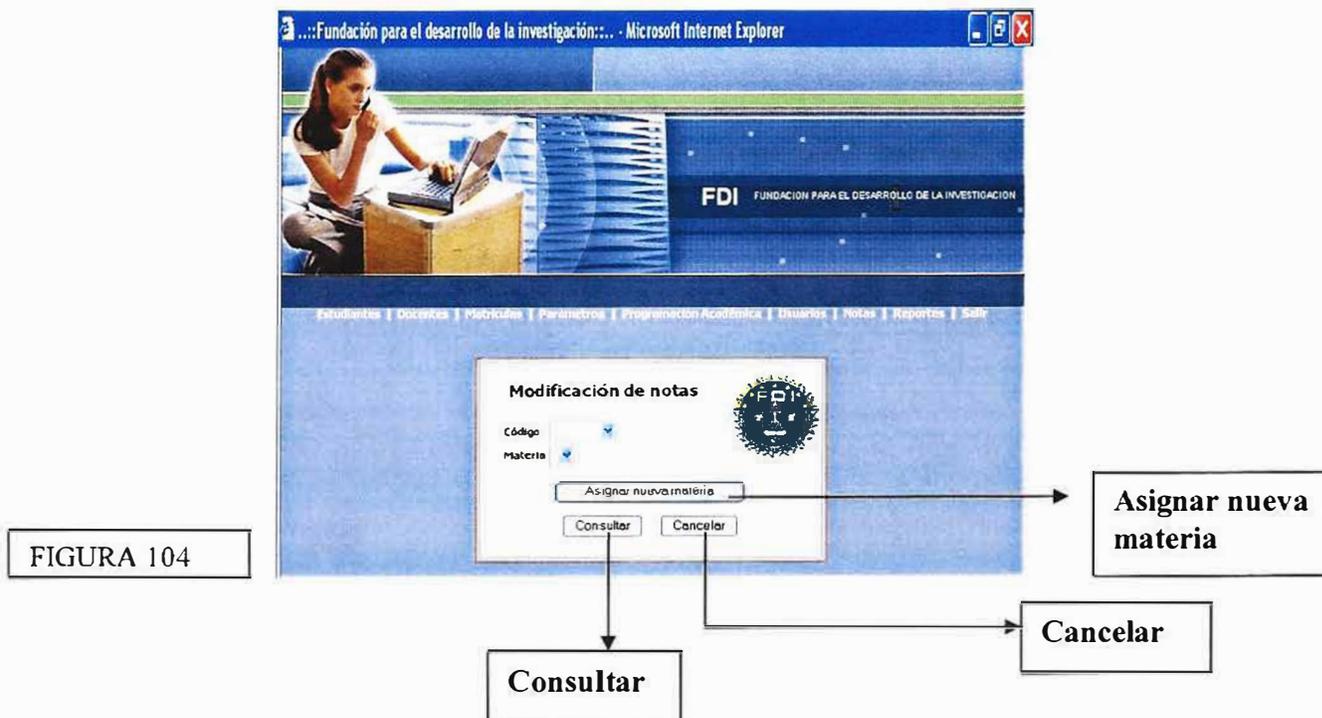


FIGURA 103

- **Guardar:** Con este botón el usuario podrá almacenar todas aquellas inserciones o ediciones que se realicen con las notas.

2.3.7.2 NOTAS / CONSULTA POR ESTUDIANTE

En esta parte de la aplicación el coordinador ingresará las materias que tiene que repetir el estudiante, el formulario **Modificación de Notas** contiene dos campos desplegables que son **Código** que es la identificación correspondiente a la persona que perdió la asignatura y **Materia** que es el campo en donde se escribirá el nombre de la materia no aprobada, a continuación se muestra el formulario por la cual se realizará la asignación de materias reprobadas.



Hay que tener en cuenta que tiene que haber previamente estudiantes y materias registradas para que en la lista desplegable puedan ver las diversas opciones.

