

**DISEÑO, DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA DE
INFORMACIÓN, PARA EL MANEJO DE LOS PROCESOS ACADÉMICOS
EN LA CORPORACIÓN TÉCNICA DE ESTUDIOS
ESPECIALIZADOS DEL CARIBE (CODETEC)**

**JOSÉ G. DE LA CRUZ
NARCISO CANTILLO
WILLIAM CASTILLO
FREDY RENY**

Trabajo presentado en la Asignatura Investigación Formativa

Al Ingeniero

FREDY BRICEÑO

**BARRANQUILLA
UNIVERSIDAD SIMÓN BOLÍVAR
INGENIERÍA DE SISTEMAS
XI SEMESTRE – JORNADA NOCTURNA
2007**

TABLA DE CONTENIDO

INTRODUCCIÓN	Pág. 5
1. PLANTEAMIENTO DEL PROBLEMA	7
1.1 DESCRIPCIÓN DEL PROBLEMA	7
1.2 FORMULACIÓN DEL PROBLEMA	8
1.3 SISTEMATIZACIÓN DEL PROBLEMA	8
2. OBJETIVOS	9
2.1 OBJETIVO GENERAL	9
2.2 OBJETIVOS ESPECIFICOS	9
3. JUSTIFICACION DEL PROYECTO	10
4. MARCO DE REFERENCIA	12
4.1 ANTECEDENTES	12
4.2 MARCO TEÓRICO	14
4.3 MARCO CONCEPTUAL	30
4.4 MARCO ESPACIAL	30
4.5 MARCO LEGAL	30
5. METODOLOGÍA	33
5.1 TIPO DE ESTUDIO	33
5.2 LÍNEA DE INVESTIGACIÓN	33
6. RECURSOS	34
6.1 RECURSOS HUMANOS	34
6.2 RECURSOS FÍSICOS	35
6.3 RECURSOS TECNOLÓGICOS	35
7. CRONOGRAMA	36
8. INGENIERÍA DE REQUISITOS	37

8.1 DESCRIPCIÓN DEL SISTEMA ACTUAL	37
8.2 DIAGRAMA DE FLUJO DE DATOS DEL SISTEMA ACTUAL	41
8.3 ANÁLISIS DE REQUISITOS	42
8.3.1 LISTADO DE REQUISITOS CODETEC	42
8.3.2 LISTADO DE REQUISITOS ESPECÍFICOS	43
8.3.3 REQUERIMIENTOS FUNCIONALES	43
8.3.4 REQUERIMIENTOS NO FUNCIONALES	44
9. INGENIERÍA DE INFORMACIÓN	45
9.1 MISIÓN	45
9.2 VISIÓN	45
9.3 GENERALIDADES	46
9.4 ORGANIGRAMA	48
10. ANÁLISIS DEL SISTEMA	49
10.1 ESPECIFICACIÓN DE ENTIDADES	67
10.2 DIAGRAMA DE FLUJO DE DATOS DEL PROYECTO	72
10.3 MODELO ENTIDAD – RELACIÓN	83
10.4 MODELO RELACIONAL	84
10.5 DICCIONARIO DE DATOS	85
11. DISEÑO DEL SISTEMA	89
11.1 ESTRUCTURA FUNCIONAL	89
11.2 DESCRIPCIÓN DE LOS COMPONENTES DE LA ESTRUCTURA	90
11.3 DISEÑO DE LAS INTERFACES	92
CONCLUSIÓN	96
BIBLIOGRAFÍA	97

INTRODUCCIÓN

El desarrollo de sistemas de información es una de las tareas a las cuales se dedica la Ingeniería de Software, generalmente se podrían realizar proyectos que satisfagan todas las necesidades del medio; ésto si ejecuta un buen plan de trabajo que lleve a desarrollar e implementar las herramientas necesarias para estar a la altura de las grandes compañías de Software. Es una realidad que el mercado actual es muy exigente, por consiguiente solo los trabajos que marquen diferencia en cuanto a su calidad y rendimiento tendrán éxito en el desarrollo de la economía, por esta razón, es cada vez mas común, ver que las industrias de Software trabajen para realizar proyectos que se acomoden a la medida de cada perfil de negocio, para garantizar un mayor grado de efectividad en el desarrollo de las aplicaciones.

Los requerimientos para la construcción del software están siendo tomados y generalizados desde el lugar donde se originan los procesos, logrando de esta manera una mayor precisión para modelar las situaciones propias de cada negocio.

Hablar de proyectos que tengan que ver con la educación es un tema de gran interés para la comunidad en general, ya que representa fortalecimiento en el desarrollo social y cultural de la sociedad. Brindando al mismo tiempo nuevas alternativas para la educación de nuevas generaciones. Para fortalecerla es necesario que se tengan en cuenta las nuevas tecnologías. Partiendo de esta premisa, se llevará a cabo un proyecto en la Corporación de Estudios Técnicos Especializados del Caribe CODETEC la cuál esta dedicada a la formación de talento humano capacitado para crear, ampliar y difundir conocimientos a la comunidad del caribe en general. CODETEC tiene como meta buscar una excelencia académica en su estudiantado; cuenta con herramientas tecnológicas de alta calidad que facilitan el aprendizaje y aportan a la investigación.

El proyecto consiste en sistematizar los procesos que se llevan a cabo en el departamento académico, ya que existen problemas en el manejo de la información y el registro de los datos. Estos inconvenientes radican en la cantidad de información que se genera a partir de los requerimientos de cada proceso; como son; registro de notas, historia de estudiantes egresados y activos, matriculas, inscripciones, horarios, contenido de los programas entre otros.

Siendo conscientes de esta realidad que los está golpeando fuerte, las directivas de CODETEC decidieron incluir en sus proyectos la utilización de herramientas tecnológicas que permitan la automatización de sus procesos diarios, de esta forma inició la investigación con el cual se pretende dar soluciones al problema propuesto y desarrollar el proyecto de sistematización del departamento académico. El sistema de información se desarrollará a partir de tecnologías de diseño y desarrollo de software, se seguirán patrones que definirán la estructura que tendrá el proyecto. Esta estructura es realmente la radiografía del sistema propuesto, el cual se diseñará y ejecutará con una programación orientada a eventos y con procesos de diseño que permitan la retroalimentación en cada etapa del desarrollo, de manera que habrá una comunicación permanente con los actores de cada proceso. La recolección de la información se realizará a través de entrevistas, encuestas y cuestionarios; métodos que permitirán obtener información de primera mano en lo referente a cada actividad desarrollada,

Al finalizar la etapa de análisis y diseño se desarrollará un prototipo de la aplicación que mostrará como se llevaran a cabo las principales actividades, esto contribuirá al desarrollo del proyecto brindando un enfoque con mayor dirección en el desarrollo de la plataforma real.

1. PLANTEAMIENTO DEL PROBLEMA

1.1 DESCRIPCIÓN DEL PROBLEMA

CODETEC en el desarrollo de sus actividades diarias en el manejo de los procesos académicos, está presentando dificultad a la hora de realizar las actividades que están involucradas con el manejo de los registros que conforman sus archivos de datos, lo que representa los siguientes inconvenientes:

- Redundancia de información.
- Falta de organización en la documentación.
- Retraso en los procesos de matrícula.
- Gasto excesivo de papelería.
- Inconsistencia en horario de los docentes.
- Desactualización de los datos.
- Pérdida de registro en los archivos.
- Atraso en la búsqueda de información.
- Inconsistencia en los datos.

CODETEC ha tenido un crecimiento importante en su población estudiantil y en la cobertura que antes ofrecía, la demanda de estudiantes que se está presentando ha sobrepasado estadísticamente a los años anteriores; esto trae consigo congestión en los procesos por la gran cantidad de registros que ahora se tienen que manejar.

El volumen de información que actualmente se está generando en CODETEC puede afectar seriamente el desarrollo de sus actividades, se podrían presentar

inconsistencias, saturación y duplicación de la información; lo que entorpecería los procesos que diariamente se realizan.

Frente a esta problemática sería conveniente la automatización de todos estos procesos en los que CODETEC presenta inconvenientes, mediante un sistema de información que permita organizar y desarrollar de manera más eficiente la gestión académica que adelanta la institución.

1.2 FORMULACIÓN DEL PROBLEMA

¿Cómo se podrían estructurar y organizar los procesos de gestión académica en la Corporación De Estudios Técnicos Especializados Del Caribe, para lograr la eficiencia en el desarrollo de sus actividades?

1.3 SISTEMATIZACIÓN DEL PROBLEMA

- ¿Qué procesos y actividades se llevan a cabo en la gestión académica en la Corporación De Estudios Técnicos Especializados Del Caribe?
- ¿Cómo es el manejo de la información y que procesos intervienen en la gestión académica en la Corporación De Estudios Técnicos Especializados Del Caribe?
- ¿De que forma se podrían optimizar los procesos de matrícula e inscripción que se llevan a cabo en la Corporación De Estudios Técnicos Especializados Del Caribe?
- ¿Cuáles procedimientos se deberían utilizar para obtener información actualizada en tiempo real?
- ¿Qué procedimientos se deberían implementar para lograr una adecuada gestión en el manejo de las notas?

2. OBJETIVOS

2.1 OBJETIVO GENERAL

Construir un sistema de información que permita optimizar los procesos y actividades de gestión académica que se llevan a cabo en la Corporación De Estudios Técnicos Especializados Del Caribe.

2.2 OBJETIVOS ESPECIFICOS

- Identificar los procesos y actividades de gestión académica que se realizan en la Corporación De Estudios Técnicos Especializados Del Caribe.
- Analizar como es el manejo de la información en los procesos académicos que se llevan a cabo en la Corporación De Estudios Técnicos Especializados Del Caribe.
- Diseñar las estructuras y procedimientos necesarios que optimicen los procesos de matrícula e inscripción que se llevan a cabo en la Corporación De Estudios Técnicos Especializados Del Caribe.
- Especificar procedimientos que permitan disponibilidad en la información actualizada.
- Diseñar los procedimientos que permitan una adecuada gestión académica en el manejo de las notas.

3. JUSTIFICACION DEL PROYECTO

Para la Corporación De Estudios Técnicos Especializados Del Caribe "CODETEC", es de gran valor la ejecución del proyecto, porque optimizara sus actividades diarias y realizará de forma más eficiente todos los Procesos Académicos que se manejan en la institución, entre los que se encuentran: Registro de matrícula, registro de notas, consulta de notas, pagos de módulos, certificados de notas, registro de asignaturas, registro de docentes, registro de programas, asignación de horarios entre otros.

Sin duda la realización del proyecto genera gran valor tecnológico para la Corporación ya que los procesos serán automatizados mediante herramientas que permitirán mas rapidez, mejor organización y mayor consistencia en la gestión de la información, lo que conlleva a un menor tiempo de respuesta en los requerimientos que ayudan a la dirección.

La utilización de esta herramienta permitirá mayor proyección para la institución ya que se tendrá acceso a información precisa y mejor organizada, lo que facilitara las decisiones que se tengan que tomar, la propuesta consiste en un software dinámico que genere información mediante una estructura de datos que permita obtener respuestas rápidas a las búsquedas, en todo sentido la ejecución del proyecto traerá consigo desarrollo en la estructura organizacional de CODETEC, lo que permite crecimiento inmediato y posicionamiento ante la comunidad educativa, de esta forma podrán ampliar su cobertura, y lograr llevar mas conocimiento a la población en general,

La ejecución del proyecto traerá progreso y generará un ambiente de seguridad en el tratamiento de la información; tendrá la responsabilidad de manejar el activo más valioso de la Corporación y sin duda será el centro de operaciones de todas las actividades, lo que representa un alto grado de responsabilidad en su construcción.

Al estar al frente del proyecto es fácil darse cuenta que hay un mercado que está creciendo y exige mayor organización en su gestión, lo que permite establecer parámetros y fijar objetivos de dirección, que conlleven a explorar esas oportunidades que están esperando que alguien las aproveche; ésto se convierte en motivación para realizar un trabajo que pueda ser accesible para sectores que no son ajenos a una estructura similar.

4. MARCO DE REFERENCIA

4.1 ANTECEDENTES

1 - NOMBRE DEL PROYECTO: DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE INFORMACIÓN DE INSCRIPCIÓN, MATRÍCULA Y NOTAS PARA LA INSTITUCIÓN EDUCATIVA AMPARO DE DIOS AJAIB SING DE BARRANQUILLA.

INTEGRANTES:

Acosta Torres Saleh Alfonso.

Almanza Orozco Leonid.

Escorcía Maya Ramsés Raúl.

Ojeda López Deivis Howar.

OBJETIVOS: Desarrollar un sistema de información que facilite el proceso de inscripción, matrícula y notas en la institución educativa con el fin de agilizar y optimizar sus procesos haciendo uso de herramientas GNU.

2 - NOMBRE DEL PROYECTO: DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE INFORMACIÓN PARA EL MANEJO Y CONTROL DE LOS PROCESOS ACADÉMICOS (INSCRIPCIÓN, MATRÍCULA, Y REGISTRO DE NOTAS) DEL COLEGIO MARCO FIDEL SUÁREZ DE BARRANQUILLA.

INTEGRANTES:

Neisa Pedroza Deisy Esperanza.

Rada Castro Lorena Patricia.

Rincón Salas Nini Johana.

Torres Camargo Martha Cecilia.

OBJETIVOS: Desarrollar un sistema de información para el manejo y control de los procesos académicos (inscripción, matrícula, documentos y registro de notas) del colegio Marco Fidel Suárez.

3 - NOMBRE DEL PROYECTO: DISEÑO Y DESARROLLO DE UN SOFTWARE QUE PERMITA AGILIZAR EL PROCESO DE MATRÍCULA Y NOTAS DEL INSTITUTO TÉCNICO MICROEMPRESARIAL DE SOLEDAD.

INTEGRANTES:

Acevedo Vivier.

Barrios Zamora Guiseth Yasmín.

Maury Gustavo.

OBJETIVOS: Desarrollar e implementar un software seguro que permita agilizar el proceso de matrícula y notas de los alumnos del Instituto Microempresarial de Soledad.

4 - NOMBRE DEL PROYECTO: SISTEMA DE INFORMACIÓN ACADÉMICO-ADMINISTRATIVO PARA LA GESTIÓN DE DATOS EN LOS DEPARTAMENTOS DE MATRÍCULA Y PROCESOS ACADÉMICOS EN LA INSTITUCIÓN EDUCATIVA DISTRITAL SAN SALVADOR DE BARRANQUILLA.

INTEGRANTES:

Albor Arrieta Elías.

Barros Ortega Luis.

Insignares Rodríguez Marlon.

Padilla Romero Jair.

OBJETIVOS: Diseñar y desarrollar un sistema de información que administre y gestione la información en los departamentos de matrícula y procesos académicos en el Colegio Distrital San Salvador.

4.2 MARCO TEÓRICO

En los inicios de la industria del software se manejaban hipótesis que hacían creer que los sistemas de información estaban muy lejos de convertirse en parte esencial de las economías en el mundo, sólo bastaron dos décadas para mostrar que el software es el “motor del futuro”, sin duda está presente en cada situación que representa una actividad comercial, gubernamental o educativa. Diferentes autores hablan del tema refiriéndose a la plataforma como “la cuna de la economía mundial¹”. Las decisiones más importantes son tomadas a partir de información proporcionada por sistemas de computadoras que tiene la gran responsabilidad de procesar millones de datos para generarla.

Sin duda la importancia que ha tenido el desarrollo de hardware ha influido de manera positiva a potencializar las aplicaciones que se producen, aprovechando todo lo que tiene que ver con utilización de recursos, lo que representa avances tecnológicos que todos días salen al mercado. Pero detrás de todos estos avances existen situaciones que han marcado el camino del éxito en esta industria, al inicio no había ningún tipo de estructura para construir sistemas de información, el diseño era desordenado y sólo algunas aplicaciones contaban con suerte y podían ser terminadas, esto generaba grandes volumen de código fuente, errores lógicos en la estructura de la sentencias y costes altos de mantenimiento que se veían reflejados en el poco acceso al mercado, ser programador definitivamente representaba contar con un arte mágico o poco común. Al ver los pobres resultados los desarrolladores se vieron en la necesidad de

¹ Ingeniería del software shock, Pressman y Herron, 1997.

cambiar estos esquemas y construir un camino bajo mejores expectativas, fue de esta forma como se empezaron a crear modelos de desarrollo que contaban con etapas de diseño y de análisis para construir aplicaciones, lo que permitió reducir de manera considerable los márgenes de error y las líneas de código fuente, de esta forma se hacia menos complejo el mantenimiento y la corrección de las sentencias. Se empezó hablar entonces de **Ingeniería del software**, y era definida como “el establecimiento y uso de principios robustos a fin de obtener económicamente software que sea fiable y que funcione eficientemente sobre maquinas reales² “. Para sustentar esta definición se tomó como referencia el libro de Pressman Ingeniería de Software donde se argumenta lo siguiente: “Casi todos los lectores tendrían la tentación de seguir esta definición, no dice mucho sobre los aspectos técnicos de la calidad del software; no se enfrenta directamente con la necesidad de satisfacción del cliente o de la entrega oportuna del producto, omite la mención de la importancia de mediciones y métricas; tampoco expresa la importancia de un proceso avanzado. Y sin embargo la definición de Bauer nos proporciona un línea base ¿Cuáles son los principios robustos de la ingeniería aplicables al software de computadoras?, ¿Cómo se construye el software económicamente para que sea fiable?, ¿Qué se necesita para crear programas de computadoras que funcionen eficientemente no en una maquina si no en diferentes maquinas reales? Estas son cuestiones que siguen siendo un reto para los ingenieros del software³”.

Estos esquemas se ubican en una realidad que compromete a seguir estas corrientes, que de alguna u otra forma fueron las que marcaron los caminos que hoy por hoy se recorren en esta industria, o tras de las referencias que hay que tener en cuenta es la de la **IEEE** que deja claro en su definición lo siguiente, “la ingeniería del software es la

² Ingeniería del software shock, Pressman y Herron, 1997.

³ Ingeniería del Software shock, Pressman y Heron, 1997.

aplicación de un enfoque sistemático, disciplinado y cuantificable hacia el desarrollo, operación y mantenimiento del software⁴.

Sin duda todas estas definiciones dejan algo claro en su contenido teórico, la planificación es el aspecto más importante en la construcción de sistemas de información, de ella depende el buen desarrollo de las actividades programadas y el cumplimiento de los objetivos propuestos, para tener una buena planificación hay que tener en cuenta los modelos existentes de desarrollo y trabajar con el que mejor se acomode a los requisitos del sistema, entre estos modelos se encuentran: “**el lineal secuencial**: que representa el ciclo de vida de la naturaleza, las etapas son análisis, diseño, código, prueba.

La construcción de prototipos: que presenta una muestra rápida de los posibles requerimientos de un sistema para luego convertirlo en aplicaciones mas complejas mediante etapas de diseño y desarrollo, el modelo

DRA: consiste en sistematizar el problema y dividirlo en equipos de trabajo que luego fusionan las aplicaciones y conforman el producto final, utilizan el modelo lineal secuencial para el desarrollo del software.

Modelo Incremental: combina elementos del lineal secuencial aplicados rápidamente, con la filosofía interactiva de construcción de prototipos

Modelo Estructurado: implica modelar el ambiente del usuario con diagramas de flujo de datos, diagramas entidad relación, diagramas de transición de estado y modelos de comportamiento, los cuales se combinan para formar el modelo esencial que representa una descripción formal de lo que el nuevo modelo sistema debe hacer, independientemente de la naturaleza de la tecnología que se utilice para cubrir los

⁴ Ingeniería del Software shock, Pressman y Heron, 1997.

requerimientos. También se debe preparar un conjunto de presupuestos y cálculos de costo beneficio, precisos y detallados.

Modelo Basado En Componentes: puede verse como una extensión natural de la programación orientada a objetos dentro del ámbito de los sistemas abiertos y distribuidos. Este paradigma se basa en el uso de los componentes software como entidades básicas del modelo. Una de las principales ventajas es la reutilización de los mismos. De esta forma los componentes se diseñan y desarrollan con el objetivo de poder ser reutilizados en otras aplicaciones, los aspectos mas importantes de la reutilización de componentes se pueden ver con la: reducción en el ciclo de vida del proyecto en un 70%, reducción en el costo del proyecto en un 84% y aumento de la productividad en un 65%⁵.

Lo anteriormente estudiado se refiere al diseño de las aplicaciones que gestionaran los datos, pero para que haya una verdadera armonía en el funcionamiento es importante que definamos cual es la estructura que deberían tener esos datos, "las bases de datos juegan un papel definitivo a esta altura de la aplicación, ellas determinan como debe ser el acceso a la información y como se gestionaran los procesos que generen los objetivos propuestos en las sentencias de código⁶". En el mercado existen diversidad de estas herramientas tecnológicas, algunos autores presentan productos como **MYSQL** y la definen como un "Sistema gestor de base de datos relacional cliente servidor de coste mínimo que incluye un servidor SQL diseñado para el trabajo tanto en sistemas operativos **WINDOWS** como en los sistemas **UNIX / LINUX**, implementa funcionalidades Web permitiendo un acceso seguro y sencillo a los datos a través de Internet, incluye capacidades de análisis integrada , servicios de administración y duplicación de datos. MYSQL representa un producto de libre distribución ideal para el desarrollo de proyectos, cuenta con herramientas potentes para el manejo de los datos,

⁵ Software shock, pressman y Heron, 1997.

⁶ Fundamento de bases de datos, Abraham silberschatz, Henry F. Korth, S. Sudarshan. Mc Graw – Hill, 1998.

tiene la capacidad en sus ultimas versiones de almacenar mas de seis millones de registros por cada tabla, lo que le da importancia de ser una herramienta tecnológica y competitiva⁷ “

Cuando se hace la planificación de La base de datos, “es importante tener en cuenta el uso de herramientas **CASE**, estas pueden mejorar la productividad en el desarrollo de una aplicación de base de datos, y por productividad se entiende tanto la eficiencia en el desarrollo, como la efectividad en el sistema desarrollado, la eficiencia se refiere al coste, tanto en tiempo como en dinero, la efectividad se refiere al grado del sistema de satisfacer las necesidades de los usuarios. Para obtener una buena productividad, subir el nivel de efectividad puede ser mas importante que amentar la eficiencia⁸”.