- **Consultar:** Con este botón el coordinador podrá mirar las notas(1,2,3), materias, período y año correspondiente a un estudiante en particular en una tabla localizada en en la parte inferior del formulario **Modificación de Notas**



FIGURA 105

- **Cancelar:** Con este botón el usuario limpiará todos los campos de formulario ya se para realizar un nueva búsqueda o ingresar un dato.
- **Asignar nueva materia:** Con este botón el usuario se transportará al formulario que se encuentra en la parte inferior para así pueda realizar la asignación de las materias perdidas a un estudiantes un específico. Si no se selecciona ningún código se visualizará el siguiente mensaje de advertencia.



FIGURA 106

Quando la persona haya seleccionado el código al cual le quiere asignarle la materia el usuario observará la siguiente imagen.

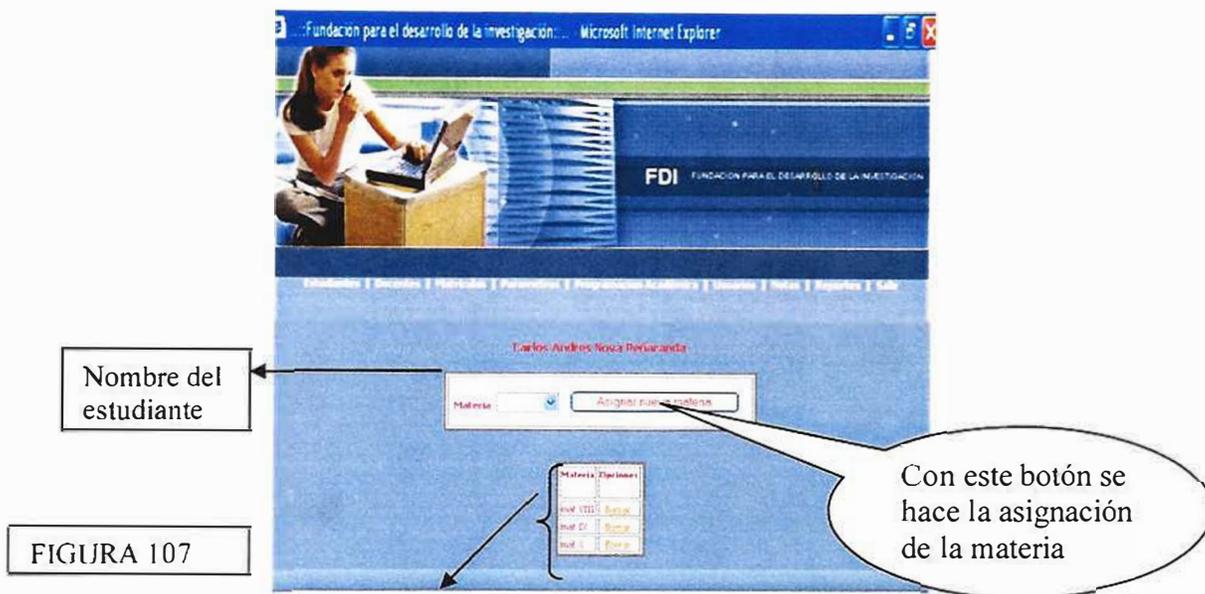


FIGURA 107

Estas son las materias que actualmente el estudiante esta viendo y adicionalmente el coordinador le agregara la materia que haya perdido. Si en el campo materia perteneciente a la imagen anterior no se ingresa ninguna materia el usuario observará este mensaje.



FIGURA 108

- **Borrar:** Con este enlace el coordinador podrá borrar cualquiera materia. Cuando el usuario de click en este link el registro de la materia que selecciono se borrará siempre y cuando le de aceptar en el mensaje de confirmación que se muestra a continuación:



FIGURA 109

Si la opción seleccionada fue aceptar borrará inmediatamente el registro de la materia seleccionada.

2.3.8 REPORTEES

En esta parte de la aplicación el coordinador encontrará un enlace llamado **Reportes** que contendrá los siguientes links: **Lista de asistencia**, **Reporte de notas** y **Reportes Dinámicos** estos dos enlaces les permitirán generar reportes de una manera fácil, eficiente y sobre todo hacer un mejor uso de todos aquellos recursos con los que cuenta la institución, a continuación se muestra la pantalla que contiene los enlaces anteriormente mencionados.

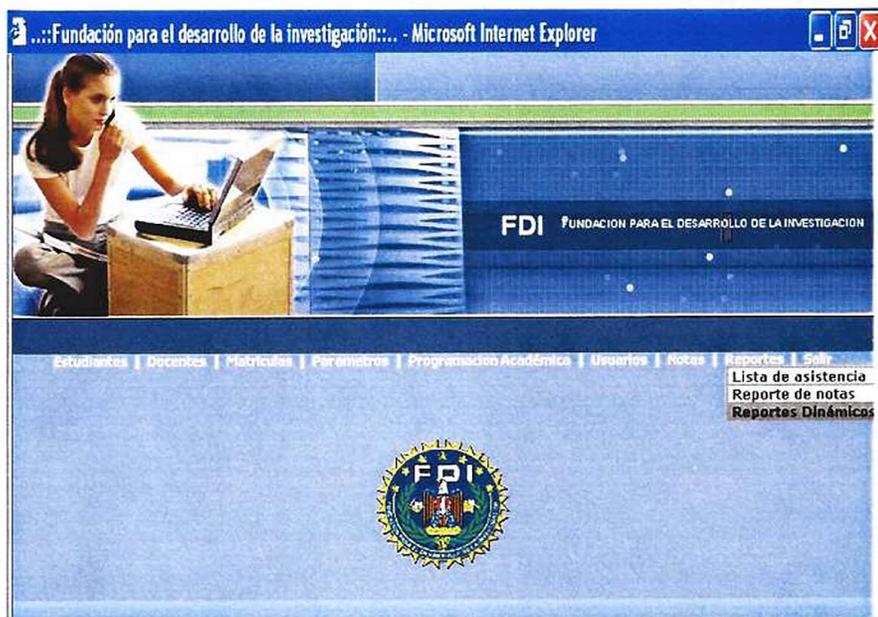


FIGURA 110

2.3.8.1 REPORTEES/ LISTA DE ASISTENCIA

En este formulario el usuario podrá generar todas las listas pertenecientes a los diferentes programas con sus respectivos estudiantes, dichos listados serán entregados a los profesores para que lleve un control de la asistencia a clase de los estudiantes. Este formulario contiene cuatro campos de selección que son: **Programas**, **Cursos**, **Año** y **Período**, hay que tener en cuenta que tienen que haber programas y cursos existentes en la base de datos para que se pueda hacer las

selecciones. A continuación se muestra la ventana que contiene el formulario de **Lista de Asistencia**



- **Generar_Reportes:** Con este botón el coordinador visualizará una ventana que contiene la lista de los estudiantes basada en los datos que digito en el formulario **Lista de asistencia**. Esta es la ventana que verán cuando den click a este botón.



2.3.8.2 REPORTES DE NOTAS

Con este formulario el coordinador podrá ingresar los parámetros necesarios para generar el **Reporte de Notas** de un estudiante en específico. Esta forma contiene los campos de **Cod.estudiante**, **Programas**, **Año** y **Período**, de los campos anteriormente mencionados lo que son **Programas** y **Período** son de selección por lo cual para que su puedan ver las opciones se deben insertar dicho datos en los formularios correspondientes a programas y períodos previamente. A continuación verán la imagen correspondiente.