Cuando se tiene una estructura de los datos definida y se realiza el análisis de todas los procesos, es muy importante tener en cuenta el lenguaje de programación que se utilizará para desarrollar las sentencias de código fuente que administraran los datos, existe una gran variedad de estos lenguajes de programación, lo mas común es que el `programador se especialice en uno de ellos y que aproveche todos los recursos que este tenga, algunos autores presentan productos como **VISUAL BASIC** y lo definen como “ un lenguaje ideal para principiantes por su facilidad de realizar pequeñas aplicaciones , y al mismo tiempo para los profesionales porque puede acometer

proyectos mas complejos con una alta garantía de éxito y a la vez facilidad de uso, la palabra visual hace referencia al método que utiliza para crear interfaz grafica de usuario, en lugar de crear aplicaciones mediante laboriosas líneas de código, se recurre a la técnica de agregación/eliminación de objetos prefabricados dentro de la pantalla, la palabra Basic hace referencia al mítico lenguaje creado en darmouth college de estados unidos y que han utilizado un mayor numero de programadores que ningún otro en la

⁷ MYSQL Para Windows y Linux. Cesar Pérez Alfa y Omega, 2005.

⁸ Ingeniería del software, pressman y Roger, 4ª edición, 1998.

historia de la informática. Las principales características de **VISUAL BASIC** son: posibilidad de acceso a los datos para creación de base datos, aplicaciones cliente y componentes del servidor que son escalables para los formatos de base de datos mas conocidos, entre ellos **SQL SERVER**. Incorporación de tecnología **Activex**, se puede extender las funciones asociadas a visual Basic instalando los controles activex asociados a Visual Basic, o los controles que cree por si mismo o que hayan sido desarrollados por otros fabricantes de programas, los controles activex que se instalen utilizan la tecnología de 32 bit, Proporciona técnicas de programación necesarias para trabajar con servidores web, protocolos y aplicaciones en dynamic HTML en Internet. Permite la creación de archivos **“.EXE “**mediante una maquina virtual de Visual Basic que permite la distribución con toda libertad sin necesidad de programar en Visual Basic⁹.

Sin duda las herramientas que se utilizan para desarrollar el proyecto son de gran importancia a la hora de medir los resultados de las aplicaciones, pero lo mas importante es aprovechar todos los recursos que ellas proporcionan, uno de esos recursos que generan gran valor es el tema de la **seguridad del software**, algunos autores definen esta seguridad como “revisión de los procedimientos de acceso al software y su documentación, identificar los procedimientos de seguridad para restringir los perfiles de cada usuario de acuerdo a su nivel de responsabilidades en la empresa. El tema de la seguridad es muy importante en el desarrollo de este tipo de proyectos representa la confianza de la reducción de riesgos de operación en el desarrollo de las operaciones otro autor plantea, los controles que se apliquen definirán el grado de permanencia de una aplicación en una organización, ya que están ligados con la parte mas sensible de la empresa, la perdida de dinero¹⁰”.

⁹ MICROSOFT Visual Basic 6.0. Luis Joyanes Aguilar, Antonio muños clemente McGraw – Hill, 1999

¹⁰ Ingenieria del software. pressman roger 4ª edición, 1998.

Sin duda que tener en cuenta todas estas consideraciones representa en gran parte el éxito de un proyecto, por esta razón también es importante tener en cuenta el “**mantenimiento del software**, que consiste en actualización del producto, en las mejoras de las versiones en curso, actualización de las librerías existentes así como las nuevas que pudieran aparecer, actualización de las nuevas versiones del software, aplicar estas ideas representa un gran valor para el desarrollo de las actividades en el curso normal de eventos del software¹¹”.

Después de haber considerado todas estas teorías de los diferentes autores y que el proyecto haya sido terminado, es importante se logre medir el resultado del trabajo hecho, para esto existen algunas **Estrategias de pruebas del software** “la prueba del software es un elemento de un tema más amplio que a menudo se le conoce como **VERIFICACION Y VALIDACION**, la verificación se refiere al conjunto de actividades que aseguran que el software implementa correctamente una función específica, la validación se refiere a un conjunto de diferentes actividades que aseguran que el software construido se ajusta a los requisitos del cliente, se puede decir que esto se responde con los siguientes interrogantes ¿se está construyendo el producto correctamente?, ¿se está construyendo el producto correcto?. La validación del software se consigue mediante una serie de pruebas de caja negra que demuestran la conformidad de los requisitos. Un plan de prueba traza las clases de pruebas que se han de llevar a cabo, y un procedimiento de prueba define los procesos específicos en un intento por descubrir errores de acuerdo con los requisitos, como resultado de la verificación y de la validación se emplea la **Depuración** considerada como el proceso que provoca la eliminación de un error, ocurre como consecuencia de una prueba efectiva y debe ser un proceso ordenado aunque tiene mucho de arte. El proceso de depuración comienza con la ejecución de un caso de prueba, se evalúan los resultados y aparece una falta de correspondencia entre los esperados y los encontrados realmente en muchos casos, los datos que no concuerdan son síntomas de una causa

¹¹ Auditoria y control de procesamiento de datos, Richard W. Lott, norma 1991

subyacente que todavía permanece oculta, el proceso de depuración intenta hacer corresponder el síntoma con una causa, llevando a sí a la corrección del error¹²”.

De esta forma se observa la suma importancia de tener referencias de cómo son los procedimientos en la construcción de sistemas de información, esto permite crear productos de calidad que puedan ser competitivos en cuanto a rendimiento y funcionamiento para el desarrollo de las aplicaciones, claro que para desarrollar sistemas de información también es importante tener referencias de proyectos creados anteriormente que sigan la misma línea de investigación, tener estas reseñas permitirá una mayor visión de los requerimientos y un mayor aprovechamiento de los recursos con que se cuenta.

PRINCIPIOS Y OBJETIVOS GENERALES DE LA EDUCACIÓN

Desde la perspectiva de nuestro devenir socio - político, el año 1991 marca para la Nación Colombiana el hito más significativo del presente siglo, con la creación y construcción en forma participativa de nuestra nueva carta constitucional, que nos define como: un Estado Social de derecho, organizado en forma de república unitaria, descentralizada, con autonomía en sus entidades territoriales, democrática, participativa y pluralista fundada en el respeto de la dignidad humana, en el trabajo y la solidaridad de las personas que la integran y en la prevalencia del interés general.

En ella, como fruto de la confluencia de múltiples y diversas visiones del futuro, ideologías, creencias, concepciones, sentimientos, intereses, aspiraciones y culturas hemos logrado concertar los derroteros que nos vienen, de tiempo atrás, orientando y guiando hacia la conquista de una Democracia Participativa, fundamentada en la promoción y desarrollo del bienestar humano y planetario.

¹² Auditoria y control de procesamiento de datos, Richard w. lott, norma 1991

Es así como la carta define a la educación como un derecho de la persona y un servicio público que tiene una función social, con ella se busca el acceso al conocimiento, a la ciencia, a la técnica, y a los demás bienes y valores de la cultura.¹³

La educación formará al colombiano en el respeto a los derechos humanos, a la paz y a la democracia; y en la práctica del trabajo y la recreación, para el mejoramiento cultural, científico, tecnológico y para la protección del ambiente.

El Estado, la sociedad y la familia son responsables de la educación, que será obligatoria entre los cinco y los quince años de edad y que comprenderá como mínimo, un año de preescolar y nueve de educación básica mínima. La educación será gratuita en las instituciones del Estado, sin perjuicio del cobro de derechos académicos a quienes puedan sufragarlos.

Corresponde al Estado regular y ejercer la suprema vigilancia de la educación con el fin de velar por su calidad, por el cumplimiento de sus fines y por la mejor formación moral, intelectual y física de los educandos; garantizar el adecuado cubrimiento del servicio y asegurar a los menores las condiciones necesarias para su acceso y permanencia en el sistema educativo.

Los padres de familia tendrán derecho a escoger el tipo de educación para sus hijos menores. En los establecimientos del Estado ninguna persona podrá ser obligada a recibir educación religiosa. Los integrantes de los grupos étnicos tendrán derecho a una formación que respete y desarrolle su identidad cultural. La erradicación del analfabetismo y la educación de personas con limitaciones físicas o mentales, o con capacidades excepcionales, son obligaciones especiales del Estado.¹⁴

¹³ <http://www.oei.org.co/quipu/colombia/resumen.htm#4>

¹⁴ <http://www.oei.org.co/quipu/colombia/resumen.htm#4>

La constitución establece que el Estado garantiza las libertades de enseñanza, aprendizaje, investigación y cátedra y obliga al estudio de la constitución nacional y de la instrucción cívica en todas las instituciones de educación tanto oficiales como privadas. Se garantiza la autonomía universitaria. Las universidades podrán darse sus directivas y regirse por sus propios estatutos, de acuerdo con la ley. La ley establecerá un régimen especial para las universidades del Estado.

El Estado fortalecerá la investigación científica en las universidades oficiales y privadas y ofrecerá las condiciones especiales para su desarrollo.

El Estado facilitará mecanismos financieros que hagan posible el acceso de todas las personas aptas a la educación superior.¹⁵

LEYES Y OTRAS NORMAS FUNDAMENTALES RELATIVAS A LA EDUCACIÓN

Además de la Constitución Nacional, las principales disposiciones legales relativas a la educación en el país son las siguientes:

La Ley No. 43 del 11 de diciembre de 1975, por la cual se nacionalizó la Educación Primaria y secundaria que oficialmente venían prestando los Departamentos, el Distrito Especial de Bogotá y los municipios.

El Decreto 088 de 1976, mediante el cual se reestructura el sistema educativo y se reorganiza el Ministerio de Educación Nacional.

La Ley 30 de 1992, por la cual se organiza el servicio público de la educación superior

¹⁵ <http://www.oei.org.co/quipu/colombia/resumen.htm#4>

La Ley 60 de 1993 de Distribución y Competencias y Recursos entre los diferentes ámbitos territoriales del país

La Ley 115 del 8 de febrero de 1994, por la cual se expide la Ley General de Educación

El Decreto 1860 de agosto de 1994, se reglamenta parcialmente la Ley 115, en los aspectos pedagógicos y organizativos generales.

El Plan decenal, establecido por la Ley General de Educación (artículo 72) y reglamentado por el Decreto 1719 de 1995, tiene por objeto fundamental elevar la educación a la categoría de política de Estado trascendiendo así los márgenes de un período de gobierno. El Plan es una guía de acción que pretende definir las metas para la educación colombiana durante los próximos 10 años y provocar una movilización no sólo institucional, sino social y continua en torno a la educación.¹⁶

El Decreto 2886 de diciembre 29 de 1994, por el cual se reglamenta los procedimientos y demás formalidades necesarias que deben cumplir las entidades territoriales para obtener la certificación del cumplimiento de los requisitos que les permita asumir la administración de los recursos del situado fiscal y prestación del servicio educativo.

El Decreto 804 del 18 de mayo de 1995, el cual reglamenta la atención educativa para grupos étnicos.

La Ley 188 de 1995, define el Plan Plurianual de Inversiones, que consagra una importante estabilidad de los recursos financieros para el sector educativo.

¹⁶ <http://www.oei.org.co/quipu/colombia/resumen.htm#4>

La resolución N° 2343 de junio 5 de 1996, por la cual se adopta un diseño de lineamientos generales de los procesos curriculares del servicio público educativo, y se establecen los indicadores de logro curriculares para la educación formal¹⁷.

El Decreto 2082 del 18 de noviembre de 1996, reglamenta la atención educativa para personas con limitaciones o con capacidades o talentos excepcionales. Establece los principios que se fundamenta, las orientaciones curriculares especiales para ofrecer dicha atención; señala reglas generales para la organización del servicio educativo que permita la atención; indica orientaciones para la formación de docentes y formula criterios para el apoyo financiero de los programas de atención educativa.

La Resolución 2707 del 26 de junio de 1996, mediante la cual se definen los criterios generales par la evaluación institucional y de docentes estatales ligada al otorgamiento de incentivos de calidad.

El Decreto 2247 del 11 de septiembre de 1997, por el cual se establecen normas relativas a la prestación del servicio educativo del nivel de preescolar.¹⁸

ADMINISTRACIÓN Y GESTIÓN DEL SISTEMA EDUCATIVO

En los últimos años se ha dado un fuerte impulso a la promulgación de normas que facilitan la modernización y el desarrollo de la educación, entre las cuales es importante destacar: a) la Ley de Distribución de Competencias y Recursos entre los diferentes ámbitos territoriales del país, mediante la cual el Estado define y asigna funciones en los tres niveles del gobierno (Nacional, Regional y Municipal) al tiempo que garantiza un

¹⁷ <http://www.oei.org.co/quipu/colombia/resumen.htm#4>

¹⁸ <http://www.oei.org.co/quipu/colombia/resumen.htm#4>

flujo adecuado de recursos a las entidades territoriales (situado fiscal y participaciones) para mantener el equilibrio entre estos y sus competencias. Esta Ley también aborda aspectos relativos a la descentralización, administración y prestación de los servicios educativos, la planeación, el manejo de la información y el desarrollo institucional; b) la Ley General de Educación a través de la cual se reafirma el proceso educativo como derecho fundamental; c) La Ley 188 que consagra una importante estabilidad de recursos financieros para el sector educativo; d) La reorganización del Ministerio de Educación Nacional como ente central, congruente con todo lo anterior.

Con el fin de viabilizar el desarrollo institucional el Ministerio de Educación promueve acciones orientadas al fortalecimiento de la capacidad institucional del sistema educativo tanto a nivel nacional, Ministerio de Educación como en el ámbito regional, Departamentos y Distritos, local, Municipios e Instituciones.

En relación con el ámbito regional y local, además de un programa permanente de asistencia técnica, se dinamiza un Plan de Descentralización como herramientas que incluye los procesos que deben seguir los departamentos y distritos para asumir las competencias que tienen asignadas por Ley para descentralizar responsabilidades y recursos a los municipios.

El énfasis mayor de la descentralización del sector educativo se centra en fortalecer el proceso de Planeación dirigido a la formulación de políticas, el establecimiento de metas y la elaboración de planes.¹⁹

Es así como en el nivel nacional se elabora el Plan Nacional de Desarrollo Educativo, sustentado en el Plan de Gobierno, pero dentro de los lineamientos del Plan decenal.

¹⁹ <http://www.oei.org.co/quipu/colombia/resumen.htm#4>

A nivel regional se elabora el Plan de Desarrollo para la prestación del servicio Educativo, se hace uno por año.

A nivel local, el municipio debe elaborar el Plan de desarrollo educativo; y por último a nivel institucional, deben diseñar y poner en funcionamiento el Proyecto Educativo Institucional (PEI).

Además, los departamentos y municipios deben presentar proyectos que sustenten las inversiones solicitadas.

Con el fin de apoyar a las regiones, localidades y establecimientos educativos, el Ministerio de Educación presta permanentemente una asesoría y diseña y da a conocer guías, manuales y metodologías.²⁰

La Nación y las entidades territoriales ejercen la inspección y vigilancia.

Las funciones en cuanto al servicio público estatal, están definidas por la Ley para cada nivel así:

1. Ministerio de Educación Nacional, tiene las siguientes funciones:

De política y planeación, en cuanto a su formulación, diseño, establecimiento de metas, elaboración de programas, diseño de lineamientos generales de los procesos curriculares, ajuste del situado fiscal, establecimiento de indicadores de logro curriculares, promoción y estimulación la investigación educativa, evaluación y control de los resultados de los planes y programas, elaboración y mantenimiento del registro único de docentes estatales, y proponer programas de inversión para el sector.

²⁰ <http://www.oei.org.co/quipu/colombia/resumen.htm#4>

De inspección y vigilancia, en cuanto a velar en el cumplimiento de la ley y de los reglamentos de la educación, asesorar a las entidades regionales (departamentos, distritos y municipios), en el desarrollo de los procesos pedagógicos, evaluar la prestación del servicio público educativo, establecer los criterios para la evaluación del rendimiento escolar y cumplir y hacer cumplir el Escalafón Nacional docente.

De administración, en relación con la coordinación, a través de las Secretarías de Educación, de los planes y programas educativos; establecer el sistema descentralizado de información y coordinar las acciones educativas del Estado.

Normativas, en relación con definir y fijar criterios para los diferentes aspectos que requiera el sector.

2. Las Asambleas departamentales y los Consejos Distritales y municipales, regulan la educación dentro de su jurisdicción. Las Secretarías de Educación en coordinación con las autoridades nacionales, ejercen las siguientes funciones:

Velar por la calidad y cobertura de la educación

Establecer políticas, planes y programas

Organizar el servicio educativo estatal

Fomentar la investigación, innovación y desarrollo de currículos, métodos y medios pedagógicos.

Diseñar y poner en marcha los programas necesarios

Prestar asistencia técnica a los municipios

Evaluar el servicio educativo

3. Las Secretarías de Educación municipales deben administrar la educación, haciendo énfasis en la organización, ejecución, vigilancia y evaluación del servicio educativo.

4. Los establecimientos educativos tienen que elaborar e implementar el Proyecto Educativo Institucional - PEI -, a través del cual se busca: un plantel organizado, con una misión claramente definida, con pedagogías activas y programas curriculares acordes con las necesidades de formación de los alumnos, con el fin de hacer todo lo posible para que los alumnos obtengan buenos logros y mejorar los niveles de eficiencia en la utilización de los recursos.²¹

Educación no formal

Esta educación se define como aquella que se ofrece con el objeto de complementar, actualizar, suplir conocimientos y formar, en aspectos académicos o laborales sin sujeción al sistema de niveles y grados.

Se rige por los principios y fines generales de la educación. La oferta se hace a través de instituciones que ofrezcan programas de educación laboral en artes y oficios, de formación académica y materias conducentes a la validación de niveles y grados propios de la educación formal.

El encargado de aprobar programas de capacitación y asesoría a microempresas es el Plan Nacional para el desarrollo de las Microempresas, al igual que los programas de apoyo microempresarial, y las instituciones aprobadas para adelantar estos programas tienen carácter de Instituciones de Educación no formal.²²

²¹ <http://www.oei.org.co/quipu/colombia/resumen.htm#4>

²² <http://www.oei.org.co/quipu/colombia/resumen.htm#4>

4.3 MARCO CONCEPTUAL

CANDIDATOS: Son las personas que se encuentran en el proceso de inscripción y que están interesados en los programas que ofrece la Corporación Técnica De Estudios Especializados del Caribe.

PROGRAMAS: Es el conjunto de cursos y otras actividades armónicamente integrado encaminado a dar formación en un determinado campo del conocimiento, en una modalidad académica y bajo una metodología y una jornada específica.

PENSUM: Corresponde al detalle del total de materias a dictarse por programas y módulos establecidos con anterioridad por los coordinadores de cada área.

PERFIL: Define los tipos de usuario que pueden tener acceso al sistema, mediante restricciones que son controladas a través de los permisos otorgados.

PERMISOS: Se refiere a los accesos que cada usuario puede tener a los recursos del sistema dependiendo de su perfil.

4.4 MARCO ESPACIAL

CODETEC se encuentra ubicado en la Cra. 44 No. 79 – 227 Barranquilla – Atlántico.

4.5 MARCO LEGAL

Los derechos de autor están contemplados en Colombia, comenzando por la propia Constitución Nacional. El artículo 61, dedicado a la protección de la propiedad intelectual expresa: “el estado protegerá la propiedad intelectual por el tiempo y mediante las formalidades que establezca la ley”.

La ley 23 de 1982 sobre derechos de autor considera por vez primera al software (“soporte lógico”) como una creación propia del dominio literario²³. Posteriormente, mediante el Decreto Presidencial número 1360 del 23 de junio de 1989, se reglamenta la inscripción del soporte lógico (software), en el Registro Nacional de Derecho de Autor. En este decreto queda escrito en forma explícita: El soporte lógico (software) comprende uno o varios de los siguientes elementos: el programa de computador, la descripción de programa y el material auxiliar.²⁴

En 1993 mediante la ley 44 se modifica el Acto de Derechos de Autor de 1982; adicionalmente, el respaldo legal a los autores de software fue ratificado y detallado en 1993, por la comisión del Acuerdo de Cartagena mediante la Decisión 351, concerniente a régimen común sobre derechos de autor y derechos conexos.²⁵

La legislación Colombiana en forma explícita establece sanciones de “prisión de dos (2) a cinco (5) años y multa de cinco (5) a veinte (20) salarios mínimos mensuales” para quien “reproduzca fonogramas, videogramas, soporte lógico (software) u obras cinematográficas sin autorización previa y expresa del titular, o transporte, almacene, conserve, distribuya, importe, venda, ofrezca, adquiera para la venta o distribución o suministre a cualquier título dichas reproducciones”.

Ley General De Educación

²³ COLOMBIA – CONGRESO DE LA REPUBLICA. Ley 23 de 1982 En página Web <http://www.indsoft.org/antipirateria/DECRETO-1360-1989.html> Versión a 31 de marzo de 1999.

²⁴ COLOMBIA – PRESIDENCIA DE LA REPUBLICA. Decreto 1360 de 1989 En página Web <http://www.indsoft.org/antipirateria/DECRETO-1360-1989.html> Versión a 31 de marzo de 1999.

²⁵ COMISIÓN DEL ACUERDO DE CARTAGENA. Decisión 351, 1993. En página Web <http://www.indsoft.org/antipirateria/Acuerdo.html> Versión a 31 de marzo de 1999.

Num. 2 Art. 5º ley 115 de 1994 establece los mecanismos necesarios para llevar a cabo la misión encomendada por la constitución. A través de esta Ley se dan todos los mecanismos para que la Educación contribuya a la formación de personas que participen activamente en el desarrollo, y que permita al país entrar al Siglo XXI con dignidad y con mejores seres humanos.

Considera la Educación como 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.

Señala las normas generales para regular el servicio público de la Educación que cumple una función social a las necesidades inherentes de las personas, de la familia y la sociedad.²⁶

²⁶ COLOMBIA – CONGRESO DE LA REPUBLICA. Ley 44 de 1993 En página Web <http://www.indsoft.org/antipirateria/CircularSuperSociedades.html> Versión a 31 de marzo de 1999.

5. METODOLOGÍA

5.1 TIPO DE ESTUDIO

El tipo de estudio en base el cual se desarrolla el proyecto de investigación, es el explicativo; este permite identificar las causas de la problemática existente, determinar posibles efectos y plantear soluciones que permitan desarrollar con eficiencia los procesos estudiados.

5.2 LÍNEA DE INVESTIGACIÓN

La línea de investigación base de nuestro proyecto se denomina "Desarrollo, evaluación, implementación de estándares y paradigmas en Ingeniería del Software y ASQ", con un eje temático fundamentado en los Sistemas de Información Organizacional.

6. RECURSOS

6.1 RECURSOS HUMANOS

INTEGRANTES DEL PROYECTO

- **William Enrique Castillo Field:**

Estudios realizados.

- Técnico en Análisis y programación de computadores – CUC
- Estudiante de Ingeniería de Sistemas – 9 Semestre. Universidad Simón Bolívar.

- **Narciso Antonio Cantillo Álvarez**

Estudios realizados.

- Técnico en Mantenimiento y ensamble de computadores – CEP Autónoma
- Estudiante de Ingeniería de Sistemas – 9 Semestre. Universidad Simón Bolívar.

- **José Gregorio De La Cruz Del Toro**

- Tecnólogo en Informática - ITSA
- Estudiante de Ingeniería de Sistemas – 9 Semestre. Universidad Simón Bolívar.

- **Fredy Farit Renyz Suárez**

- Estudiante de Ingeniería de Sistemas – 9 Semestre. Universidad Simón Bolívar.

- **Paty Pedraza**

- Ingeniera de sistemas, especialista en Ingeniería del software
Universidad del norte.

6.2 RECURSOS FISICOS

DESCRIPCIÓN
Transporte
Impresión
Copias
Internet
Memoria USB – 512 Mb
CD ROM
Papelería

6.3 RECURSOS TECNOLOGICOS

DESCRIPCIÓN	CANTIDAD
Software	
Licencia Microsoft Windows XP Profesional SP2	2
Licencia Microsoft Office 2003 Profesional	1
Licencia Visual Basic 6.0	2
Licencia MySQL	1
Hardware	
PC Escritorio Clon	2
Impresora HP LaserJet b Color 2820	1

7. CRONOGRAMA

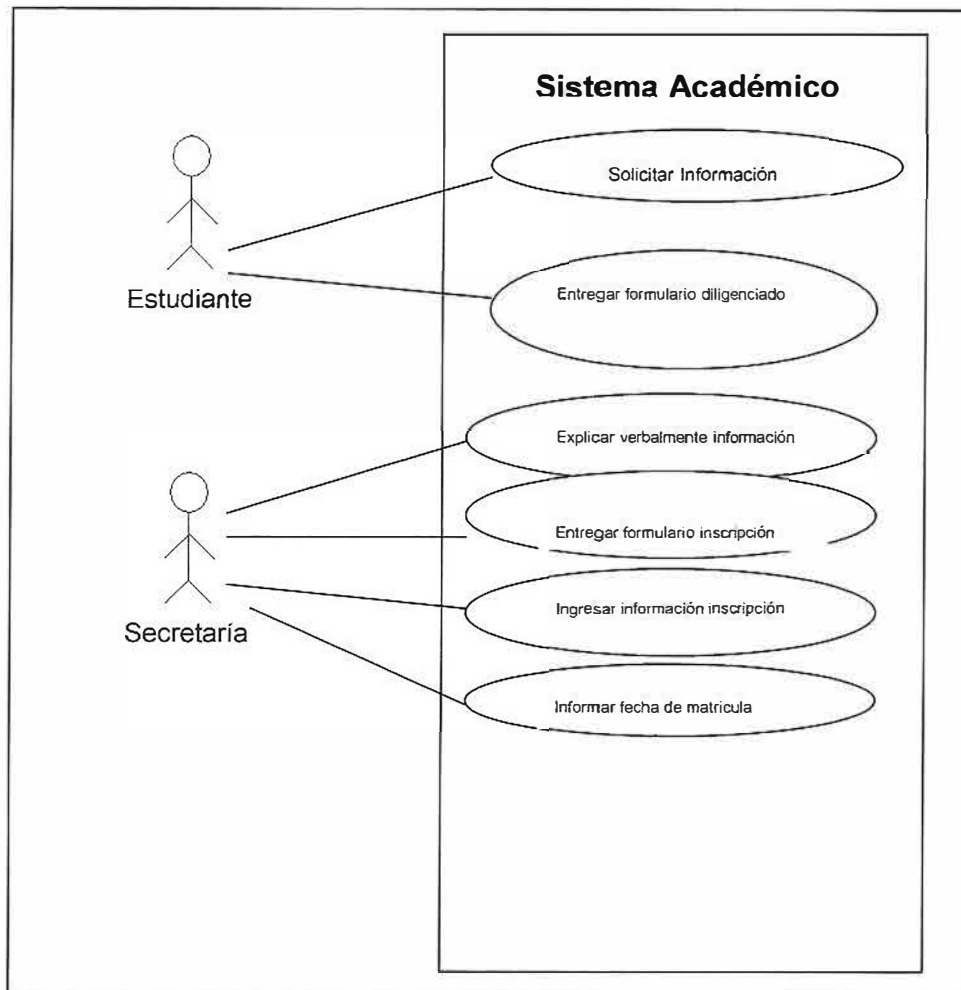
		2006																																								
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	
		Febrero				Marzo				Abril				Mayo				Junio				Julio				Agosto				Septiembre				Octubre				Noviembre				
No.	Actividades																																									
1	Recolección de datos en información																																									
2	Entrega 1er. Adelanto propuesta																																									
3	Entrega 2do. Adelanto propuesta																																									
4	Entrega Final Proyecto Formativa																																									
5	Vacaciones Mitad de año																																									
6	Adquisición de las herramientas Software para el desarrollo de proyecto																																									
7	Aprendizaje del manejo de las herramientas adquiridas																																									
8	Iniciación de la fase de diseño																																									
9	Depuración del diseño realizado																																									
10	Etapa de prueba																																									

		2007																																									
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4		
No.	Actividades	Febrero				Marzo				Abril				Mayo				Junio				Julio				Agosto				Septiembre				Octubre				Noviembre					
11	Descripción paso a paso de los procesos realizados en el manual																																										
12	Implementación del diseño																																										
13	Revisión final																																										
14	Corrección de errores																																										
15	Presentación y sustentación de proyectos																																										

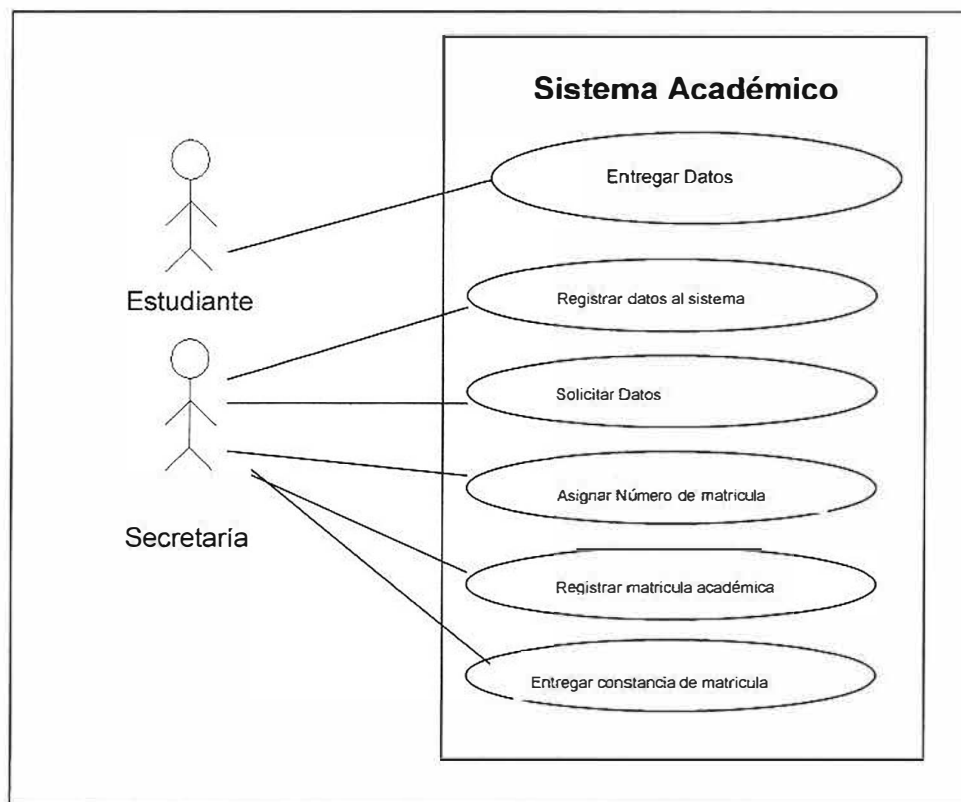
8. INGENIERÍA DE REQUISITOS

8.1 DESCRIPCIÓN DEL SISTEMA ACTUAL

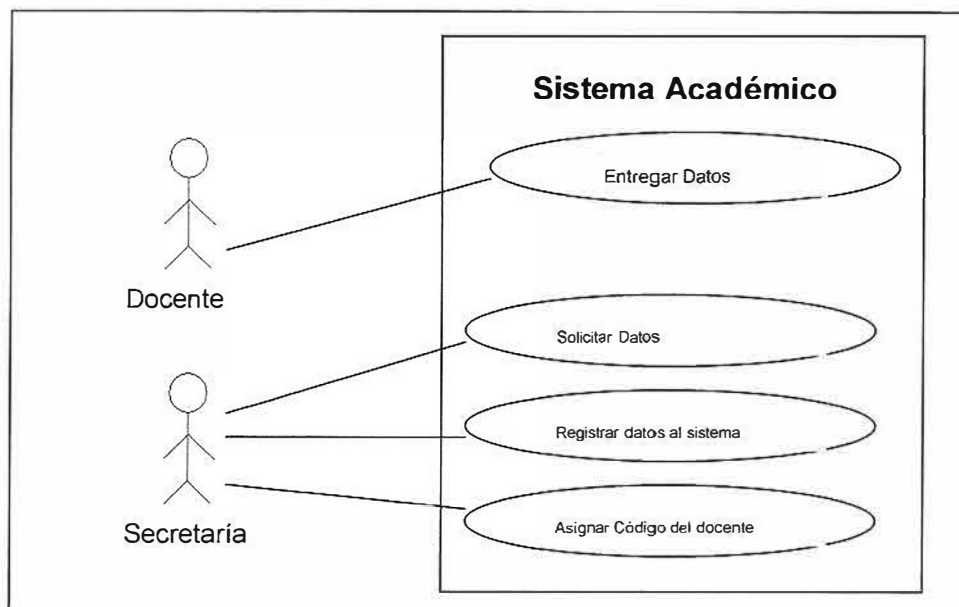
Caso de uso: Inscribir estudiante



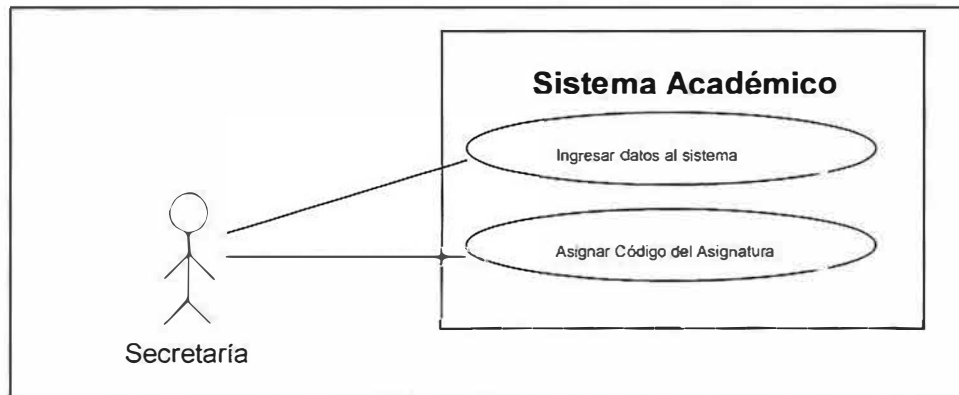
Caso de uso: Matricular Estudiante



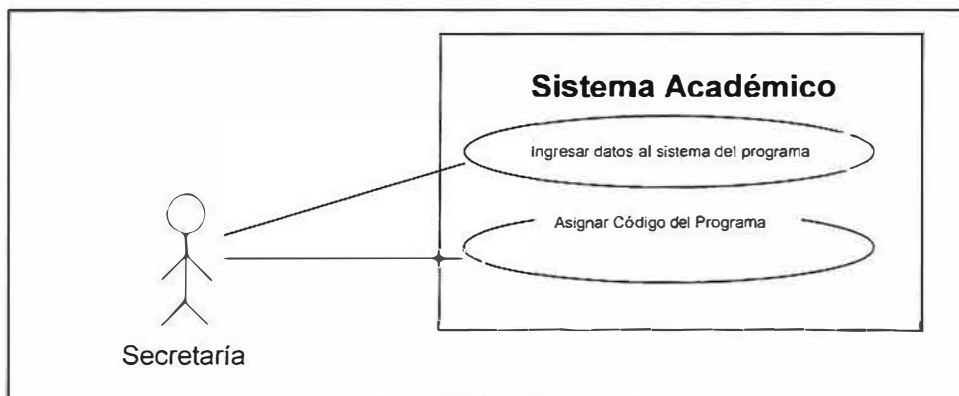
Caso de uso: Registrar docente



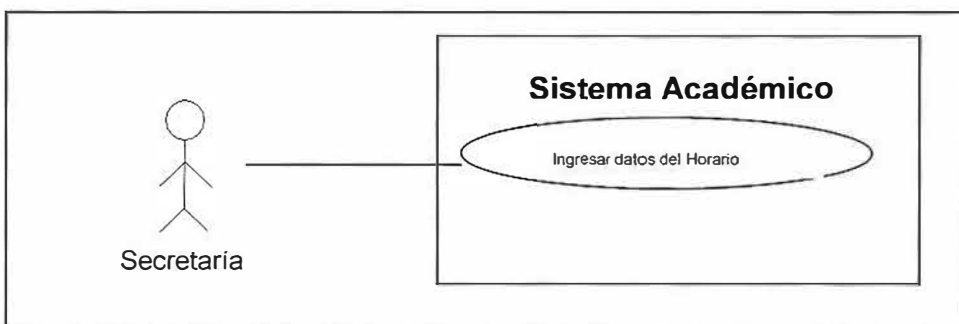
Caso de uso: Registrar Asignaturas



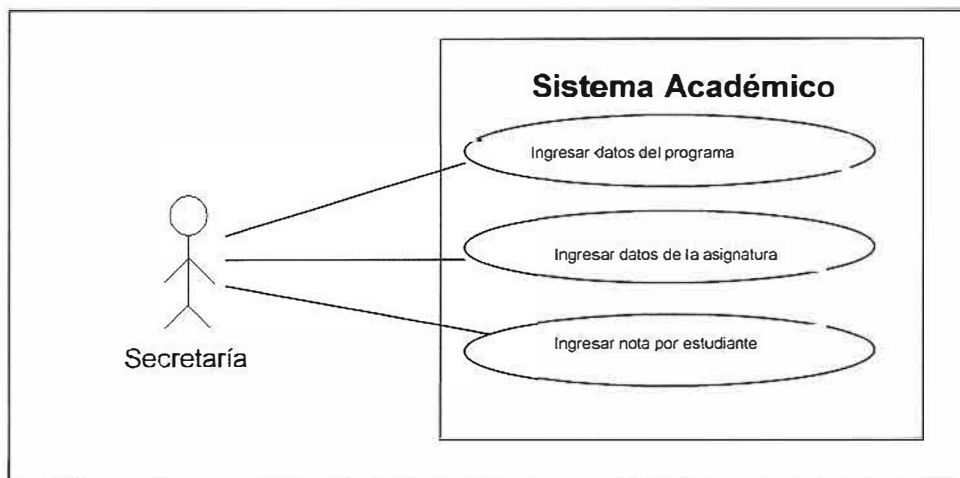
Caso de uso: Registrar programas



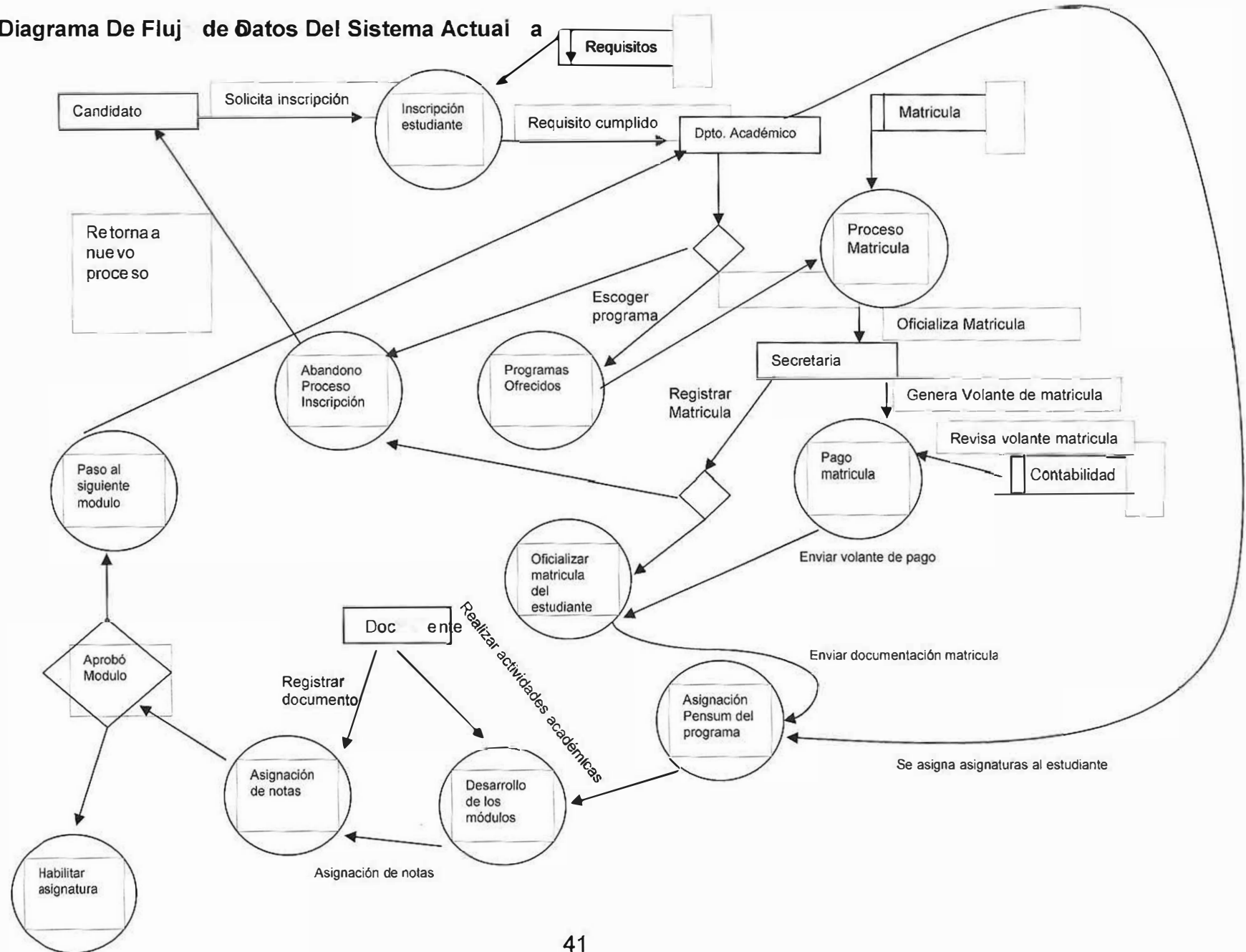
Caso de uso: Registrar horario



Caso de uso: Registrar Notas



8.2 Diagrama De Flujo de Datos Del Sistema Actual a



8.3. ANÁLISIS DE REQUISITOS

8.3.1 LISTADOS DE REQUERIMIENTOS CODETEC

1. Registro de las personas inscritas en la corporación (información, personal, profesional, academia).
2. Registros de las personas matriculadas en la corporación (información, personal, profesional, academia).
3. Registros de las personas egresadas de la corporación (información, personal, profesional, academia).
4. Registrar la información de los cursos, programas, especializaciones que ofrece la institución con cada uno de los pensul y requisitos para cursarlos.
5. Registro de cada docente de la corporación y personal a fin (información, personal, profesional, academia)
6. Registro de los horarios de cada docente (información, hora, programa, curso)
7. Estadística de rendimiento de cada programa ofrecido (información, matrícula, rendimiento académico, historias de meses anteriores).
8. Registros del rendimiento académico de cada estudiante (información, notas, logros, módulos hechos, módulos por alcanzar, rendimiento)
9. Estadísticas gráficas sobre el rendimiento de cada programa (rendimiento académico).
10. Que ayude a tomar decisiones sobre la apertura de otros o nuevos programas (base de datos de personas con sus preferencias, teorías, profesionales, estudiantes universitarios y trabajadores).
11. Generación de los aportes concernientes a cada registro del sistema.
12. Fácil manejo de aplicaciones
13. Que permita obtener ayuda de forma rápida
14. Que permita el acceso al sistema dependiendo el perfil del usuario
15. Que permita acceso remoto
16. Registro de los horarios de los estudiantes

17. Copia de seguridad

18. Consulta de notas de los estudiantes

8.3.2 LISTADO REQUERIMIENTO ESPECÍFICOS

1. Registro de las personas inscritas en la corporación
2. Registro de las personas matriculadas
3. Registro de las personas egresadas
4. Registro información programas ofrecidos
5. Registro de los docentes
6. Registro de los horarios de los docentes
7. Registros de los horarios de los estudiantes
8. Rendimiento de los programas
9. Rendimiento de los estudiantes
10. Información sobre nuevos programas
11. Generación de reportes
12. Fácil manejo de aplicaciones
13. Ayudas de soporte
14. Acceso al sistema dependiendo el perfil
15. Módulo para evaluar estudiantes
16. Copia de seguridad

8.3.3 REQUERIMIENTOS FUNCIONALES

1. Registros de las personas inscritas en la corporación
2. Registro de las matriculas
3. Registro de las personas egresadas
4. Registro información programas ofrecidos
5. Registros de docentes
6. Registros horarios docentes
7. Registros horarios estudiantes

8. Rendimientos de los programas
9. Rendimientos de los estudiantes
10. Información sobre nuevos programas
11. Acceso al sistema dependiendo el perfil

8.3.4 REQUERIMIENTOS NO FUNCIONALES

1. Generación de reportes
2. Ayudas de soporte
3. Fácil manejo de aplicaciones
4. Copia de seguridad

9. INGENIERÍA DE INFORMACIÓN

9.1 MISIÓN

La Corporación Técnica De Estudios Especializados Del Caribe “CODETEC”, tiene como misión formar ciudadanos profesionales, consientes, responsables, críticos, investigadores de alta calidad, dotados de una cultura humanista, científica, técnica; provistos de un comportamiento ético ante la sociedad; independiente de la raza, credo, nacionalidad fomentando el espíritu participativo, pluralista, deliberante, creativo y autónomo.

La formación integral ofrecida por CODETEC hace posible el trabajo en equipo, la responsabilidad de llevar a un término, proyectos iniciados, el compromiso social y la capacidad de asumir el proceso continuo de aprendizaje como éxito personal.