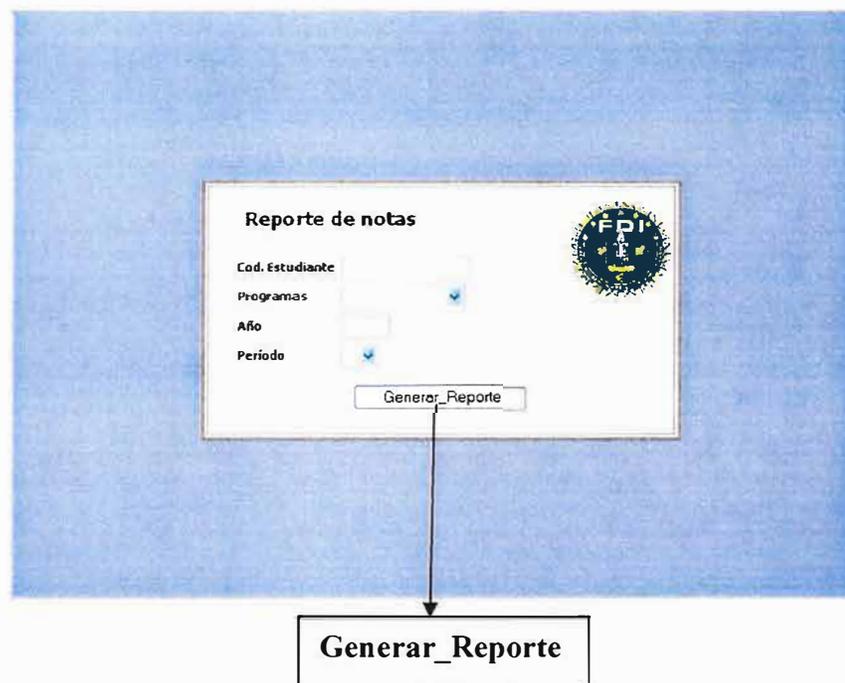
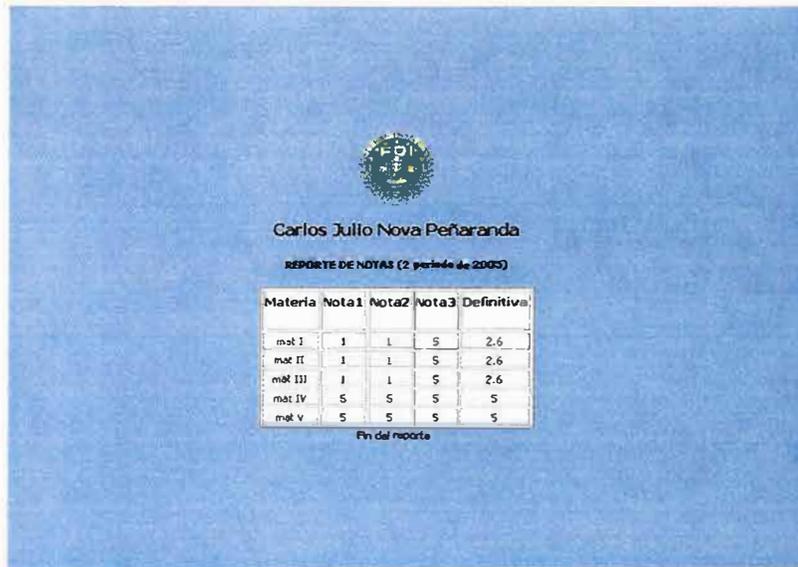


FIGURA 113

- **Generar Reportes:** Dando clic a este botón el usuario observará la siguiente ventana que contiene el nombre completo del estudiante, período, año, las materias que ve y primera, segunda, tercera y nota definitiva.