9.2 VISIÓN

La Corporación Técnica De Estudios Especializados Del Caribe “CODETEC”, se proyecta como una institución con un modelo de desarrollo que consolide su autonomía hacia la construcción de una dinámica, administrativa y académica facilitada de la búsqueda del liderazgo científico, pedagógico y culturalmente trascienda en el ámbito regional, nacional ó internacional.

Sus egresados serán reconocidos por sus compromisos en la solución de problemas de la sociedad; por su responsabilidad en el desarrollo de sus proyectos y su espíritu inquieto en busca del mejoramiento continuo de sus conocimientos.

9.3 GENERALIDADES

CODETEC cuenta con los siguientes programas académicos:

PROGRAMAS TÉCNICOS LABORALES

- Diseño Gráfico.
- Análisis y programación de computadores.
- Mercadeo publicidad y venta.
- Contabilidad sistematizada.
- Secretariado ejecutivo sistematizado.
- Administración financiera.

PROGRAMAS DE SALUD

- Cosmetología y estética con énfasis en terapias alternativa.
- Auxiliar de enfermería.
- Promotor de salud.

CENTRO DE IDIOMAS

- Inglés especializado.

CURSOS

- Informática básica.
- Excel avanzado.
- Access.
- Mantenimiento y reparación de computadores.
- Internet.
- Pagina Web.
- Autocad.
- Vacunación.
- Primeros auxilios.

CURSOS DE ASCENSO AL ESCALAFÓN NACIONAL DOCENTE (3,4,5 y 6 Créditos)

- Informática para docentes.
- Inglés para docentes.

ESPECIALIZACIONES

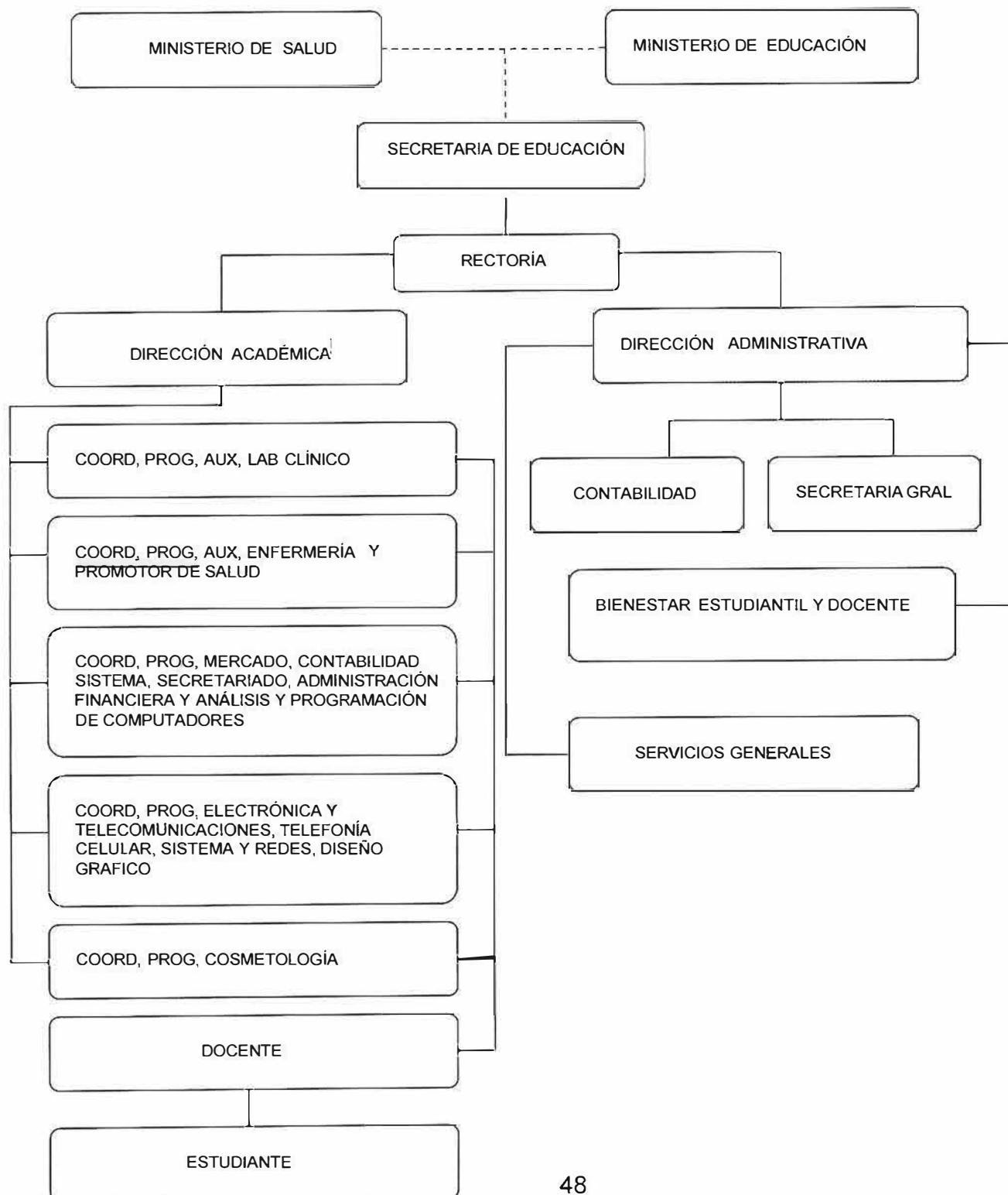
- Informática y Multimedia. Convenio Fundación Universitaria Los Libertadores.

DIPLOMADOS

- Competencias laborales.
- Sistemas de gestión de la calidad.
- Estética para profesionales de la salud.
- Cosmiatría facial.
- Urgencias y cuidado críticos de adultos.
- Urgencias y cuidado críticos de neonatos.

Cabe agregar que todos estos programas tienen una duración máxima en promedio de 3 semestres, de ahí a que la población estudiantil varíe mucho en el tiempo.

9.4 ORGANIGRAMA



REQUERIMIENTOS NO FUNCIONALES

Referencia No.	Función
RNF1	Ser fácil de utilizar
RNF 2	Interfaz gráfica amigable
RNF 3	Documentación de ayuda
RNF4	Realizar Backups de información
RNF5	Mantenimiento del sistema accesible

CASOS DE USO DE ALTO NIVEL

1. Caso de uso: Inscribir candidato

Actores: Secretaria, Candidato

Descripción: Un candidato se acerca al Departamento de Registro Académico para realizar la respectiva inscripción al programa de su elección. La secretaria solicita datos del candidato para generar volante de pago de inscripción, luego de cancelado el valor de la inscripción, la secretaria registra el pago en el sistema, y oficializa la inscripción

Tipo: Primario

Referencias: R2, R3, R4

2. Caso de uso: Matricular estudiante

Actores: Secretaria, Estudiante

Descripción: El candidato previamente inscrito se acerca al Departamento de Registro Académico con los documentos requeridos para legalizar la matricula, la secretaria solicita requisitos. Después de confirmar que todos los documentos requeridos se encuentran en orden procede a registrar la información en el sistema para entregar el volante de pago. Después de cancelado el volante de matricula el estudiante entrega a la secretaria el volante de pago, ella registra pago de matricula correspondiente y entrega listado de asignaturas del semestre a cursar.

Tipo: Primario

Referencias: R5, R6, R7, R8

3. Caso de uso: Registrar Docente

Actores: Secretaria

Descripción: La secretaria registra los datos generales de cada docente.

Tipo: Primario

Referencias: R9

4. Caso de uso: Registrar Asignatura

Actores: Secretaria

Descripción: La secretaria ingresa al sistema los datos de la asignatura.

Tipo: Primario

Referencias: R10

5. Caso de uso: Registrar Programa

Actores: Secretaria

Descripción: La secretaria registra en el sistema los datos de cada uno de los programas que están acreditados.

Tipo: Primario

Referencias: R11

6. Caso de uso: Registrar Pensum

Actores: Secretaria

Descripción: La secretaria ingresa al sistema la información de cada pensum por programa.

Tipo: Primario

Referencias: R12

7. Caso de uso: Ingresar Notas

Actores: Secretaria, Docente

Descripción: Un docente se acerca al Departamento de Registro Académico para hacer la entrega oficial a la secretaria del listado de notas de la(s) asignatura(s) que el dicta. La secretaria recibe las notas por parte del docente y procede a ingresarlas al sistema.

Tipo: Primario

Referencias: R19, R20

8. Caso de uso: Consultar Notas

Actores: Estudiante

Descripción: Un estudiante consulta las calificaciones que ha obtenido en cada una de las asignaturas en las cuales se encuentra matriculado.

Tipo: Primario

Referencias: R13

9. Caso de uso: Registrar usuario

Actores: Administrador del sistema

Descripción: El administrador del sistema registra cada uno de los usuarios que van a utilizar el sistema.

Tipo: Primario

Referencias: R14

10. Caso de uso: Registrar perfiles de usuarios

Actores: Administrador del sistema

Descripción: El administrador del sistema le asigna a cada usuario el tipo de perfil de acceso al sistema.

Tipo: Primario

Referencias: R15, R16

11. Caso de uso: Iniciar sesión.

Actores: Administrador del sistema, Docente, Secretaria

Descripción: Un usuario del sistema desea iniciar una sesión en el sistema, para ella digita su de nombre de usuario y contraseña.

Tipo: Primario

Referencias: R1

12. Caso de uso: Registrar permisos

Actores: Administrador del sistema

Descripción: El administrador del sistema le asigna a cada usuario el área de trabajo que tendrá derecho a acceder dentro de su perfil.

Tipo: Primario

Referencias: R15, R17

13. Caso de uso: Generar reportes

Actores: Administrador del sistema, Secretaria

Descripción: El Departamento De Registro Académico necesita presentar Reportes sobre los siguientes aspectos:

- Listado de estudiantes por asignatura.
- Listado de asignaturas por programa.
- Listado de estudiantes por programa.
- Listado de docentes por programa.
- Numero de personas inscritas por programas.
- Relación de estudiantes por promedio de notas.
- Listado de estudiante matriculados por periodo.
- Listado de candidatos inscritos por periodo.
- Listado de estudiantes reprobados por asignatura y programa.
- Listado de estudiantes aprobados por asignatura y programa.
- Certificado de estudios.

- Certificado de notas.

Tipo: Primario

Referencias: R18

CASOS DE USO EXTENDIDOS

1. Caso de uso: Inscribir candidato

Actores: Secretaria, Candidato

Propósito: Registrar los datos de inscripción de un candidato en el sistema.

Tipo: Primario

Referencias: R2, R3, R4

Curso Normal de los eventos

1. Un candidato se acerca al Departamento de Registro Académico para realizar la respectiva inscripción al programa de su elección.
2. La secretaria solicita datos del candidato.
3. La secretaria registra los datos en el sistema.
4. Utilizando el sistema La secretaria genera e imprime el volante de pago de inscripción.
5. Después de cancelado el volante de pago de inscripción el candidato se acerca al Departamento de Registro Académico para oficializar la inscripción.
6. La secretaria registra el pago en el sistema y oficializa la inscripción.

Curso alterno de los eventos

Línea 3: Digitación errónea de datos del candidato.

Línea 4: Error en la impresión del volante de pago.

2. Caso de uso: Matricular estudiante

Actores: Secretaria, Estudiante

Propósito: Registrar los datos de Matricula de un estudiante en el sistema.

Tipo: Primario

Referencias: R5, R6, R7, R8

Curso Normal de los eventos

1. El candidato previamente inscrito se acerca al Departamento de Registro Académico con los documentos requeridos para legalizar la matricula.
2. La secretaria solicita requisitos
3. La secretaria confirma que todos los documentos requeridos se encuentran en orden.
4. La secretaria registra la información del estudiante a matricular en el sistema.
5. Utilizando el sistema la secretaria genera e imprime el volante de pago de matricula.
6. Después de cancelado el volante de pago de matricula el estudiante se acerca al Departamento de Registro Académico para oficializar la matricula.
7. La secretaria registra el pago en el sistema y oficializa la matricula.
8. La secretaria registra el programa y las asignaturas del semestre a cursar.
9. El sistema genera un Reporte con el listado de asignaturas del semestre a cursar.

Curso alterno de los eventos

Línea 4: El estudiante no se encuentra inscrito.

Línea 4: Digitación errónea de datos del estudiante.

Línea 5: Error en la impresión del volante de pago.

Línea 8: El sistema genera un reporte con el listado de asignaturas a cursar no correspondiente con el del estudiante.

3. Caso de uso: Registrar Docente

Actores: Secretaria

Propósito: Registrar los docentes al sistema.

Tipo: Primario

Referencias: R9

Curso Normal de los eventos

1. La secretaria solicita a la dirección la hoja de vida del docente.
2. La secretaria registra datos los del docente en el sistema.

Curso alternativo de los eventos

Línea 2: Digitación errónea de datos del docente.

4. Caso de uso: Registrar Asignatura

Actores: Secretaria

Propósito: Registrar una Asignatura en el sistema.

Tipo: Primario

Referencias: R10

Curso Normal de los eventos

1. La secretaria solicita al Secretario Académico la información de la asignatura a registrar al sistema.
2. La secretaria registra el código del programa al que corresponde esta asignatura.
2. La secretaria registra los datos de la asignatura en el sistema.

Curso alternativo de los eventos

Línea 2: Digitación errónea de datos de la asignatura.

5. Caso de uso: Registrar Programa

Actores: Secretaria

Propósito: Registrar un Programa en el sistema.

Tipo: Primario

Referencias: R11

Curso Normal de los eventos

1. La secretaria solicita al Secretario Académico la información del Programa a registrar en el sistema.
2. La secretaria registra los datos del Programa en el sistema.

Curso alterno de los eventos

Línea 2: Digitación errónea de datos del programa.

6. Caso de uso: Registrar Pensum

Actores: Secretaria

Propósito: Registrar el Pensum de un Programa en el sistema.

Tipo: Primario

Referencias: R12

Curso Normal de los eventos

1. La secretaria solicita al Secretario Académico la información del Programa y las asignaturas asociadas a este, para registrar el Pensum en el sistema.
2. La secretaria registra los datos del Pensum en el sistema.

Curso alterno de los eventos

Línea 2: Digitación errónea de datos del Pensum.

Línea 2: Un programa no se encuentra registrada en el sistema.

Línea 3: Una asignatura no se encuentra registrada en el sistema.

7. Caso de uso: Ingresar Notas

Actores: Secretaria, Docente

Propósito: Registrar las notas en el sistema de los estudiantes.

Tipo: Primario

Referencias: R19, R20

Curso Normal de los eventos

1. Un docente se acerca al Departamento de Registro Académico para hacer la entrega oficial a la secretaria del listado de notas de la(s) asignatura(s) que el dicta.
2. La secretaria recibe las notas por parte del docente.
3. La secretaria ingresa código de la asignatura.
4. La secretaria verifica que el estudiante este matriculado.
5. La secretaria procede a registrar las notas al sistema.

Curso alterno de los eventos

Línea 3: El docente no se encuentra registrado en el sistema.

Línea 3: El estudiante no se encuentre matriculado.

Línea 3: El estudiante está matriculado, pero no tiene registrada la asignatura.

8. Caso de uso: Consultar Notas

Actores: Estudiante

Propósito: El estudiante desea informarse sobre las notas obtenidas en las asignaturas que cursa.

Tipo: Primario

Referencias: R13

Curso Normal de los eventos

1. Un estudiante se acerca a una terminal de consulta de notas para conocer las notas obtenidas en las asignaturas que cursa.
2. El estudiante ingresa su número de cedula para realizar la consulta.
3. El sistema despliega en pantalla las asignaturas y las calificaciones que posee el estudiante por un lapso de tiempo de 20 segundos.

Curso alterno de los eventos

Línea 3: El número de cedula ingresado es erróneo, por lo cuál el sistema no lo deja consultar.

Línea 3: El estudiante no se encuentra matriculado, por tal motivo no muestra ningún resultado.

9. Caso de uso: Registrar usuario

Actores: Administrador del sistema

Propósito: Registrar un usuario en el sistema.

Tipo: Primario

Referencias: R14

Curso Normal de los eventos

1. El administrador del sistema registra un usuario en el sistema, asignándole un nombre de usuario y una contraseña.

10. Caso de uso: Registrar perfiles de usuarios

Actores: Administrador del sistema

Propósito: Asignar a los usuarios registrados, un perfil específico de acceso al sistema.

Tipo: Primario

Referencias: R15, R16

Curso Normal de los eventos

1. El administrador del sistema busca un usuario.
2. El administrador del sistema le asigna a cada usuario un perfil específico de acceso al sistema, dependiendo de los procesos que maneje

Curso alterno de los eventos

Línea 1: El usuario no se encuentra registrado en el sistema.

11. Caso de uso: Iniciar sesión.

Actores: Administrador del sistema, Docente, Secretaria

Propósito: Iniciar una sesión en el sistema.

Tipo: Primario

Referencias: R1

Curso Normal de los eventos

1. Un usuario del sistema desea iniciar sesión.
2. El usuario digita su nombre de usuario y la contraseña para acceder al sistema.

Curso alternativo de los eventos

Línea 2: El usuario no se encuentra registrado en el sistema.

Línea 2: El nombre de usuario o la contraseña no son válidos.

12. Caso de uso: Registrar permisos

Actores: Administrador del sistema

Propósito: Asignar los permisos de los usuarios registrados.

Tipo: Primario

Referencias: R15, R17

Curso Normal de los eventos

1. El administrador del sistema busca un usuario.
2. El administrador del sistema le asigna permisos a cada usuario dependiendo del perfil que le fue asignado previamente, el cual le permitirá tener acceso a una determinada área.

Curso alternativo de los eventos

Línea 1: El usuario no se encuentra registrado en el sistema.

13. Caso de uso: Generar reportes

Actores: Administrador del sistema, Secretaria

Propósito: Generar reportes sobre consultas específicas.

Tipo: Primario

Referencias: R18**Curso Normal de los eventos**

1. El Administrador del sistema ó la Secretaria necesita presentar Reportes sobre los siguientes aspectos:

- Listado de estudiantes por asignatura.
- Listado de asignaturas por programa.
- Listado de estudiantes por programa.
- Listado de docentes por programa.
- Numero de personas inscritas por programas.
- Relación de estudiantes por promedio de notas.
- Listado de estudiante matriculados por periodo.
- Listado de candidatos inscritos por periodo.
- Listado de estudiantes reprobados por asignatura y programa.
- Listado de estudiantes aprobados por asignatura y programa.
- Certificado de estudios.
- Certificado de notas.

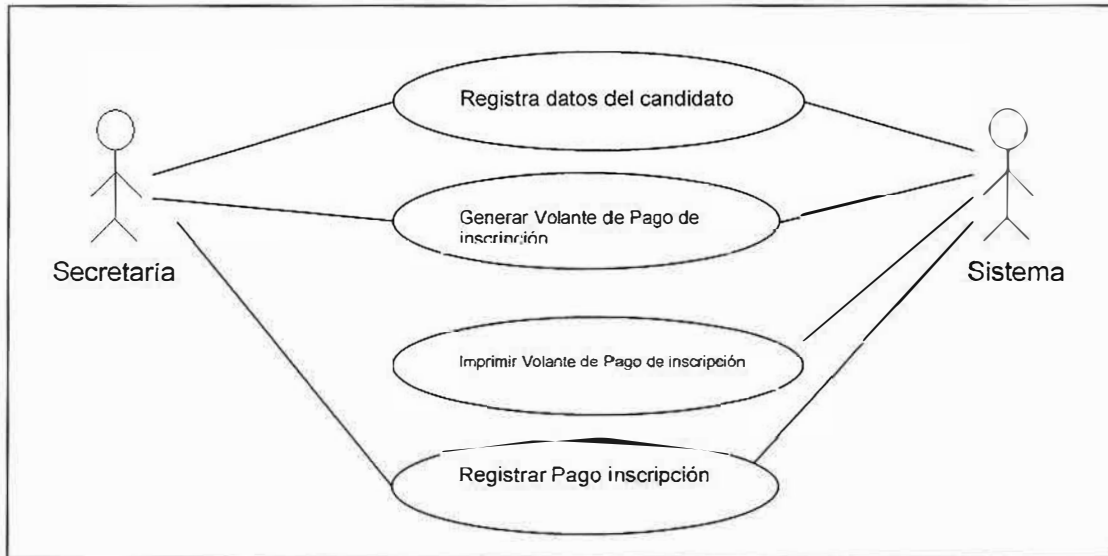
2. El Administrador del sistema ó la Secretaria escoge el reporte que desea consultar.

3. El sistema muestra el reporte solicitado.

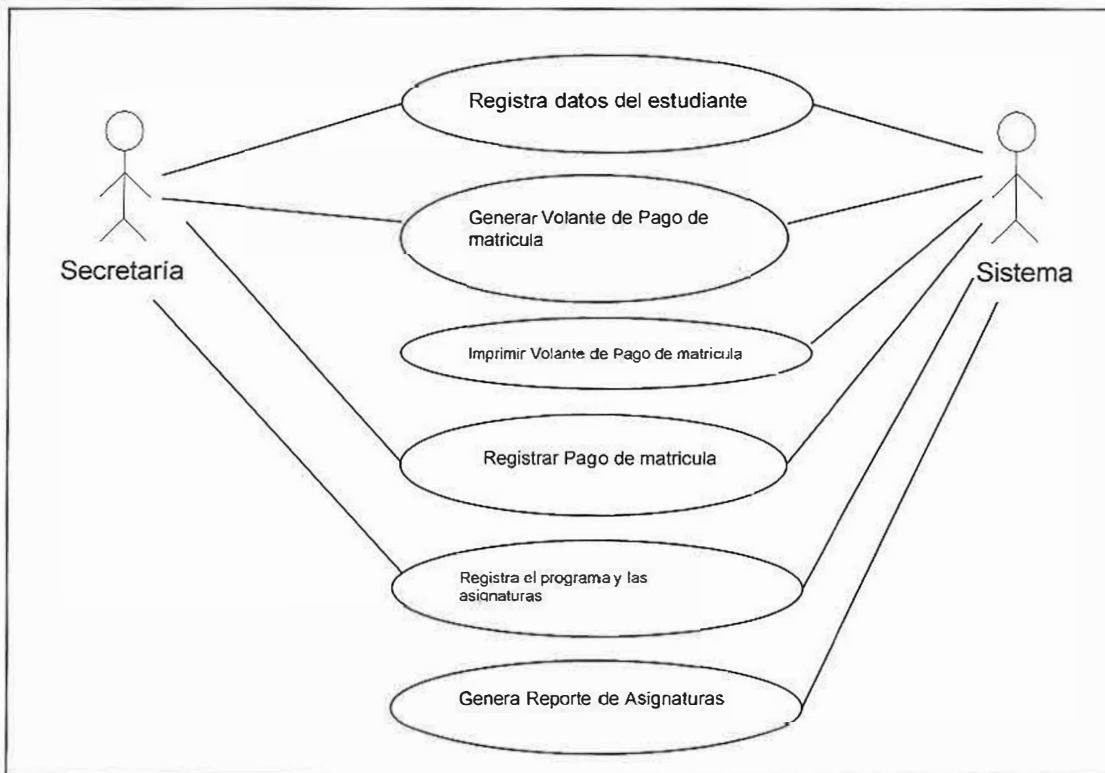
4. El sistema permite guardar en disco e imprimir el reporte.