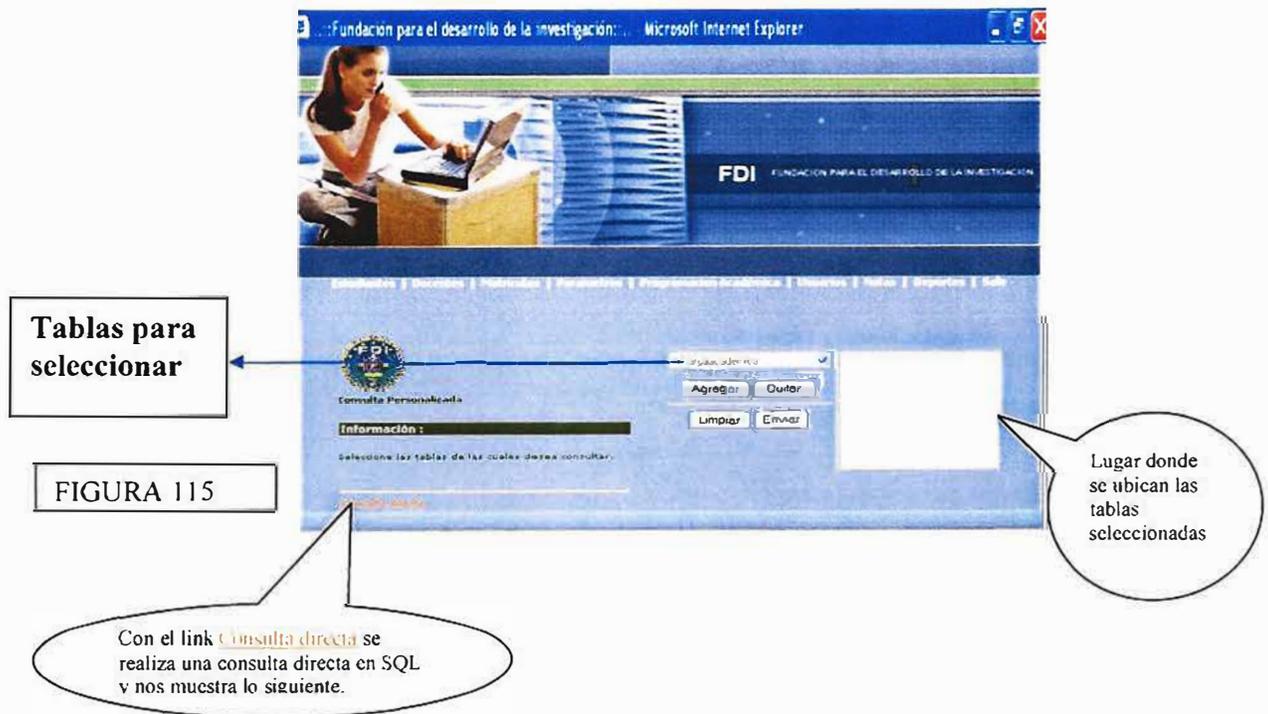
| Materia | Nota1 | Nota2 | Nota3 | Definitiva |
|---------|-------|-------|-------|------------|
| mat I | 1 | 1 | 1 | 2.6 |
| mat II | 1 | 1 | 1 | 2.6 |
| mat III | 1 | 1 | 1 | 2.6 |
| mat IV | 5 | 5 | 5 | 5 |
| mat V | 5 | 5 | 5 | 5 |

Fin del reporte

FIGURA 114

2.3.8.3 REPORTE DINÁMICOS

En esta parte de la aplicación el coordinador podrá generar sus informes según sea su necesidad seleccionando las tablas y los campos de dicha tabla que sean requeridos para realizar la consulta solicitada, a continuación se muestra un pantallazo en donde se realizará la selección de las tablas.



Tablas para seleccionar

FIGURA 115

Lugar donde se ubican las tablas seleccionadas

Con el link [Consulta directa](#) se realiza una consulta directa en SQL y nos muestra lo siguiente.