DIAGRAMA DE CASOS DE USOS – SISTEMA PROPUESTO

Caso de uso: Inscribir estudiante



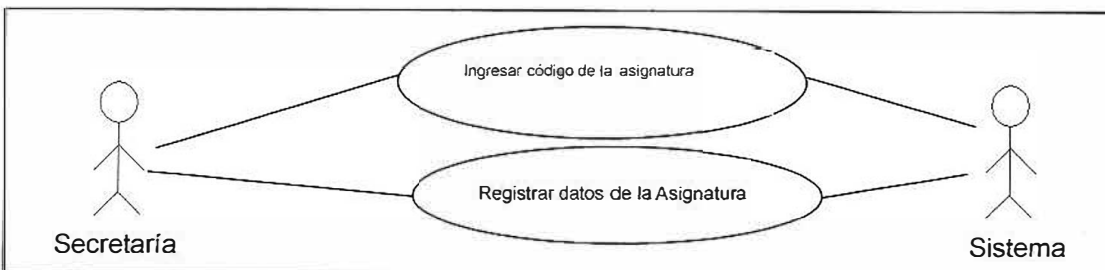
Caso de uso: Matricular Estudiante



Caso de uso: Registrar docente



Caso de uso: Registrar Asignatura



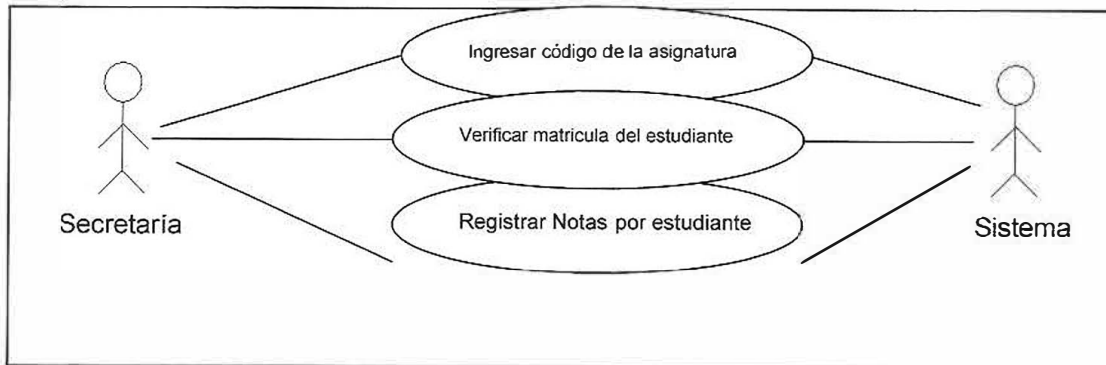
Caso de uso: Registrar Programa



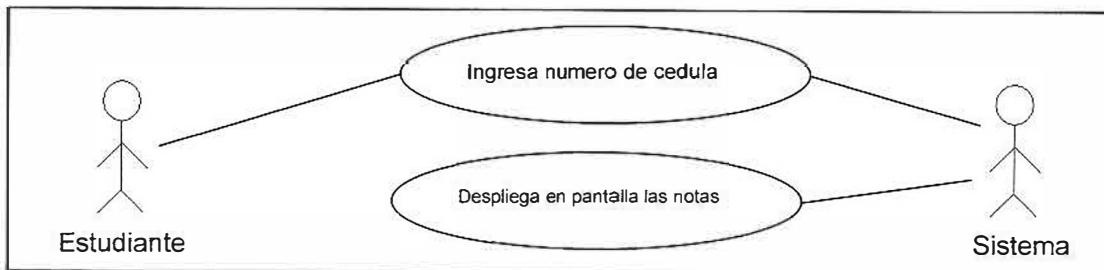
Caso de uso: Registrar Pensum



Caso de uso: Ingresar Notas



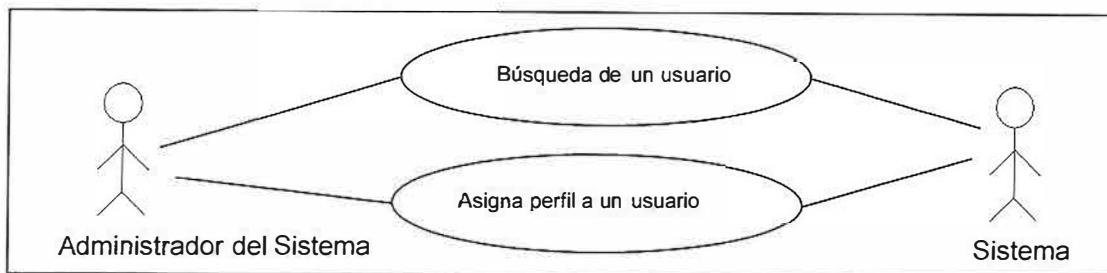
Caso de uso: Consultar Notas



Caso de uso: Registrar usuario



Caso de uso: Registrar perfiles de usuario



Caso de uso: Iniciar Sesión



Caso de uso: Registrar Permisos de usuario



Caso de uso: Generar Reportes



10.1 ESPECIFICACIÓN DE ENTIDADES

* Llave Primaria

Candidato

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cedula *	numérico	20	Cédula del candidato	S
nombres	texto	50	Nombre del candidato	S
apellidos	texto	50	Apellidos del candidato	S
direccion	texto	50	Dirección del candidato	S
telefono	numérico	20	Teléfono del candidato	S
email	texto	50	email del candidato	N
Sexo	texto	1	Sexo del candidato	S
organización	texto	50	Nombre de la Organización ó institución de procedencia	N
telefono_organizacion	numérico	20	Teléfono de la organización	N
direccion_organizacion	texto	50	Dirección de la organización	N

Docente

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cedula *	numérico	20	Cédula del docente	S
nombres	texto	50	Nombre del docente	S
apellidos	texto	50	Apellido del docente	S
direccion	texto	50	Dirección del docente	S
Telefono	numérico	20	Teléfono del docente	S
email	texto	50	email del docente	N
Sexo	texto	1	Sexo del docente	S
Info_profesional	texto	100	Referencia profesional del docente	S

Estudiante

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cedula *	numérico	20	Cédula del estudiante	S
nombres	texto	50	Nombre del estudiante	S
apellidos	texto	50	Apellido del estudiante	S
direccion	texto	50	Dirección del estudiante	S
telefono	numérico	20	Teléfono del estudiante	S
email	texto	50	email del estudiante	N
Sexo	texto	1	Sexo del estudiante	S
estudios	texto	100	Referencia de estudios realizados por el estudiante	S
fechanacimiento	fecha		fecha de nacimiento del estudiante	S
estadocivil	texto	15	estado civil del estudiante	S

Usuario

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cedula *	numérico	20	Cédula del usuario	S
nombres	texto	50	Nombre del usuario	S
apellidos	texto	50	Apellido del usuario	S
cargo	texto	50	Dirección del usuario	S
contraseña	texto	20	Teléfono del usuario	S
nusuario *	texto	50	Nombre del usuario en el sistema	S

Perfil

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
codigo *	numérico	20	Código del perfil	S
rango	texto	5	Nombre del perfil	S

Permiso

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
codigo *	numérico	20	Código del permiso	S
nombre	texto	5	Nombre del permiso	S
tipoacceso	texto	20	tipo de acceso que tendrá el permiso ya sea total o con ciertas restricciones	S

Matricula

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
codigo *	numérico	20	Código de la matricula	S
fecha	fecha/hora		Fecha de la matricula	S
hora	fecha/hora		Hora de la matricula	S
cedula	numérico	20	Cedula de estudiante	S

Programa

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
Nombre	texto	50	Nombre del programa	S
codigo *	numérico	20	Código del programa	S
Duración	numérico	5	Duración del programa	S
Registro	texto	50	registro lcfes del programa	S
Valor	numérico	20	valor del programa	S

Pensum

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cod_programa *	numérico	20	Código del programa	S
Cod_asignatura	numérico	20	Código de la asignatura	S
Intensidad	numérico	20	Intensidad del pensum	S

Asignatura

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
nombre	texto	50	Nombre de la asignatura	S
codigo *	numérico	20	Código de la asignatura	S

Docente_Programa

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cedula *	numérico	20	Cédula del docente	S
cod_programa *	numérico	20	Código del programa	S

Docente_Asignatura

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cedula *	numérico	20	Cédula del docente	S
cod_asignatura *	numérico	20	Código de la asignatura	S

Inscripción

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cod_programa *	numérico	20	Código del programa	S
cedula_candidato *	numérico	20	Cédula del candidato	S

Estudiante_programa

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cod_programa *	numérico	20	Código del programa	S
cedula_estudiante *	numérico	20	Cédula del estudiante	S
Modulo	numérico	2	Modulo que cursa el estudiante	S

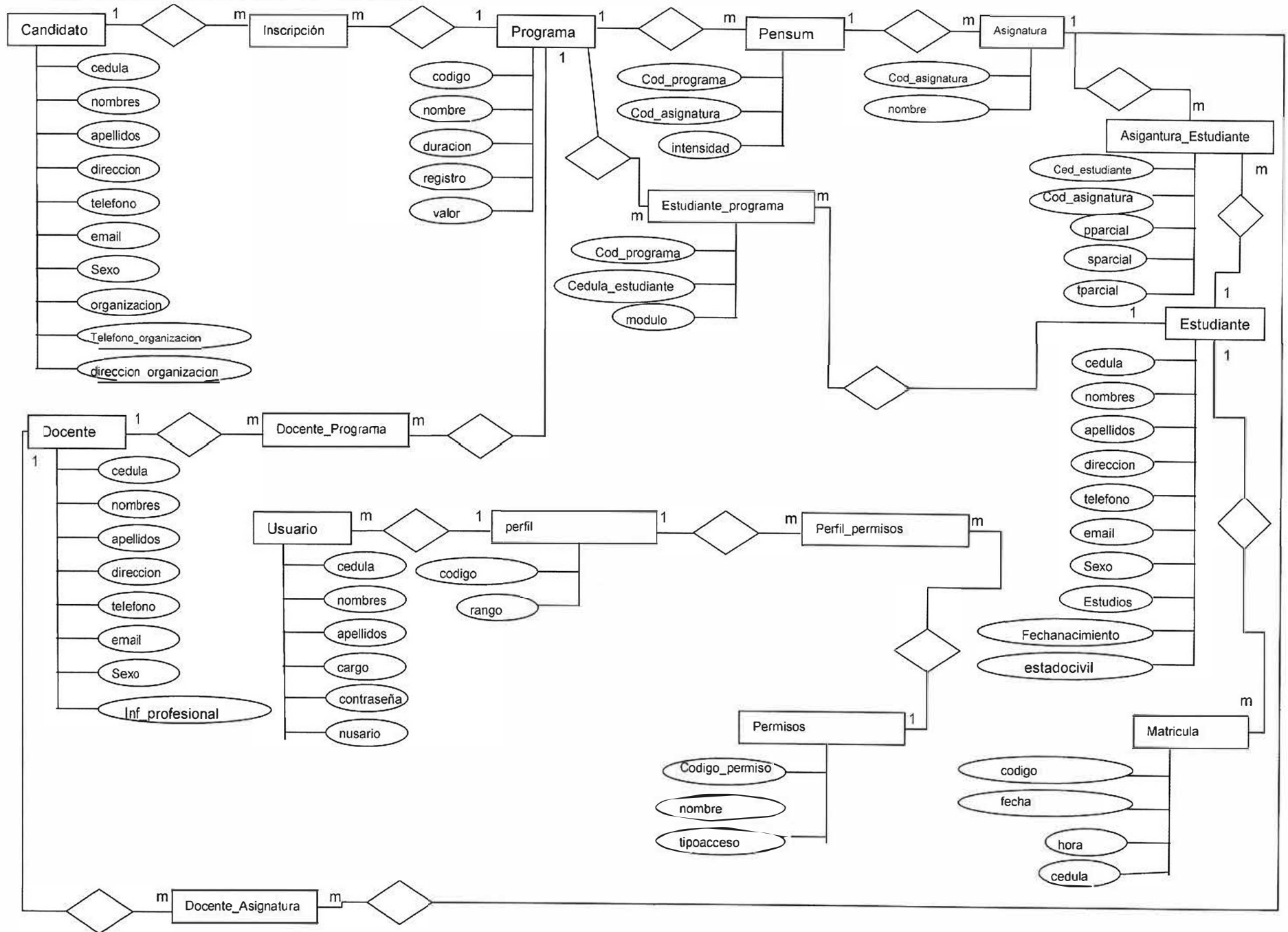
Asignatura_Estudiante

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cod_asignatura *	numérico	20	Código del programa	S
cedula_estudiante *	numérico	20	Cédula del estudiante	S
pparcial	numérico	2	Nota primer parcial	S
sparcial	numérico	2	Nota segundo parcial	
tparcial	numérico	2	Nota tercer parcial	

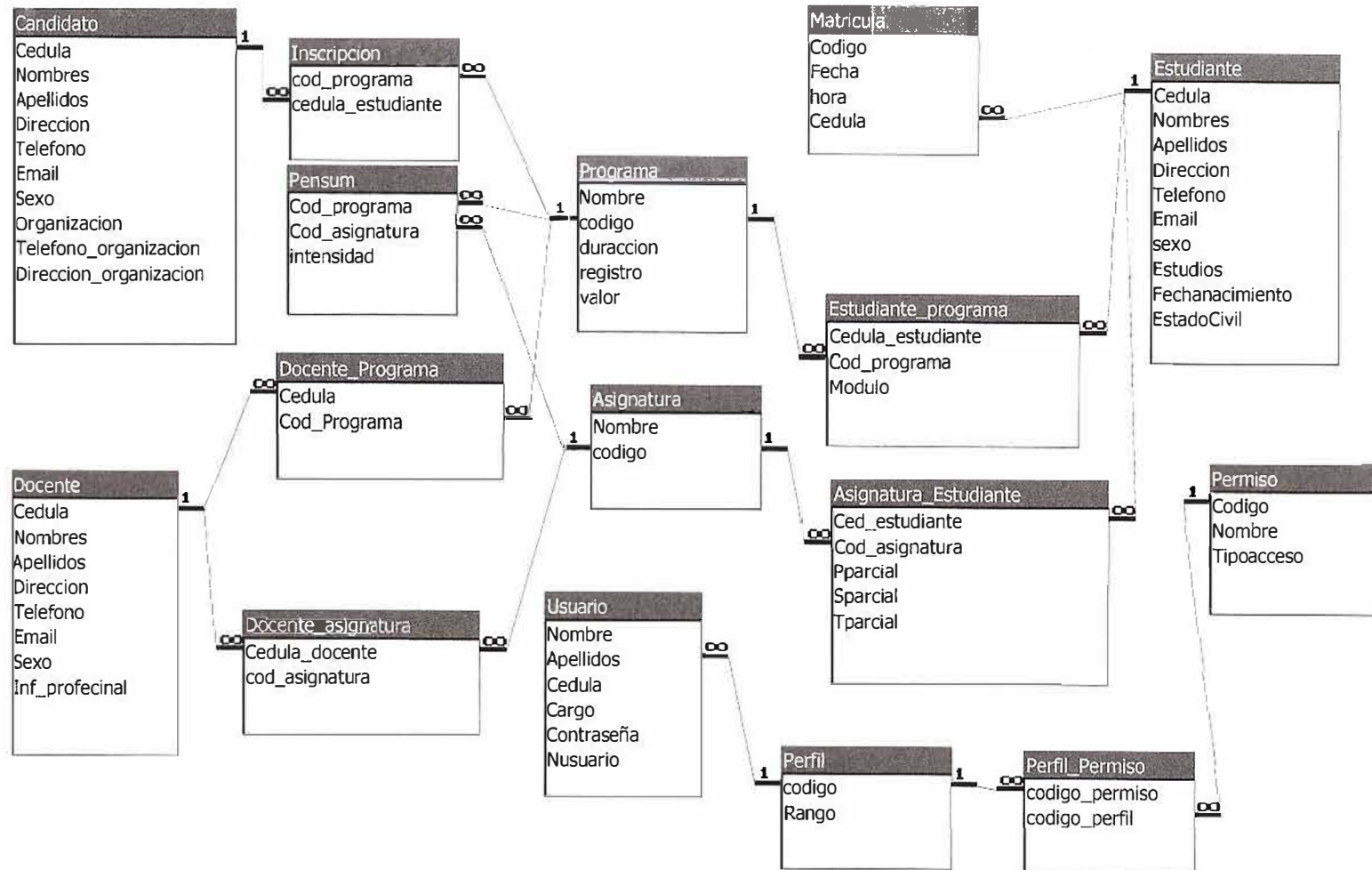
Perfil_Permiso

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cod_permiso *	numérico	20	Código del permiso	S
Cod_perfil *	numérico	20	Cédula del perfil	S

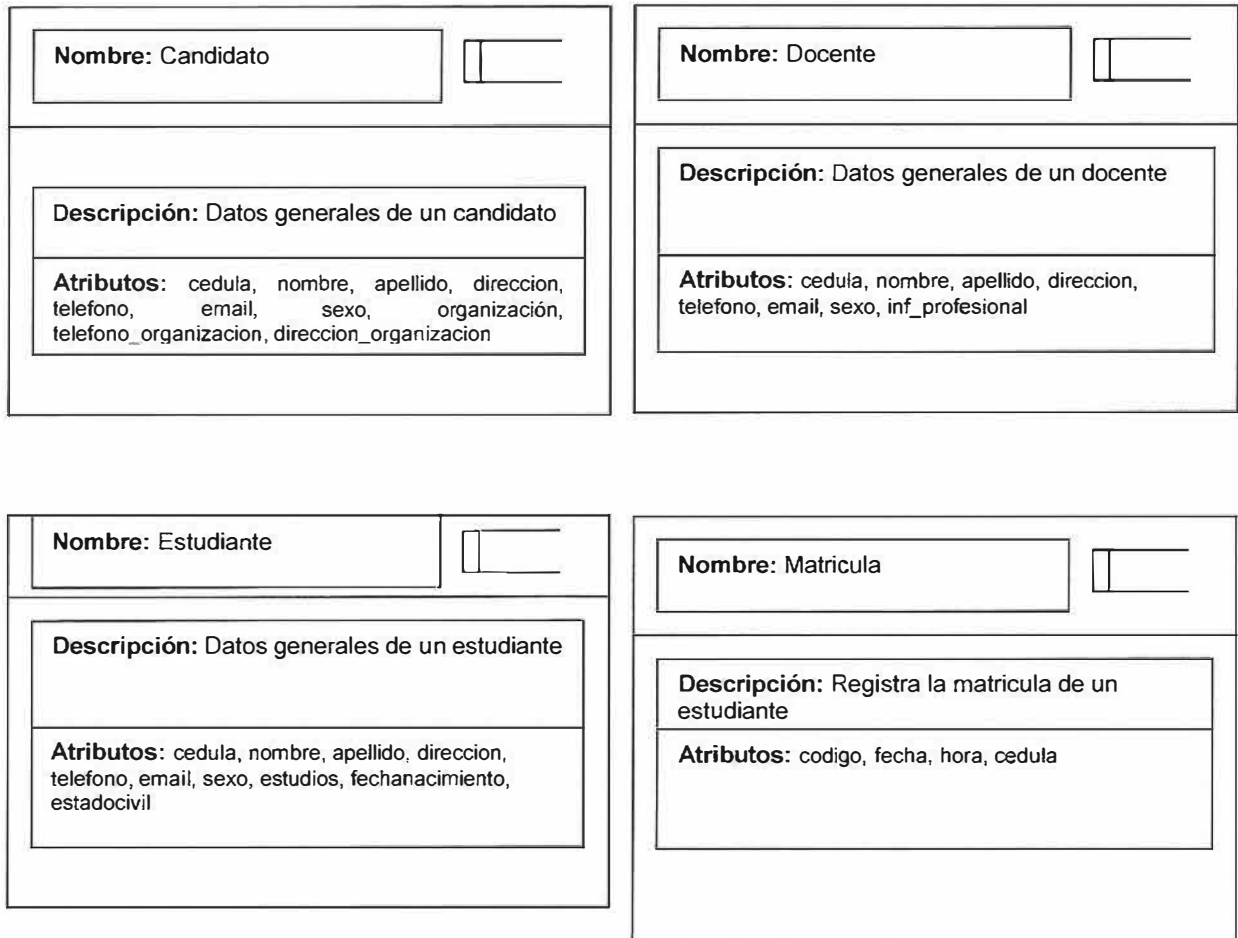
10.3 MODELO ENTIDAD RELACIÓN



10.4 MODELO RELACIONAL



10.5 DICCIONARIO DE DATOS



Nombre: Programa	
Descripción: Datos generales de un programa	
Atributos: nombre, codigo, duracion, valor, registro	

Nombre: Asignatura	
Descripción: Datos generales de una asignatura	
Atributos: nombre, codigo	

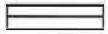
Nombre: Permisos	
Descripción: Identificar cada uno de los permisos que serán asignados dependiendo de su perfil a cada usuario	
Atributos: nombre, codigo, tipoacceso	

Nombre: Inscripción	
Descripción: Registra la información correspondiente a la inscripción de un candidato	
Atributos: cod_programa, cedula_candidato	

Nombre: Penum	
Descripción: Almacena información correspondiente al los diferente Penum que tiene registrado cada programa	
Atributos: cod_programa, cod_asignatura, intensidad	

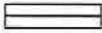
Nombre: Estudiante_programa	
Descripción: Almacena información correspondiente en que modulo se encuentra registrado el estudiante	
Atributos: cedula_estudiante, cod_programa, modulo	

Nombre: Asignatura_estudiante	
Descripción: almacena calificaciones de cada uno de los estudiantes que cursan una asignatura	
Atributos: ced_estudiante, cod_asignatura, parcial, sparcial, tparcial	

Nombre: Inscripción Candidato	
Descripción: durante este proceso se oficializa la inscripción de un candidato	

Nombre: Matricular estudiante	
Descripción: durante este proceso se oficializa la matricula académica de un estudiante	

Nombre: Registro de notas	
Descripción: Registrar las notas de cada estudiante matriculado	

Nombre: Reportes	
Descripción: generar reportes sobre diferentes criterios en la gestión académica, los cuales tienen como objetivo obtener resultados fundamentales para la toma de decisiones.	

Nombre: Consulta de notas	
Descripción: informar a los estudiantes cada una de las calificaciones obtenidas en las asignaturas en las cuales se encuentra matriculado.	

Nombre: Ayuda	
Descripción: muestra una ayuda la cual sirve de guía para la utilización de la aplicación.	

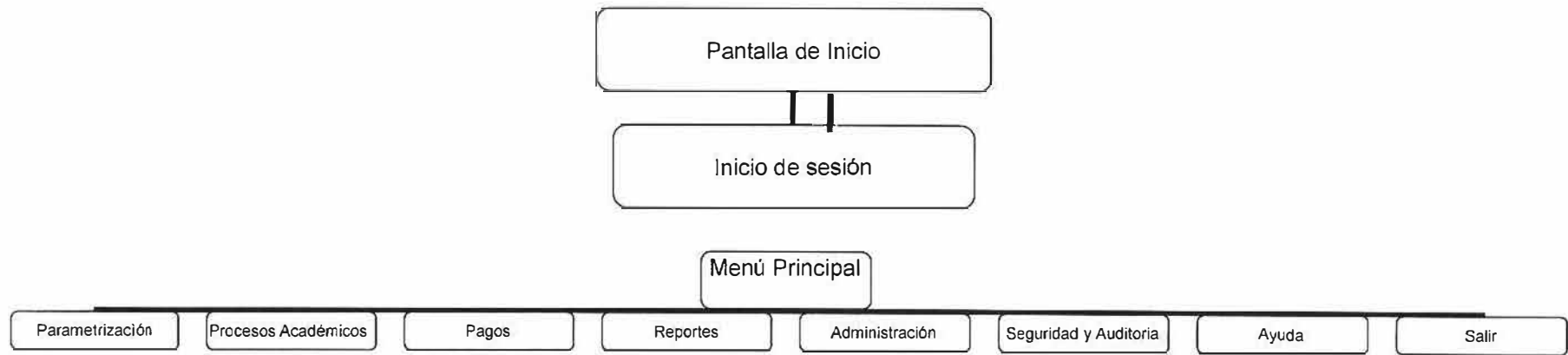
Nombre: Acerca de	
Descripción: muestra información del sistema, entre la que se puede destacar: fecha, empresa desarrolladora, número de licencia, datos de autor.	

Nombre: Salir del sistema	
Descripción: permite salir de la aplicación	

Nombre: Cerrar sesión	
Descripción: permite cerrar una sesión iniciada previamente por el usuario.	

11. DISEÑO DEL SISTEMA

11.1 ESTRUCTURA FUNCIONAL



11.2 DESCRIPCIÓN DE LOS COMPONENTES DE LA ESTRUCTURA

INFORMACIÓN ACADÉMICA

En este modulo de la aplicación se realizan las diferentes operaciones de registro de información, necesaria para llevar los procesos de gestión académica.

- **Candidato:** se registra los datos personales y generales de cada persona interesada en ingresar a los programas académicos que ofrece CODETEC.
- **Estudiante:** se registra la información completa de las personas que ya se encuentran oficialmente matriculadas en alguno de los programas académicos que ofrece CODETE.
- **Docente:** registro de cada uno de los docentes vinculados a la institución.
- **Programa:** registro de cada uno de los programas avalados por el ICFES.
- **Asignatura:** se registra cada una de las asignaturas correspondientes por programa y modulo.
- **Pensum:** se almacena la información correspondiente de cómo esta constituido cada de los programas académicos de la institución.

PROCESOS ACADÉMICOS

En este modulo de la aplicación se registran los principales procesos académicos que se llevan a cabo durante la gestión académica.

- **Inscripción:** se realiza la inscripción de un candidato en la institución, el proceso inicia con el registro del candidato y se oficializa con el registro del pago en el sistema.
- **Matricula:** se realiza la matricula académica de un estudiante en la institución, el proceso inicia con el registro del estudiante y se oficializa con el ingreso del pago en el sistema y la generación de listado de asignaturas a cursar.
- **Registro de Notas:** en esta opción se registra las notas obtenidas por cada uno de los estudiantes por asignatura.

REPORTES

Este modulo el sistema genera información a través de consultas específicas.

ADMINISTRACIÓN

En este modulo del sistema se realiza la gestión de usuarios que tienen acceso al sistema.

- **Usuarios:** se registra y consulta cada uno de los usuarios del sistema
- **Perfiles:** se registra cada uno de los perfiles de acceso al sistema.
- **Permisos:** se registra cada uno de los permisos que tienen asignado un determinado perfil.

AYUDA

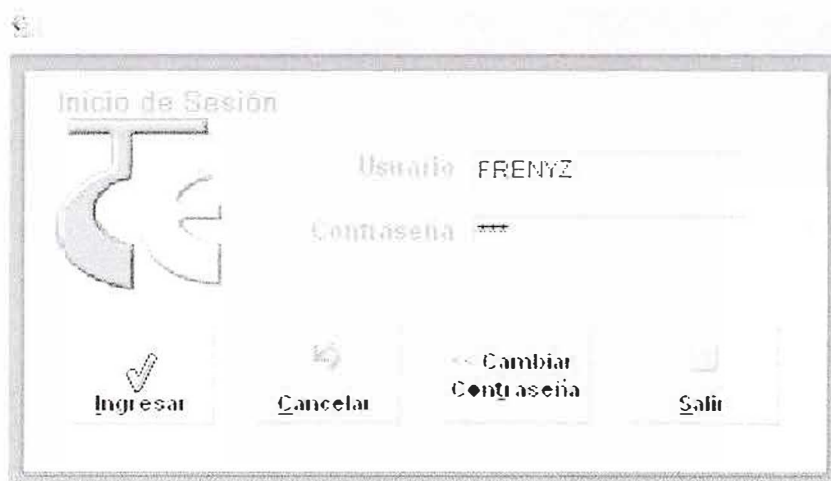
- **Ayuda:** muestra el contenido de la ayuda de la aplicación.
- **Acerca de:** Información de la aplicación.

SALIR

- **Salir del sistema:** esta opción permite salir completamente del sistema.
- **Cerrar sesión:** permite cerrar la sesión del usuario que esta utilizando el sistema.

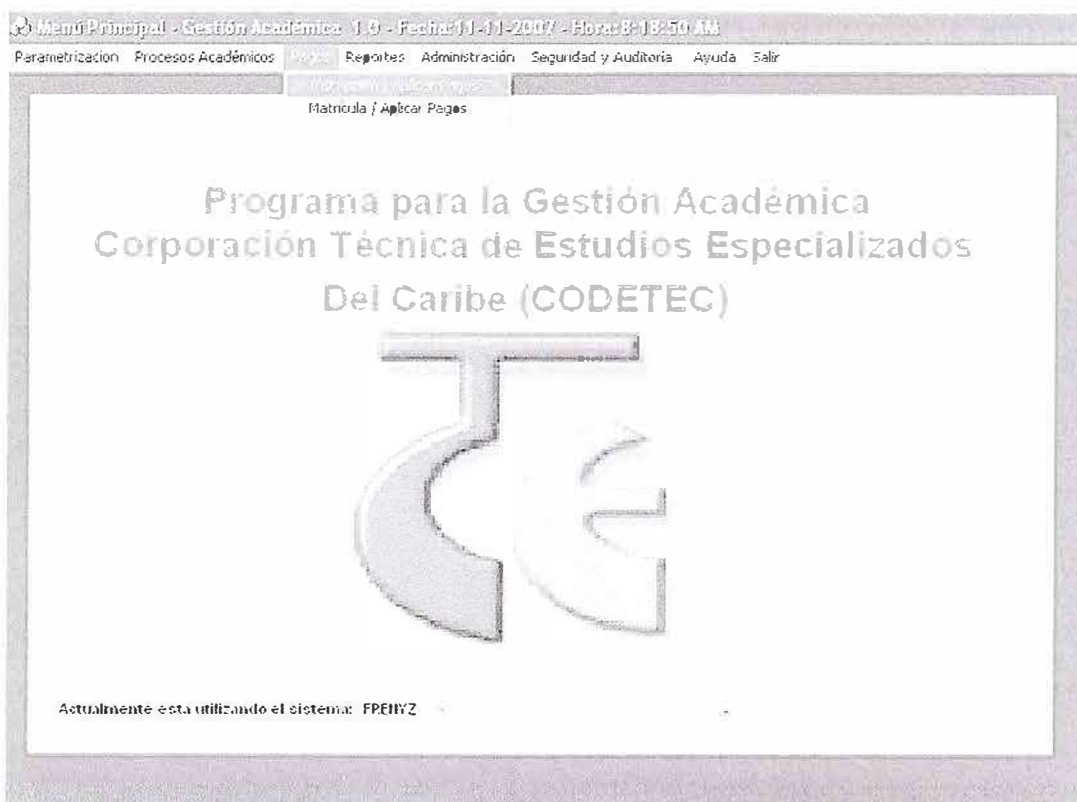
11.3 DISEÑO DE LAS INTERFACES – PROTOTIPO

Inicio de sesión



The login form is titled "Inicio de Sesión". It features a large logo on the left consisting of a stylized 'T' and 'E'. To the right of the logo are two input fields: "Usuario" with the text "FRENZY" and "Contraseña" with masked characters. Below the input fields are four buttons: "Ingresar" (with a checkmark icon), "Cancelar" (with an 'X' icon), "<< Cambiar Contraseña" (with a left arrow icon), and "Salir" (with a door icon).

Menú Principal



The main menu interface has a title bar that reads "Menú Principal - Gestión Académica 1.0 - Fecha: 11-11-2007 - Hora: 8:18:50 AM". Below the title bar is a navigation menu with the following items: "Parametrización", "Procesos Académicos", "Pagos", "Reportes", "Administración", "Seguridad y Auditoría", "Ayuda", and "Salir". The "Pagos" item is currently selected. Below the navigation menu is a sub-menu with "Matrícula" and "Aplicar Pagos". The main content area displays the text "Programa para la Gestión Académica", "Corporación Técnica de Estudios Especializados", and "Del Caribe (CODETEC)", followed by the large logo. At the bottom, it states "Actualmente está utilizando el sistema: FRENZY".

Registro de Información de Candidato

The screenshot shows a window titled "Candidato - Gestión Académica 1.0". Inside, there is a form titled "Información del Candidato". The form contains the following fields:

- No. de Documento
- Especialización
- Nombre
- Apellido
- Celular
- Teléfono Fijo
- Sexo
- E-mail
- Domicilio
- Unidad académica administrativa de pertenencia
- Nombre
- Dirección
- Teléfono
- Fax

At the bottom of the form, there are five buttons: "Agregar", "Modificar", "Eliminar", "Buscar", and "Cerrar".

Registro de datos de un Programa Académico

The screenshot shows a window titled "Programa - Gestión Académica 1.0". Inside, there is a form titled "Datos del Programa Académico". The form contains the following fields:

- Código
- Nombre
- Duración
- Semestre (s)
- Registro ICFES
- Valor \$

At the bottom of the form, there are five buttons: "Agregar", "Modificar", "Eliminar", "Buscar", and "Cerrar".

Registro de Notas

Menú Principal - Gestión Académica 1.0 - Fecha: 12-11-2007 - Hora: 10:52:29 PM

Parámetros: [Procesos Académicos](#) [Pagos](#) [Reportes](#) [Administración](#) [Seguridad y Auditoría](#) [Ayuda](#) [Salir](#)

[Matricula](#) [Registro de Notas](#)

Registro de Notas - Gestión Académica 1.0

Registro de Notas

Entidad del documento: Semestre: Estado:

Nombre: Apellido:

Programa

Asignatura

Identificación	Nombre	Escuelas	Nota1	Nota2	Nota3	Ensayo

Actualmente está utilizando el sistema: WCASTILLO

Logs de transacciones

CONCLUSIÓN

La Corporación Técnica De Estudios Especializados Del Caribe (CODETEC), es una institución que esta dedicada a la formación de talento humano capacitado para crear, ampliar y difundir conocimientos a la comunidad del caribe en general. CODETEC tiene como meta buscar una excelencia académica en su estudiantado; cuenta con herramientas tecnológicas de alta calidad que facilitan el aprendizaje y aportan a la investigación.

A través de la investigación que se realizó, se llego a la conclusión de que se estaban presentando muchas inconvenientes en el manejo de la información correspondiente a la gestión académica de la institución; esto debido al crecimiento importante en su población estudiantil y en la cobertura que antes ofrecía, la demanda de estudiantes que se está presentando ha sobrepasado estadísticamente a los años anteriores; esto ha traído consigo congestión en los procesos por la gran cantidad de registros que ahora se tienen que manejar.

Es por esta razón que el objetivo principal de este proyecto es construir un sistema de información que permita optimizar los procesos y actividades de gestión académica que se llevan a cabo en la Corporación De Estudios Técnicos Especializados Del Caribe.

BIBLIOGRAFÍA

- Microsoft ® Encarta ® 2006. © 1993-2005 Microsoft Corporation.
- Fundamentos de Bases de datos. Presman. Mc Graw Hill.
- UML y Patrones. Mc Graw Hill.
- Metodología de la investigación. Cesar Augusto Bernal. Bogota. Prentice Hall. Bogotá, 2000.
- Metodología de la investigación. Roberto Hernández Sampieri. Mc Graw Hill. México, 1991.
- Análisis y diseños de sistemas. Kendall Keneth E. Prentice Hall. México, 1998.
- Ingeniería del Software. Roger Pressman. Mc Graw Hill. México, 1998.
- Ingeniería del Software. Fairley Richard. Mc Graw Hill. México, 1998.
- Análisis de sistemas. Daniel Alan. Paraninfo. Madrid, 1991.
- Sistemas de Información para la administración. Senn James. Iberoamericana. México, 1990.
- www.mysql.com
- [http://www.utem.cl/mecegestioninstitucional ht.htm#objetivos](http://www.utem.cl/mecegestioninstitucional_ht.htm#objetivos)
- <http://www.monografias.com/trabajos7/sisinf/sisinf.shtml>
- <http://www.monografias.com/trabajos27/sistemas-informacion/sistemas-informacion.shtml>



**DISEÑO, DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA DE
INFORMACIÓN, PARA EL MANEJO DE LOS PROCESOS ACADÉMICOS
EN LA CORPORACIÓN TÉCNICA DE ESTUDIOS ESPECIALIZADOS DEL
CARIBE (CODETEC)**

MANUAL DEL SISTEMA

JOSÉ G. DE LA CRUZ

NARCISO CANTILLO

WILLIAM CASTILLO

FREDY RENY

BARRANQUILLA

UNIVERSIDAD SIMÓN BOLÍVAR

INGENIERÍA DE SISTEMAS

XI SEMESTRE – JORNADA NOCTURNA

2007

TABLA DE CONTENIDO

- 1. OBJETIVOS**
 - 1.1 OBJETIVO GENERAL**
 - 1.2 OBJETIVOS ESPECÍFICOS**
 - 2. ESPECIFICACIONES TÉCNICAS**
 - 3. DEFINICIÓN DE LA BASE DE DATOS DEL SISTEMA**
 - 4. LISTADO Y DESCRIPCIÓN DE LAS TABLAS DE LA BASE DE DATOS**
 - 5. SCRIPTS DE LA BASE DE DATOS**
 - 6. FORMULARIOS**
 - 7. MÓDULOS**
 - 8. DIAGRAMA DE FLUJO DE DATOS**
 - 9. CODIGO FUENTE APLICACIÓN**
-

1. OBJETIVOS

1.1 OBJETIVO GENERAL

Proporcionar al usuario conceptos claros y concisos acerca del manejo del software Gestión Académica 1.0 (Sistema De Información, Para El Manejo De Los Procesos Académicos En La Corporación Técnica De Estudios Especializados Del Caribe (Codetec)).

1.2 OBJETIVOS ESPECIFICOS

- Dar a conocer el funcionamiento interno del sistema de información.
- Proporcionar una descripción de los procedimientos del sistema.
- Servir de soporte en caso de un futuro mantenimiento de la aplicación.

2. ESPECIFICACIONES TÉCNICAS

Las herramientas que soportan la información son: Motor de base de datos MySQL y lenguaje de programación Visual Basic 6.0

Requerimientos mínimos

a. Servidor

- **Software**

Windows 98 ó superior. (Windows Vista no recomendable)

MySQL Servers and Clients 4

MySQL Front 2.5

MySQL ODBC 3.51 Driver

- **Hardware**

Procesador: Pentium III ó superior

Memoria Principal: 256 MB Hacia arriba

Disco duro: 20 GB hacia arriba para programa y datos.



Pantalla: Súper VGA. Debe estar configurada con 800*600 píxeles. De otra manera aunque el programa opera correctamente su presentación y control visual será deficiente.

b. Cliente

c. Software

Windows 98 ó superior. (Windows Vista no recomendable)

d. Hardware

Procesador: Pentium II ó superior

Memoria Principal: 128 MB Hacia arriba

Disco duro: 10 GB hacia arriba para programa y datos.

Pantalla: Súper VGA. Debe estar configurada con 800*600 píxeles. De otra manera aunque el programa opera correctamente su presentación y control visual será deficiente.

3. DEFINICIÓN DE LA BASE DE DATOS DEL SISTEMA

El motor de base de datos usado para el manejo de la información de la aplicación y soporte del sistema es **MySql**.

MySQL es un sistema de gestión de base de datos, multihilo y multiusuario. MySQL AB desarrolla MySQL como software libre en un esquema de licenciamiento dual. Por un lado lo ofrece bajo la GNU GPL, pero, empresas que quieran incorporarlo en productos privativos pueden comprar a la empresa una licencia que les permita ese uso. Está desarrollado en su mayor parte en ANSI C.

Al contrario de proyectos como el Apache, donde el software es desarrollado por una comunidad pública, y el copyright del código está en poder del autor individual, MySQL está poseído y patrocinado por una empresa privada, que posee el copyright de la mayor parte del código. Esto es lo que posibilita el esquema de licenciamiento anteriormente mencionado.

SQL (Lenguaje de Consulta Estructurado) fue comercializado por primera vez en 1981 por IBM, el cual fue presentado a ANSI y desde ese entonces ha sido considerado como un estándar para las bases de datos

relacionales. Desde 1986, el estándar SQL ha aparecido en diferentes versiones como por ejemplo: SQL:92, SQL:99, SQL:2003. MySQL es una idea originaria de la empresa opensource MySQL AB establecida inicialmente en Suecia en 1995 y cuyos fundadores son David Axmark, Allan Larsson, y Michael "Monty" Widenius. El objetivo que persigue esta empresa consiste en que MySQL cumpla el estándar SQL, pero sin sacrificar velocidad, fiabilidad o usabilidad.

Plataformas

MySQL funciona sobre múltiples plataformas, incluyendo AIX, BSD, FreeBSD, HP-UX, GNU/Linux, Mac OS X, NetBSD, Novell Netware, OpenBSD, OS/2 Warp, QNX, SGI IRIX, Solaris, SunOS, SCO OpenServer, SCO UnixWare, Tru64, Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows Vista y otras versiones de Windows.

Características de la versión 5.0.22

- Un amplio subconjunto de ANSI SQL 99, y varias extensiones.
 - Soporte a multiplataforma
 - Procedimientos almacenados
 - Triggers
 - Cursors
 - Vistas actualizables
-



- Soporte a VARCHAR
 - INFORMATION_SCHEMA
 - Modo Strict
 - Soporte X/Open XA de transacciones distribuidas; transacción en dos fases como parte de esto, utilizando el motor InnoDB de Oracle
 - Motores de almacenamiento independientes (MyISAM para lecturas rápidas, InnoDB para transacciones e integridad referencial)
 - Transacciones con los motores de almacenamiento InnoDB, BDB Y Cluster; puntos de recuperación(savepoints) con InnoDB
 - Soporte para SSL
 - Query caching
 - Sub-SELECTs (o SELECTs anidados)
 - Replication with one master per slave, many slaves per master, no automatic support for multiple masters per slave.
 - indexing y buscando campos de texto completos usando el motor de almacenamiento MyISAM
 - Embedded database library
 - Soporte completo para Unicode
-

- Conforme a las reglas ACID usando los motores InnoDB, BDB y Cluster
- Shared-nothing clustering through MySQL Cluster

Características distintivas

Las siguientes características son implementadas únicamente por MySQL:

- Múltiples motores de almacenamiento (MyISAM, Merge, InnoDB, BDB, Memory/heap, MySQL Cluster, Federated, Archive, CSV, Blackhole y Example en 5.x), permitiendo al usuario escoger la que sea más adecuada para cada tabla de la base de datos.
- Agrupación de transacciones, reuniendo múltiples transacciones de varias conexiones para incrementar el número de transacciones por segundo.

Estado actual

La serie en desarrollo de MySQL Server actualmente, es la 5.1 al cual se añaden nuevas características en relación a la serie 5.0. La serie de producción actual de MySQL es 5.0, cuya penúltima versión estable es la 5.0.26 lanzada en octubre de 2006. Actualmente, se puede descargar la serie 5.0.27. La serie de producción anterior fue la 4.1, cuya versión estable es 4.1.7 lanzada en octubre de 2004. A estas versiones de producción sólo se arreglan problemas, es decir, ya no se añaden nuevas características. Y a las versiones anteriores solamente se les corrigen bugs críticos.