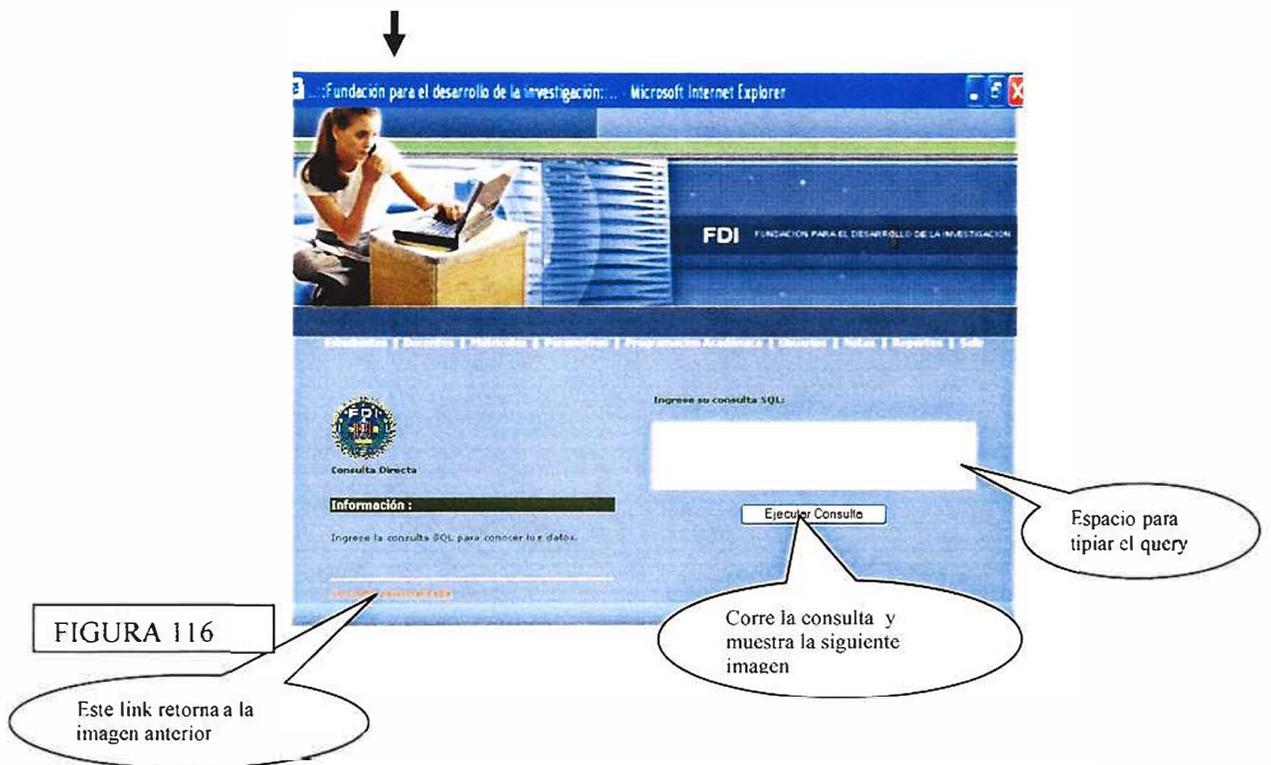


FIGURA 116

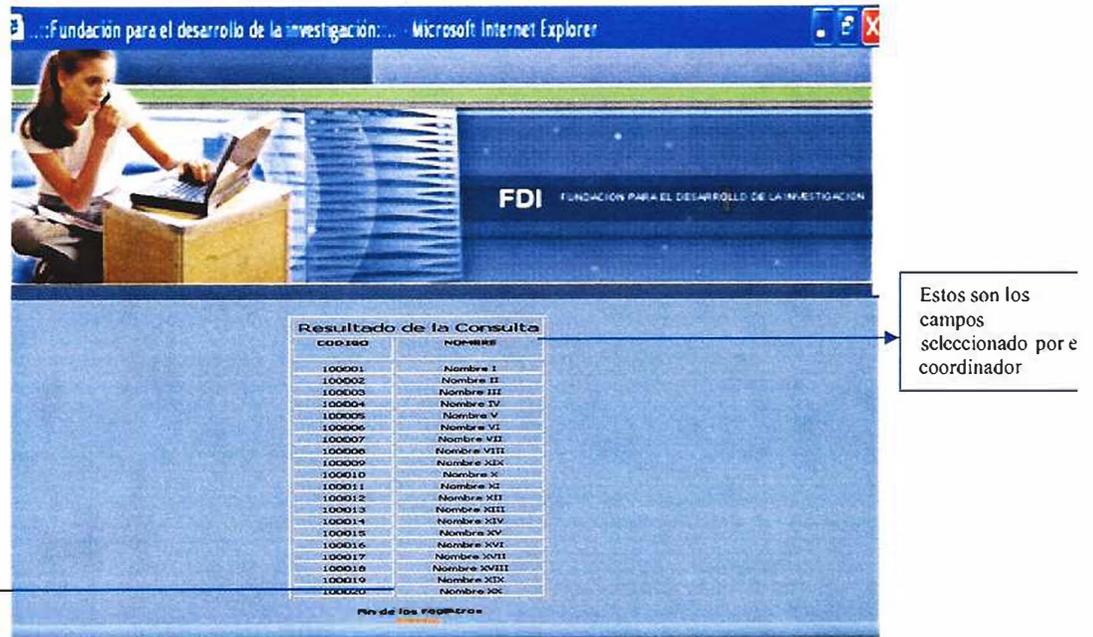
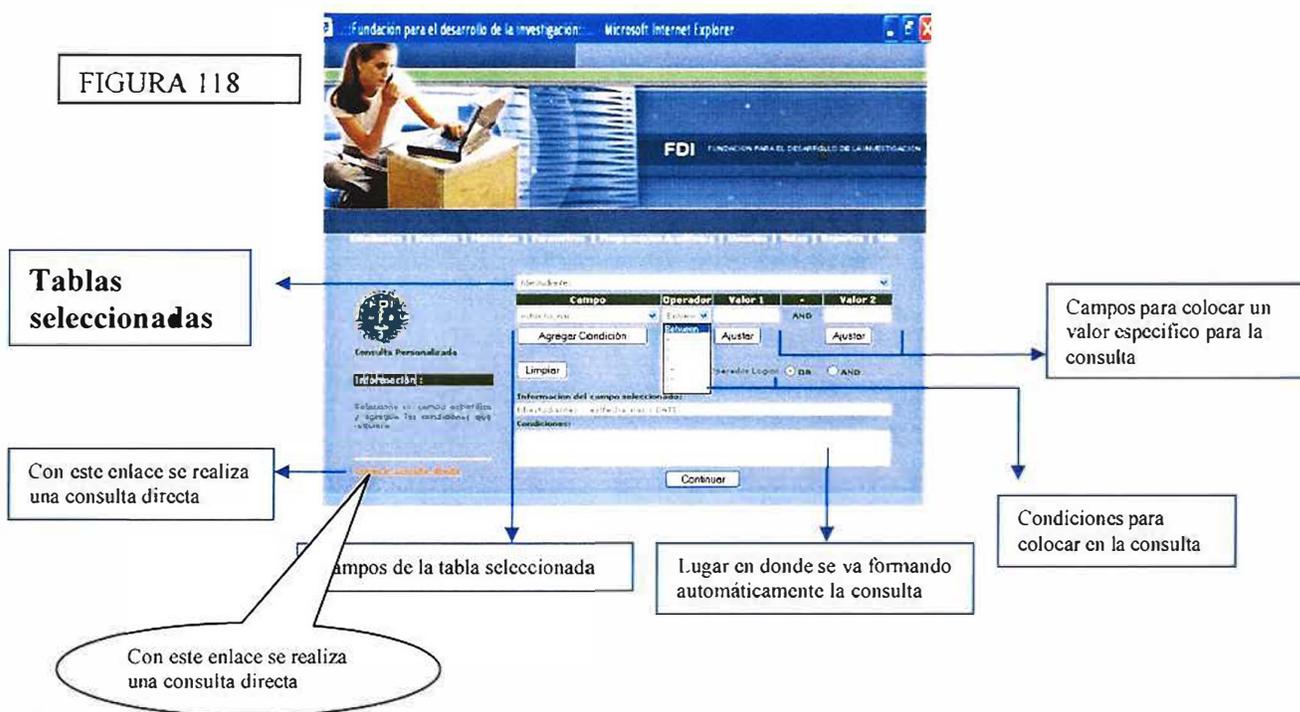


FIGURA 117

- **Agregar:** Con este botón se adicionarán las tablas seleccionadas al espacio en blanco que se encuentra en el lado derecho del primer formulario de Reportes Dinámicos.

- **Quitar:** Este botón eliminará todas aquellas tablas que el usuario haya seleccionado por error.
- **Limpiar:** Cuando el usuario presione este botón dejara el espacio en blanco que se encuentra a la parte derecha del formulario sin ninguna tabla.
- **Enviar:** Cuando el coordinador de click en este botón lo transportará a la siguiente parte de la aplicación en donde seleccionará un campo específico de la tabla escogida y también agregará las condiciones que desee. A continuación se muestra la parte de formulario anteriormente mencionada.



- **Agregar Condición:** Como su mismo nombre lo dice le permitirá al coordinador adicionar las condiciones (Between, =, >, <, >=, <= y <>) que el desee para su consulta.
- **Ajustar:** Con este botón el usuario podrá determinar cualquier fecha que el requiera para una consulta a través de un calendario que se despliega cuando se presiona este botón, hay dos botones **Ajustar** porque si el coordinador quiere saber cuales son los estudiantes que se matricularon entre enero del 2001 hasta marzo del 2006 lo realizará



por medio de estos dos botones en donde seleccionará la fecha de inicio y de finalización.