4. LISTADO Y DESCRIPCIÓN DE LAS TABLAS DE LA BASE DE DATOS

* Llave Primaria

Candidato

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cedula *	numérico	20	Cédula del candidato	S
nombres	texto	50	Nombre del candidato	S
apellidos	texto	50	Apellidos del candidato	S
direccion	texto	50	Dirección del candidato	S
telefono	numérico	20	Teléfono del candidato	S
email	texto	50	email del candidato	N
Sexo	texto	1	Sexo del candidato	S
organización	texto	50	Nombre de la Organización ó institución de procedencia	N



CODETE

Gestión Académica 1.0

telefono_organizacion	numérico	20	Teléfono de la organización	N
direccion_organizacion	texto	50	Dirección de la organización	N

Docente

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cedula *	numérico	20	Cédula del docente	S
nombres	texto	50	Nombre del docente	S
apellidos	texto	50	Apellido del docente	S
direccion	texto	50	Dirección del docente	S
Telefono	numérico	20	Teléfono del docente	S
email	texto	50	email del docente	N
Sexo	texto	1	Sexo del docente	S
Info_profesional	texto	100	Referencia profesional del docente	S



Gestión Académica 1.0

Estudiante

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cedula *	numérico	20	Cédula del estudiante	S
nombres	texto	50	Nombre del estudiante	S
apellidos	texto	50	Apellido del estudiante	S
direccion	texto	50	Dirección del estudiante	S
telefono	numérico	20	Teléfono del estudiante	S
email	texto	50	email del estudiante	N
Sexo	texto	1	Sexo del estudiante	S
estudios	texto	100	Referencia de estudios realizados por el estudiante	S
fechanacimiento	fecha		fecha de nacimiento del estudiante	S
estadocivil	texto	15	estado civil del estudiante	S



Gestión Académica 1.0

Usuario

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cedula *	numérico	20	Cédula del usuario	S
nombres	texto	50	Nombre del usuario	S
apellidos	texto	50	Apellido del usuario	S
cargo	texto	50	Dirección del usuario	S
contraseña	texto	20	Teléfono del usuario	S
nusuario *	texto	50	Nombre del usuario en el sistema	S

Perfil

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
codigo *	numérico	20	Código del perfil	S
rango	texto	5	Nombre del perfil	S



Permiso

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
codigo *	numérico	20	Código del permiso	S
nombre	texto	5	Nombre del permiso	S
tipoacceso	texto	20	tipo de acceso que tendrá el permiso ya sea total o con ciertas restricciones	S

Matricula

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
codigo *	numérico	20	Código de la matricula	S
fecha	fecha/hora		Fecha de la matricula	S



Gestión Académica 1.0

hora	fecha/hora		Hora de la matricula	S
cedula	numérico	20	Cedula de estudiante	S

Programa

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
Nombre	texto	50	Nombre del programa	S
codigo *	numérico	20	Código del programa	S
Duración	numérico	5	Duración del programa	S
Registro	texto	50	registro lcfes del programa	S
Valor	numérico	20	valor del programa	S



Gestión Académica 1.0

Pensum

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cod_programa *	numérico	20	Código del programa	S
Cod_asignatura	numérico	20	Código de la asignatura	S
Intensidad	numérico	20	Intensidad del pensum	S

Asignatura

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
nombre	texto	50	Nombre de la asignatura	S
codigo *	numérico	20	Código de la asignatura	S



Docente_Programa

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cedula *	numérico	20	Cédula del docente	S
cod_programa *	numérico	20	Código del programa	S

Docente_Asignatura

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cedula *	numérico	20	Cédula del docente	S
cod_asignatura *	numérico	20	Código de la asignatura	S



Gestión Académica 1.0

Inscripción

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cod_programa *	numérico	20	Código del programa	S
cedula_candidato *	numérico	20	Cédula del candidato	S

Estudiante_programa

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cod_programa *	numérico	20	Código del programa	S
cedula_estudiante *	numérico	20	Cédula del estudiante	S
Modulo	numérico	2	Modulo que cursa el estudiante	S



Gestión Académica 1.0

Asignatura_Estudiante

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cod_asignatura *	numérico	20	Código del programa	S
cedula_estudiante *	numérico	20	Cédula del estudiante	S
pparcial	numérico	2	Nota primer parcial	S
sparcial	numérico	2	Nota segundo parcial	
tparcial	numérico	2	Nota tercer parcial	

Perfil_Permito

Nombre del Campo	Tipo	Tamaño	Descripción	Obligatorio S/N
cod_permiso *	numérico	20	Código del permiso	S
Cod_perfil *	numérico	20	Cédula del perfil	S



5. SCRIPTS DE LA BASE DE DATOS

```
# MySQL-Front Dump 2.5

#

# Host: localhost  Database: codetec

# -----

# Server version 4.0.17-nt

#

# Table structure for table 'asignatura'

#

CREATE TABLE asignatura (
  asi_codigo varchar(20) NOT NULL default "",
  asi_nombre varchar(40) NOT NULL default "",
  asi_usuario varchar(30) NOT NULL default "",
  asi_semestre varchar(40) default NULL,
  PRIMARY KEY (asi_codigo)
) TYPE=MyISAM;

#

# Table structure for table 'asignatura_estudiante'

#

CREATE TABLE asignatura_estudiante (
  ases_codasignatura varchar(20) NOT NULL default "",
  ases_idestudiante varchar(20) NOT NULL default '0',
```



```
ases_nombre varchar(40) default NULL,  
ases_apellido varchar(40) default NULL,  
ases_notas decimal(3,0) default '0',  
ases_notas2 decimal(3,0) default '0',  
ases_notas3 decimal(3,0) default '0',  
ases_notafinal decimal(3,0) default '0',  
ases_usuario varchar(40) NOT NULL default "",  
ases_matricula varchar(20) NOT NULL default ""  
) TYPE=MyISAM;  
  
#  
# Table structure for table 'auditor'  
#  
CREATE TABLE auditor (  
    aud_consecutivo bigint(10) unsigned NOT NULL auto_increment,  
    aud_usuario varchar(40) default NULL,  
    aud_operacion varchar(40) default NULL,  
    aud_fecha date default NULL,  
    aud_hora time default NULL,  
    aud_modulo varchar(40) default NULL,  
    aud_identificador varchar(40) default NULL,  
    aud_identificador2 varchar(40) default NULL,  
    aud_identificador3 varchar(40) default NULL,  
    PRIMARY KEY (aud_consecutivo)
```



```
) TYPE=MyISAM;

#

# Table structure for table 'candidato'

#

CREATE TABLE candidato (
  can_cedula varchar(20) NOT NULL default '0',
  can_tipo varchar(30) NOT NULL default "",
  can_nombres varchar(50) NOT NULL default "",
  can_apellidos varchar(50) NOT NULL default "",
  can_direccion varchar(40) NOT NULL default "",
  can_telefijo varchar(20) default NULL,
  can_telemovil varchar(30) default NULL,
  can_email varchar(40) default NULL,
  can_sexo char(1) NOT NULL default "",
  can_organizacion varchar(50) default NULL,
  can_telorganizacion varchar(15) default NULL,
  can_dirorganizacion varchar(40) default NULL,
  can_egresado char(2) default NULL,
  can_usuario varchar(30) NOT NULL default "",
  can_semestre varchar(10) default 'Primero',
  PRIMARY KEY (can_cedula,can_tipo)
) TYPE=MyISAM;
```



```
#  
  
# Table structure for table 'codigos'  
  
#  
CREATE TABLE codigos (  
    codigo_ID varchar(20) default '0'  
)  
TYPE=MyISAM;  
  
#  
  
# Table structure for table 'consulta'  
  
#  
CREATE TABLE consulta (  
    con_temporal varchar(10) default NULL  
)  
TYPE=MyISAM;  
  
#  
  
# Table structure for table 'docente'  
  
#  
CREATE TABLE docente (  
    doc_identi bigint(20) unsigned NOT NULL default '0',  
    doc_tipoid varchar(40) NOT NULL default 'Cedula',  
    doc_nombres varchar(40) NOT NULL default "",  
    doc_apellidos varchar(40) NOT NULL default "",  
    doc_direccion varchar(40) NOT NULL default "",  
    doc_telefono varchar(40) default NULL,
```



```
doc_celular varchar(40) default NULL,
doc_email varchar(40) default NULL,
doc_sexo char(1) NOT NULL default "",
doc_infoprofesional longtext,
doc_usuario varchar(40) NOT NULL default "",
PRIMARY KEY (doc_identi,doc_tipoid)
) TYPE=MyISAM;

#
# Table structure for table 'docente_asignatura'
#
CREATE TABLE docente_asignatura (
  doas_ceddocente varchar(20) NOT NULL default "",
  doas_codasignatura varchar(20) NOT NULL default "",
  doas_usuario varchar(40) NOT NULL default ""
) TYPE=MyISAM;

#
# Table structure for table 'docente_programa'
#
CREATE TABLE docente_programa (
  dopr_ceddocente bigint(20) unsigned NOT NULL default '0',
  dopr_codprograma int(4) unsigned NOT NULL default '0',
  dopr_usuario varchar(40) NOT NULL default ""
) TYPE=MyISAM;
```




```
#  
# Table structure for table 'estudiante'  
#  
CREATE TABLE estudiante (  
    est_identi bigint(20) unsigned NOT NULL default '0',  
    est_tipodi varchar(20) NOT NULL default "",  
    est_nombres varchar(40) NOT NULL default "",  
    est_apellidos varchar(40) NOT NULL default "",  
    est_direccion varchar(40) NOT NULL default "",  
    est_telefijo varchar(40) default NULL,  
    est_telemovil varchar(40) default NULL,  
    est_email varchar(40) default NULL,  
    est_sexo char(1) NOT NULL default "",  
    est_fnacimiento date NOT NULL default '0000-00-00',  
    est_ecivil varchar(40) NOT NULL default "",  
    est_usuario varchar(40) NOT NULL default "",  
    est_estado varchar(20) default NULL,  
    PRIMARY KEY (est_identi,est_tipodi)  
) TYPE=MyISAM;
```



```
#  
  
# Table structure for table 'estudiante_programa'  
  
#  
CREATE TABLE estudiante_programa (  
    espro_idestudiante bigint(20) unsigned NOT NULL default '0',  
    espro_programa varchar(40) NOT NULL default "",  
    espro_semestre int(1) unsigned NOT NULL default '0',  
    espro_usuario varchar(40) NOT NULL default ""  
) TYPE=MyISAM;  
  
#  
  
# Table structure for table 'grupos'  
  
#  
CREATE TABLE grupos (  
    gru_codigo varchar(20) default NULL,  
    gru_inteligente varchar(30) default NULL,  
    gru_semestre varchar(20) default NULL,  
    gru_programa varchar(20) default NULL,  
    gru_asignatura varchar(20) default NULL,  
    gru_docente varchar(20) default NULL,  
    gru_grupo char(1) default NULL  
) TYPE=MyISAM;
```



#

Table structure for table 'inscripcion'

#

```
CREATE TABLE inscripcion (  
    ins_codprograma varchar(10) default '0',  
    ins_idcandidato varchar(20) default '0',  
    ins_usuario varchar(40) default NULL  
) TYPE=MyISAM;
```

#

Table structure for table 'mat_asignatura'

#

```
CREATE TABLE mat_asignatura (  
    Cod_mat varchar(20) default NULL,  
    Cod_m1 varchar(20) default NULL,  
    Cod_m2 varchar(20) default NULL,  
    Cod_m3 varchar(20) default NULL,  
    Cod_m4 varchar(20) default NULL,  
    Cod_m5 varchar(20) default NULL,  
    Cod_m6 varchar(20) default NULL,  
    Cod_m7 varchar(20) default NULL,  
    Cod_m8 varchar(20) default NULL,  
    Cod_m9 varchar(20) default NULL,
```



```
Cod_m10 varchar(20) default NULL
) TYPE=MyISAM;
#
# Table structure for table 'matricula'
#
CREATE TABLE matricula (
  mat_codigo bigint(20) unsigned NOT NULL default '0',
  mat_idestudiante bigint(20) NOT NULL default '0',
  mat_fecha varchar(20) default NULL,
  mat_hora varchar(20) default NULL,
  mat_usuario varchar(40) default NULL,
  mat_programa varchar(20) default NULL,
  mat_semestre varchar(20) default NULL,
  mat_inte varchar(20) default NULL,
  mat_volante varchar(20) default NULL,
  mat_grupo varchar(20) default NULL,
  mat_codgrupo varchar(20) default NULL,
  PRIMARY KEY (mat_codigo)
) TYPE=MyISAM;
```



```
#  
  
# Table structure for table 'pagos'  
  
#  
  
CREATE TABLE pagos (  
    pag_cedula varchar(20) default NULL,  
    pag_valor varchar(20) default NULL,  
    pag_volante varchar(20) default NULL  
) TYPE=MyISAM;  
  
#  
  
# Table structure for table 'pensum'  
  
#  
  
CREATE TABLE pensum (  
    pen_codprograma varchar(20) default NULL,  
    pen_codasignatura varchar(20) default NULL,  
    pen_usuario varchar(40) default NULL,  
    pen_semestre varchar(40) default NULL,  
    pen_inteligente varchar(20) default NULL  
) TYPE=MyISAM;  
  
#  
  
# Table structure for table 'perfil'  
  
#  
  
CREATE TABLE perfil (  

```



```
per_codigo varchar(4) NOT NULL default "",
per_nombre varchar(40) NOT NULL default "",
per_usuario varchar(40) NOT NULL default "",
PRIMARY KEY (per_codigo)
) TYPE=MyISAM;

#
# Table structure for table 'perfil_permiso'
#
CREATE TABLE perfil_permiso (
  cod_perfil varchar(4) default NULL,
  pcan char(1) default '0',
  pest char(1) default '0',
  pdoc char(1) default '0',
  pprog char(1) default '0',
  pasig char(1) default '0',
  ppen char(1) default '0',
  pasigna char(1) default '0',
  pmat char(1) default '0',
  pnotas char(1) default '0',
  ppmat char(1) default '0',
  ppins char(1) default '0',
  plistard char(1) default '0',
  prpen char(1) default '0',
```



```
pvol char(1) default '0',
prasi char(1) default '0',
ppcand char(1) default '0',
prest char(1) default '0',
pusu char(1) default '0',
plogs char(1) default '0'
) TYPE=MyISAM;

#
# Table structure for table 'programa'
#
CREATE TABLE programa (
  pro_codigo varchar(20) NOT NULL default "",
  pro_nombre varchar(40) default NULL,
  pro_duracion varchar(10) default NULL,
  pro_regicfes varchar(20) default NULL,
  pro_valor bigint(20) unsigned default '0',
  pro_usuario varchar(40) default NULL,
  pro_valins bigint(20) unsigned default NULL,
  PRIMARY KEY (pro_codigo)
) TYPE=MyISAM;
```



```
#  
  
# Table structure for table 'relaciones'  
  
#  
CREATE TABLE relaciones (  
    rel_origen varchar(40) default NULL,  
    rel_destino varchar(40) default NULL,  
    rel_llaveorigen varchar(40) default NULL,  
    rel_llavedestino varchar(40) default NULL  
) TYPE=MyISAM;  
  
#  
  
# Table structure for table 'tipo_asignacion'  
  
#  
CREATE TABLE tipo_asignacion (  
    tipo_asidep varchar(10) default NULL,  
    tipo_asiind varchar(20) default NULL,  
    tipo_programa varchar(10) default NULL,  
    tipo_tipo varchar(10) default NULL  
) TYPE=MyISAM;  
  
#  
  
# Table structure for table 'usuario'  
  
#  
CREATE TABLE usuario (  

```



```
usu_cedula varchar(20) NOT NULL default '0',
usu_nombre varchar(40) NOT NULL default "",
usu_apellidos varchar(40) NOT NULL default "",
usu_perfil varchar(40) NOT NULL default "",
usu_contrasena varchar(10) NOT NULL default "",
usu_usuario varchar(40) NOT NULL default "",
usu_nomusu varchar(30) default NULL,
PRIMARY KEY (usu_cedula,usu_contrasena)
) TYPE=MyISAM;
```

#

Table structure for table 'volantes'

#

```
CREATE TABLE volantes (
  vol_cedula varchar(20) default '0',
  vol_numero varchar(20) NOT NULL default '0',
  vol_tipo varchar(20) default NULL,
  vol_estado varchar(20) default NULL,
  vol_programa varchar(20) default NULL,
  vol_valor bigint(20) unsigned default NULL,
  vol_inte varchar(20) default NULL,
  vol_semestre varchar(20) default 'Primero',
  PRIMARY KEY (vol_numero)
) TYPE=MyISAM;
```



CODETE

Gestión Académica 1.0

6. FORMULARIOS

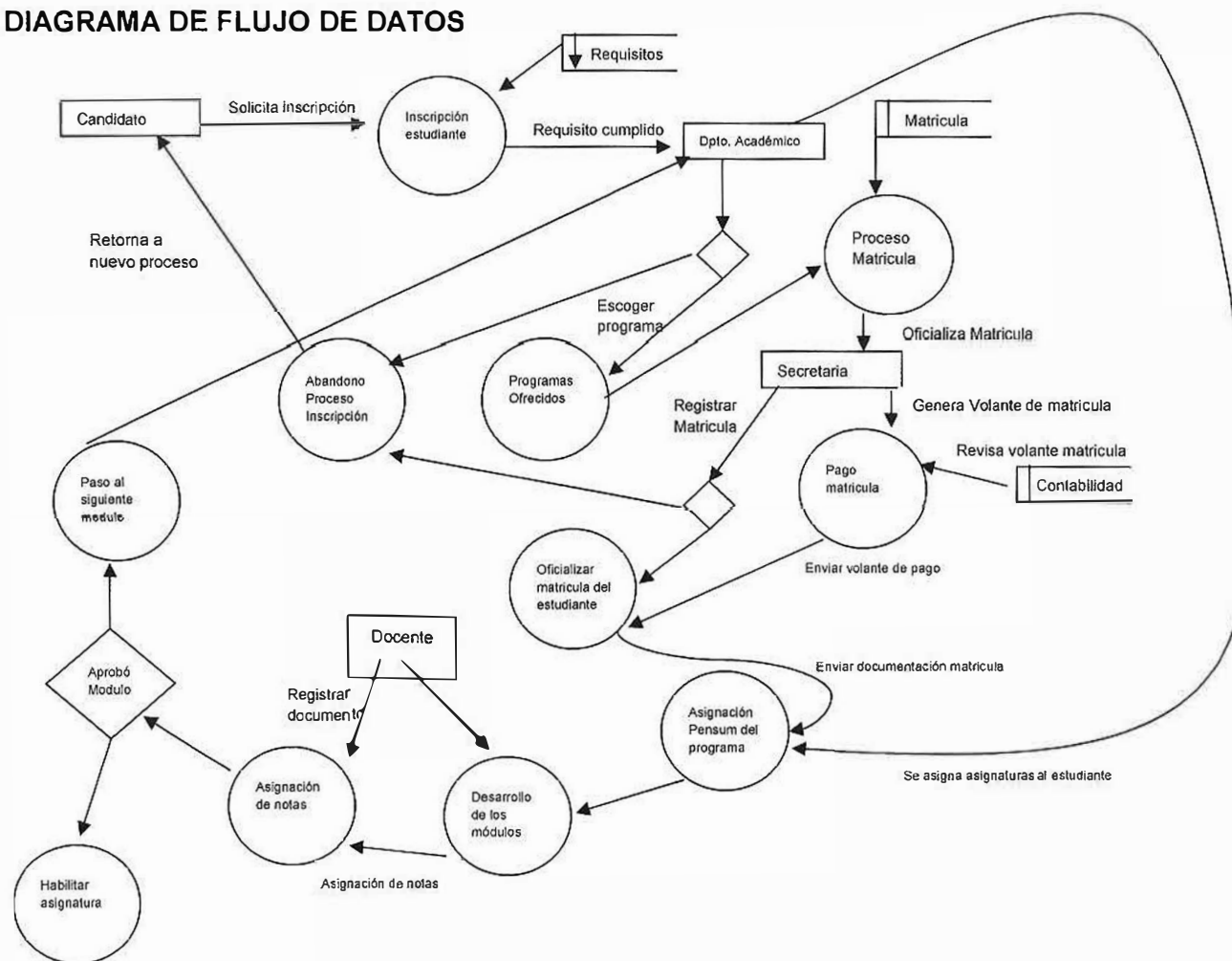
☐ frmacerca.frm	☐ frmperfil.frm
☐ frmabartar.frm	☐ frmpermiso.frm
☐ frmapplicar.frm	☐ frmprincipal.frm
☐ frmassignaciones.frm	☐ frmprograma.frm
☐ frmassignatura.frm	☐ frmreg_notas.frm
☐ frmbackup.frm	☐ frmusu.frm
☐ frmcambio.frm	☐ frmusuario.frm
☐ frmccandidato.frm	
☐ frmcconstig.frm	
☐ frmcconpen.frm	
☐ frmconsultas.frm	
☐ frmconvol.frm	
☐ frmddoc_asig.frm	
☐ frmddocente.frm	
☐ frmestudiante.frm	
☐ frmexport.frm	
☐ frmgenerador.frm	
☐ frmgrupos.frm	
☐ frmimport.frm	
☐ frminicio.frm	
☐ frminscripcion.frm	
☐ frminscritos.frm	
☐ frmlistadep.frm	
☐ frmlogin.frm	
☐ frmlogs.frm	
☐ frmmatricula.frm	
☐ frmmatriculados.frm	
☐ frmminiprograma.frm	
☐ frmnoinscritos.frm	
☐ frmnot.frm	
☐ frmnotdoc.frm	
☐ frmofcmatricula.frm	
☐ frmPenResul.frm	
☐ frmpensum.frm	



7. MÓDULOS

- Parametrización
 - Procesos Académicos
 - Reportes
 - Administración
 - Seguridad y Auditoria
 - Ayuda
-