Este es el calendario que se despliega cuando se presiona el botón **Ajustar**.



FIGURA 119

- **Continuar:** Después de haber seleccionado la tabla y los campos correspondiente se presiona este botón para que el coordinador seleccione los campos que quiere que se visualice en reporte, además le puede colocar un alias al nombre de campo por ejemplo el campo estcodigo de la tabla FDIestudiante se le puede colocar el alias de código y con este nombre saldrá en el reporte.

Esta es la parte de la aplicación en donde se selecciona los campos que van aparecer en el reporte.



Tablas Seleccionadas

FIGURA 120

Consulta directa

Organización de la consulta(cod, nombre, etc)

- **Agregar Campo:** Este botón permitirá agregar todos aquellos campos que se quieran visualizar en el reporte.
- **Limpiar Campo:** Presionando este botón se eliminarán toda la información que se encuentra reflejada en los tres campos ubicados en la parte inferior de la imagen anterior.
- **Finalizar:** Presionando este botón el coordinador observará el reporte , por ejemplo este que se encuentra en la parte inferior.

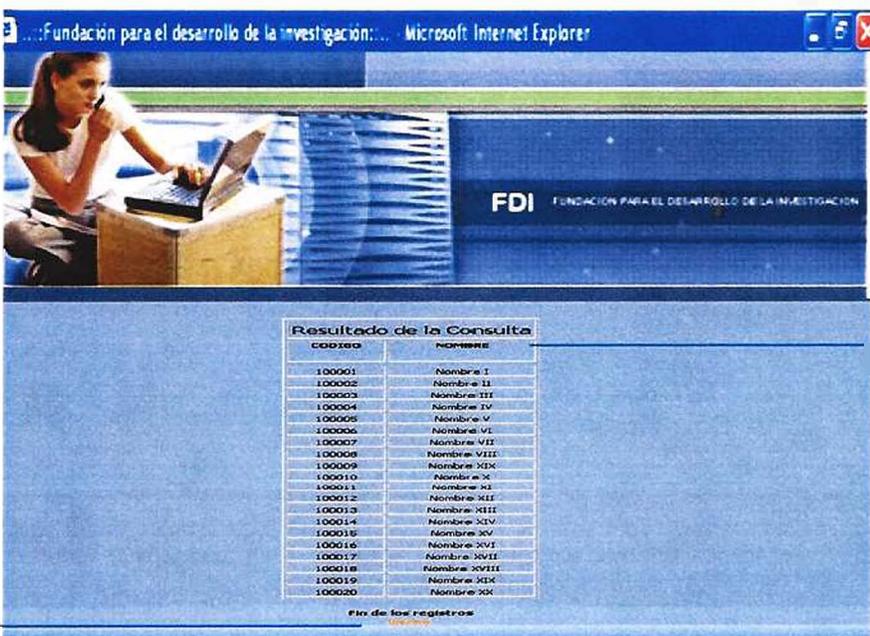


FIGURA 121

Campos seleccionado por el coordinador

Con este link imprime el reporte

| Resultado de la Consulta | |
|--------------------------|--------------|
| CODIGO | NOMBRE |
| 100001 | Nombre I |
| 100002 | Nombre II |
| 100003 | Nombre III |
| 100004 | Nombre IV |
| 100005 | Nombre V |
| 100006 | Nombre VI |
| 100007 | Nombre VII |
| 100008 | Nombre VIII |
| 100009 | Nombre IX |
| 100010 | Nombre X |
| 100011 | Nombre XI |
| 100012 | Nombre XII |
| 100013 | Nombre XIII |
| 100014 | Nombre XIV |
| 100015 | Nombre XV |
| 100016 | Nombre XVI |
| 100017 | Nombre XVII |
| 100018 | Nombre XVIII |
| 100019 | Nombre XIX |
| 100020 | Nombre XX |

Fin de los registros