8. DIAGRAMA DE FLUJO DE DATOS



9. CODIGO FUENTE DE LA APLICACIÓN

Frmapartar

```
Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As Long) As Long
```

```
Const SW_NORMAL = 1
```

```
Private Sub cmdayuda_Click()
```

```
ShellExecute frmprincipal.hwnd, "Open", "parametrizacion.chm", &O0, &O0, SW_NORMAL
```

```
End Sub
```

```
Private Sub cmdasignar_Click()
```

```
Do
```

```
    If txtminimo.Text = "" Then
```

```
        txtminimo.Text = InputBox("debe asignar un cupo minimo para comenzar el curso")
```

```
    End If
```

```
    If txtmaximo.Text = "" Then
```

```
        txtmaximo.Text = InputBox("debe asignar un cupo maximo para comenzar el curso")
```

```
    End If
```

```
    Loop Until Not (txtminimo.Text = "") And Not (txtmaximo.Text = "")
```

```
    'txtcodigo2.Text = txtcodigo.Text
```

```
    txtcod2.Text = txtcod.Text
```

```
    'txtcedula2.Text = txtcedula.Text
```

```
    'txtsemestre2.Text = txtsemestre.Text
```

```
    txtasi2.Text = txtasi.Text
```

```
    'txtinte.Text = txtinteligente.Text
```

```
    txtgrupo.Text = Combo2.Text
```

```
    txtgsemestre.Text = txtsemestre.Text
```

```
    txtgcodigo.Text = txtcod.Text
```

```
    txtginteligente.Text = txtinteligente.Text
```

```
    txtgsemestre.Text = txtsemestre.Text
```

```
    txtgcodigo.Text = txtcodigo.Text
```

```
    txtgdocente.Text = txtcedula.Text
```

```
    txtggrupo.Text = txtgrupo.Text
```

```
    ado_auditor.Recordset.AddNew
```

```
    txtiden2.Text = txtasi2.Text
```

```
    txtidenti.Text = txtcod2.Text
```

```
    txtident3.Text = txtginteligente.Text
```

```
    txtoperacion.Text = "Asignacion Grupos"
```

```
    txtusu.Text = frmprincipal.Label4
```

```
    txtfecha.Text = Format(Date, "dd-mm-yyyy")
```

```
    txthora.Text = Format(Now, "h:nn:ss")
```

```
    txtmodulo.Text = "Grupos"
```

```
    ado_auditor.Recordset.Update
```

```
ado_actualizacion.Recordset.Update
```

```
ado_actualizacion.Refresh
```

```
'ado_grupos.Recordset.Update
```

```
MsgBox "Registro guardado", vbInformation
```

```
'cmdasignar.Enabled = False
```

```
'cmdcancelar.Enabled = False
```

```
'cmdasignar.Enabled = False
```

```
'cmdcancelar.Enabled = False
```

```
'bloquear
```

```
cmdnuevo.Enabled = False
```

```
'ado_grupos.Recordset.AddNew
```

```
ado_actualizacion.Recordset.AddNew
```

```
cmdterminar.Enabled = True
```

```
End Sub
```

```
Private Sub cmdcancelar_Click()
```

```
ado_grupos.Recordset.CancelUpdate
```

```
txtminimo.Text = ""
```

```
txtmaximo.Text = ""
```

```
txtinteligente.Text = ""
```

```
cmdasignar.Enabled = False
```

```
cmdnuevo.Enabled = True
```

```
bloquear  
End Sub
```

```
Private Sub cmdcerrar_Click()  
X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")  
If X = vbYes Then  
    Unload Me  
End If  
End Sub
```

```
Private Sub cmdnuevo_Click()  
desbloquear  
MsgBox "Se esta iniciando el proceso de asignación de grupos", vbInformation, "Asignación de Grupos - Gestión Académica 1.0"  
If txtnumero.Text = "" Then  
    ado_consulta.Refresh  
    txtcod.Text = 100  
Else  
    txtcod.Text = txtnumero.Text  
    ado_consulta.Refresh  
End If  
MsgBox "Escoger el semestre del grupo a apartar", vbInformation, "Escoger Semestre"  
Combo1.SetFocus  
'ado_grupos.Recordset.AddNew  
ado_actualizacion.Recordset.AddNew  
cmdasignar.Enabled = True  
cmdcancelar.Enabled = True  
cmdnuevo.Enabled = False  
cmdterminar.Enabled = True  
End Sub
```

```
Private Sub cmdterminar_Click()  
ado_actualizacion.Recordset.CancelUpdate  
cmdterminar.Enabled = False  
cmdasignar.Enabled = False  
cmdnuevo.Enabled = True  
End Sub
```

```
Private Sub Combo1_Click()  
MsgBox "Asignar Grupo correspondiente", vbInformation, "Escoger Grupo"  
Combo2.SetFocus  
End Sub
```

```
Private Sub Combo1_KeyPress(KeyAscii As Integer)  
If KeyAscii = 13 Then  
    MsgBox "Asignar Grupo correspondiente", vbInformation, "Escoger Grupo"  
    Combo2.SetFocus  
End If  
End Sub  
Private Sub Combo2_Click()  
MsgBox "Asignar Programa correspondiente", vbInformation, "Escoger Programa"  
Command1.SetFocus  
End Sub
```

```
Private Sub Combo2_KeyPress(KeyAscii As Integer)  
If KeyAscii = 13 Then  
    MsgBox "Asignar Programa correspondiente", vbInformation, "Escoger Programa"  
    Command1.SetFocus  
End If  
End Sub
```

```
Private Sub Command1_Click()  
frmminiprograma.cmdasignar.Caption = "Asignar Apertura"  
txtsemestre.Text = Combo1.Text  
frmminiprograma.Show vbModal  
End Sub
```

```
Private Sub DataGrid1_Click()  
txtasi.Text = DataGrid1.Columns("Codigo").Text  
Label5.Caption = DataGrid1.Columns("Nombre").Text
```

End Sub

```
Private Sub DataGrid2_RowColChange(LastRow As Variant, ByVal LastCol As Integer)
txtcedula.Text = DataGrid2.Columns("Documento").Text
Label7.Caption = DataGrid2.Columns("Nombres").Text
Label11.Caption = DataGrid2.Columns("Apellidos").Text
MsgBox "Escoger Asignar correspondiente". vbInformation, "Escoger Asignatura"
End Sub
```

```
Sub bloquear()
Combo1.Enabled = False
Combo2.Enabled = False
Command1.Enabled = False
txtminimo.Enabled = False
txtmaximo.Enabled = False
DataGrid1.Enabled = False
DataGrid2.Enabled = False
End Sub
```

```
Sub desbloquear()
Combo1.Enabled = True
Combo2.Enabled = True
Command1.Enabled = True
txtminimo.Enabled = True
txtmaximo.Enabled = True
DataGrid1.Enabled = True
DataGrid2.Enabled = True
```

End Sub

```
Private Sub Form_Load()
bloquear
End Sub
```

Frmaplicar

```
Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "pagos.chm", &O0, &O0, SW_NORMAL
End Sub
```

```
Private Sub cmdaplicar_Click()
Dim s As String
Dim bus As String
ado_pagos.Recordset.AddNew
bus = txtvolante.Text
If bus <> vbNullString Then
ado_bvolante.Recordset.MoveFirst
ado_bvolante.Recordset.Find ("vol_inte=" & bus & "")
If ado_bvolante.Recordset.EOF Then
MsgBox "Verifique numero de volante", vbInformation + vbOKOnly, "Código Incorrecto"
ado_bvolante.Recordset.MoveFirst
Else
If Frame1.Caption = "Aplicar Pago Matricula" Then
txtpagvol.Text = txtvolante.Text
txtpagval.Text = txtvalor.Text
txtced.Text = txtcedula.Text
txtestado.Text = txtest.Text
s = "select can_nombres, can_apellidos from candidato where " & txtced.Text & " = can_cedula"
ado_nombre.RecordSource = s
ado_nombre.Refresh
Inombre.Caption = txtnombres.Text & " " & txtapellidos.Text
If txtest.Text = "Tramite" Then
txtest.Text = "Cancelado"
txtestado.Text = txtest.Text

ado_auditor.Recordset.AddNew
txtiden2.Text = txtvolante.Text
txtidenti.Text = txtcedula.Text
txtoperacion.Text = "Aplicar-Pago-Matricula"
```

```

txtusu.Text = frmprincipal.Label4
txtfecha.Text = Format(Date, "dd-mm-yyyy")
txthora.Text = Format(Now, "h:nn:ss")
txtmodulo.Text = "Aplicar Pago Matrícula"
ado_auditor.Recordset.Update

ado_bvolante.Recordset.Update
ado_pagos.Recordset.Update
MsgBox "El pago de Matrícula ha sido aplicado", vbApplicationModal + vbInformation, "Pago Aplicado"
txtvolante.SetFocus
txtvolante.Text = " "
Else
MsgBox "Ya el volante ha sido cancelado por favor revise el numero", vbCritical, "Numero de volante ya
aplicado"
Beep
End If
Else
txtpagvol.Text = txtvolante.Text
txtpagval.Text = txtvalor.Text
txtced.Text = txtcedula.Text
txtestado.Text = txttest.Text
s = "select can_nombres, can_apellidos from candidato where " + txtced.Text + " = can_cedula"
ado_nombre.RecordSource = s
ado_nombre.Refresh
Inombre.Caption = txtnombres.Text + " " + txtapellidos.Text
If txttest.Text = "Tramite" Then
txttest.Text = "Cancelado"
txtestado.Text = txttest.Text

ado_auditor.Recordset.AddNew
txtiden2.Text = txtvolante.Text
txtidenti.Text = txtcedula.Text
txtoperacion.Text = "Aplicar-Pago-Inscripción"
txtusu.Text = frmprincipal.Label4
txtfecha.Text = Format(Date, "dd-mm-yyyy")
txthora.Text = Format(Now, "h:nn:ss")
txtmodulo.Text = "Aplicar Pago Inscripción"
ado_auditor.Recordset.Update

ado_bvolante.Recordset.Update
ado_pagos.Recordset.Update
MsgBox "El pago ha sido aplicado por favor consulte el estado del estudiante y Legalize la Inscripción",
vbApplicationModal + vbInformation, "Pago Aplicado"
cmdlegalizar.Enabled = True
cmdlegalizar.SetFocus
Else
MsgBox "Ya el volante ha sido cancelado por favor revise el numero", vbCritical, "Numero de volante ya
aplicado"
Beep
End If
End If
End If
Else
MsgBox "Vuelva a intentar la busqueda - Código Nula", vbInformation + vbOKCancel, "Error de Busqueda"
End If
End Sub
Private Sub cmdlegalizar_Click()
ado_inscripcion.Recordset.AddNew
txtinscod.Text = txtcodProg.Text
txtinsced.Text = txtcedula.Text
txtusuario.Text = frmlogin.txtusuario.Text

ado_inscripcion.Recordset.Update
MsgBox "El proceso de inscripción ha sido legalizado", vbApplicationModal + vbInformation, "Inscripción Legalizada"
cmdlegalizar.Enabled = False
cmdsair.SetFocus
End Sub
Private Sub Cmdsair_Click()
ado_bvolante.Recordset.MoveFirst
Unload Me

```


End Sub

```
Private Sub Form_Load()  
cmdlegalizar.Enabled = False  
End Sub
```

```
Private Sub txtvolante_Change()  
Dim i As Integer  
txtvolante.Text = UCase(txtvolante.Text)  
i = Len(txtvolante.Text)  
txtvolante.SelStart = i  
End Sub
```

```
Private Sub txtvolante_KeyPress(KeyAscii As Integer)  
If KeyAscii = 13 Then  
cmdaplicar.SetFocus  
End If  
End Sub
```

Frmasignacion

```
Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation  
As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As  
Long) As Long  
Const SW_NORMAL = 1  
Private Sub cmdayuda_Click()  
ShellExecute frmprincipal.hwnd, "Open", "parametrizacion.chm", &O0, &O0, SW_NORMAL  
End Sub
```

```
Private Sub cmdcancelar_Click()  
X = MsgBox("¿Desea cancelar el proceso de asignación de asignaturas?", vbYesNo + vbInformation, "Cancelar")  
If X = vbYes Then  
ado_asignacion.Recordset.CancelUpdate  
cmdnuevo.Enabled = True  
cmdcancelar.Enabled = False  
cmdguardar.Enabled = False  
Frame1.Enabled = False  
txtmat1.Text = ""  
txtmat2.Text = ""  
cmdprograma.Enabled = False  
txtprograma.Enabled = False  
End If  
End Sub
```

```
Private Sub cmdguardar_Click()  
Do  
'este ciclo se hace para no dejar pasar el valor sin que este logicamente bien diligenciado  
If txtmat2.Text = "" Or txtmat1.Text = "" Then  
If txtmat1.Text = "" Then  
MsgBox "Asigne asignatura numero uno", vbOKOnly + vbInformation  
frmasignatura.cmdasignar.Visible = True  
frmasignatura.txtsw.Text = 1  
frmasignatura.Show vbModal  
End If  
If txtmat2.Text = "" Then  
MsgBox "Asigne asignatura numero dos", vbOKOnly + vbInformation  
frmasignatura.cmdasignar.Visible = True  
frmasignatura.txtsw.Text = 0  
frmasignatura.Show vbModal  
End If  
End If  
If txtmat1.Text = txtmat2.Text Then  
MsgBox "Debe asignar una dependencia diferente", vbCritical  
txtmat1.Text = ""  
txtmat2.Text = ""  
MsgBox "Asigne asignatura numero uno", vbOKOnly + vbInformation  
frmasignatura.cmdasignar.Visible = True  
frmasignatura.txtsw.Text = 1  
frmasignatura.Show vbModal
```

```

MsgBox "Asigne asignatura numero dos", vbOKOnly + vbInformation
frmasignatura.cmdasignar.Visible = True
frmasignatura.txtsw.Text = 0
frmasignatura.Show vbModal
End If

```

```

If optalt = True Then
Do
If txtprograma.Text = "" Then
MsgBox "Debe asignar un programa a la dependencia alterna", vbOKOnly + vbInformation
frmminiprograma.cmdasignar.Caption = "Asignacion dependencia"
frmminiprograma.Show vbModal
End If
Loop Until Not (txtprograma.Text = "")
End If

```

```

Loop Until Not (txtmat2.Text = "") And Not (txtmat1.Text = "") And Not (txtmat1.Text = txtmat2.Text)

```

```

'esta instruccion se utiliza para crear el registro en la tabla auditor
ado_auditor.Recordset.AddNew
txtiden2.Text = txtcod2
txtidenti.Text = txtcod1
txtident3.Text = txtprograma.Text
txtoperacion.Text = "Asignacion dependencia"
txtusu.Text = frmprincipal.Label4
txtfecha.Text = Format(Date, "dd-mm-yyyy")
txthora.Text = Format(Now, "h:nn:ss")
txtmodulo.Text = "Asignacion dependencia"
ado_auditor.Recordset.Update

```

```

ado_asignacion.Recordset.Update
MsgBox "La asignacion fue realizada con exito", vbInformation + vbOKOnly
txtmat1.Text = ""
txtmat2.Text = ""
txtprograma.Text = ""
cmdmateria1.Enabled = False
cmdmateria2.Enabled = False
cmdnuevo.Enabled = True
cmdcancelar.Enabled = False
cmdguardar.Enabled = False
Frame1.Enabled = False
cmdprograma.Enabled = False
txtprograma.Enabled = False

```

```

End Sub

```

```

Private Sub cmdmateria1_Click()
frmasignatura.cmdasignar.Visible = True
frmasignatura.txtsw.Text = 1
frmasignatura.cmdagregar.Visible = False
frmasignatura.cmdmodificar.Visible = False
frmasignatura.Cmdeliminar.Visible = False
frmasignatura.Show vbModal
If (txtmat1.Text <> "") Then
Frame1.Enabled = True
optco.Value = True
cmdmateria2.SetFocus
End If
End Sub

```

```

Private Sub cmdmateria2_Click()
frmasignatura.cmdasignar.Visible = True
frmasignatura.txtsw.Text = 0
frmasignatura.cmdagregar.Visible = False
frmasignatura.cmdmodificar.Visible = False
frmasignatura.Cmdeliminar.Visible = False
frmasignatura.Show vbModal
End Sub

```

```

Private Sub cmdnuevo_Click()
ado_asignacion.Recordset.AddNew
cmdmateria1.Enabled = True
cmdmateria2.Enabled = True

```

```

cmdguardar.Enabled = True
cmdcancelar.Enabled = True
cmdnuevo.Enabled = False
MsgBox "Seleccionar Asignatura Dependiente", vbInformation, "Asignatura Dependiente"
cmdmateria1.SetFocus
End Sub

```

```

Private Sub cmdprograma_Click()
frmminiprograma.cmdasignar.Caption = "Asignacion dependencia"
frmminiprograma.Show vbModal
End Sub

```

```

Private Sub Cmdsalir_Click()
X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
If X = vbYes Then
Unload Me
End If
End Sub

```

```

Private Sub Form_Load()
cmdguardar.Enabled = False
cmdcancelar.Enabled = False
End Sub

```

```

Private Sub optalt_Click()
If optalt.Value = True Then
txttipo.Text = "A"
MsgBox "Debe escoger el programa donde la asignatura es alterna y Seguidamente la asignatura independiente",
vbInformation, "Asignaturas Alternas"
cmdprograma.Enabled = True
txtprograma.Enabled = True
cmdprograma.SetFocus
End If
End Sub

```

```

Private Sub optco_Click()
If optco.Value = True Then
txttipo.Text = "C"
cmdprograma.Enabled = False
txtprograma.Enabled = False
End If
MsgBox "Seleccionar Asignatura Independiente", vbInformation, "Asignatura Independiente"
End Sub

```

```

Private Sub optpre_Click()
If optpre.Value = True Then
txttipo.Text = "P"
cmdprograma.Enabled = False
txtprograma.Enabled = False
End If
MsgBox "Seleccionar Asignatura Independiente", vbInformation, "Asignatura Independiente"
End Sub

```

Frmasignatura

```

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation
As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As
Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "parametrizacion.chm", &O0, &O0, SW_NORMAL
End Sub

```

```

Sub bloquear()
txtnombre.Enabled = False
txtcodigo.Enabled = False
txtsemestre.Enabled = False
End Sub
Sub desbloquear()
txtnombre.Enabled = True
txtsemestre.Enabled = True
End Sub

```

```

Private Sub cmdagregar_Click()
    If cmdagregar.Caption = "&Agregar" Then
        ado_asignatura.Refresh
        txtsw2.Text = 0
        cmdagregar.Caption = "&Guardar"
        cmdagregar.ToolTipText = "Guardar datos de la asignatura"
        cmdmodificar.Caption = "&Cancelar"
        cmdmodificar.ToolTipText = "Cancelar proceso"
        ado_asignatura.Recordset.AddNew
        desbloquear
        txtnombre.SetFocus
        ado_asignatura.Enabled = False
        cmdeliminar.Enabled = False
        cmdcerrar.Enabled = False
        cmdbuscar.Enabled = False
        If txtnumero.Text = "" Then
            txtcodigo.Text = 1000
        Else
            ado_numero.Refresh
            txtcodigo.Text = txtnumero.Text
        End If

    Else
        cmdagregar.Caption = "&Agregar"
        cmdmodificar.Caption = "&Modificar"

        Do
            If txtnombre.Text = "" Then
                X = InputBox("Debe escribir el nombre de la asignatura", "Nombre de la asignatura")
                txtnombre.Text = X
            End If
            If txtcodigo.Text = "" Then
                X = InputBox("Debe escribir el código de la asignatura", "Código de la asignatura")
                txtcodigo.Text = X
            End If
        Loop Until Not (txtnombre.Text = "") And Not (txtcodigo.Text = "")
        txtusuario.Text = frmlogin.txtusuario.Text

        ado_auditor.Recordset.AddNew
        If txtsw2.Text = 0 Then
            txtidenti.Text = txtcodigo.Text
            txtoperacion.Text = "Inserción registro"
            txtusu.Text = frmprincipal.Label4
            txtfecha.Text = Format(Date, "dd-mm-yyyy")
            txthora.Text = Format(Now, "h:nn:ss")
            txtmodulo.Text = "Asignatura"
        Else
            txtantes.Text = txtantes1.Text
            txtdespues.Text = txtcodigo.Text & ", " & txtnombre.Text & ", " & txtsemestre
            txtidenti.Text = txtcodigo.Text
            txtoperacion.Text = "Actualización registro"
            txtusu.Text = frmprincipal.Label4
            txtfecha.Text = Format(Date, "dd-mm-yyyy")
            txthora.Text = Format(Now, "h:nn:ss")
            txtmodulo.Text = "Asignatura"
        End If
        ado_auditor.Recordset.Update

        ado_asignatura.Recordset.Update
        blanco
        bloquear
        ado_asignatura.Enabled = True
        cmdeliminar.Enabled = True
        cmdcerrar.Enabled = True
        cmdbuscar.Enabled = True
    End If
End Sub

Private Sub cmdasignar_Click()
    Dim s As String
    If txtsw.Text = "1" Then
        frmasignacion.txtmat1.Text = txtnombre.Text
        frmasignacion.txtcod1.Text = txtcodigo.Text
        frmasignacion.cmdmateria2.Enabled = True
    End If
End Sub

```

```

MsgBox "Escoger Tipo de dependencia", vbInformation, "Dependencia"
Unload Me
End If
If txtsw.Text = "0" Then
    frmasignacion.txtmat2.Text = txtnombre.Text
    frmasignacion.txtcod2.Text = txtcodigo.Text
    frmasignacion.Frame1.Enabled = True
    frmasignacion.optco.Value = True
    MsgBox "Guardar la asignación de la dependencia", vbInformation, "Guardar"
    Unload Me
End If
If txtsw.Text = "2" Then
    frmlistadep.txtcodigo.Text = txtcodigo.Text
    frmlistadep.txtnombre.Text = txtnombre.Text
    frmlistadep.Label1.Caption = txtnombre.Text + " Semestre " + txtsemestre.Text
    s = "select t.tipo_tipo,a.asi_nombre,t.tipo_programa,asi_semestre from tipo_asignacion t,asignatura a where t.tipo_asidep=" + txtcodigo.Text + " and t.tipo_asiind = a.asi_codigo"
    frmlistadep.ado_listardep.RecordSource = s
    frmlistadep.ado_listardep.Refresh
    Unload Me
    frmlistadep.Show vbModal
End If
End Sub

Private Sub cmdbuscar_Click()
    Dim bus As String
    bus = InputBox("Digite el código de la asignatura", "Buscar Asignatura")
    If bus <> vbNullString Then
        ado_asignatura.Refresh
        ado_asignatura.Recordset.MoveFirst
        ado_asignatura.Recordset.Find ("asi_codigo=" & bus & "")

        If ado_asignatura.Recordset.EOF Then
            MsgBox "Verifique el código de la asignatura", vbInformation + vbOKOnly, "Código Incorrecto"
            ado_asignatura.Recordset.MoveFirst
            blanco
        End If
    Else
        MsgBox "Vuelva a intentar la búsqueda - Código Nula", vbInformation + vbOKCancel, "Error de Búsqueda"
        blanco
    End If
End Sub

Private Sub cmdeliminar_Click()
    Dim resp As String
    resp = MsgBox("¿Seguro que desea eliminar este registro?", vbYesNo + vbInformation, "Eliminar registro")
    If resp = vbYes Then
        ado_auditor.Recordset.AddNew
        txtidenti.Text = txtcodigo.Text
        ado_asignatura.Recordset.Delete
        ado_asignatura.Recordset.MovePrevious
        txtoperacion.Text = "Eliminación registro"
        txtusu.Text = frmprincipal.Label4
        txtfecha.Text = Format(Date, "dd-mm-yyyy")
        txthora.Text = Format(Now, "h:nn:ss")
        txtmodulo.Text = "Asignatura"
        ado_auditor.Recordset.Update

        End If
        ado_asignatura.Recordset.MoveFirst
        blanco
    End Sub

Private Sub cmdModificar_Click()
    If cmdmodificar.Caption = "&Modificar" Then
        'estas instrucciones dan como resultado el temporal del registro actuasi que despues se guardan en la tabla auditor
        txtantes1.Text = txtcodigo.Text & ", " & txtnombre.Text & ", " & txtsemestre
        txtsw2.Text = 1
        cmdmodificar.Caption = "&Cancelar"
        cmdagregar.Caption = "&Guardar"
        cmdagregar.ToolTipText = "Guardar datos de la asignatura"
        cmdmodificar.ToolTipText = "Cancelar proceso"
        desbloquear
        txtnombre.SetFocus
        ado_asignatura.Enabled = False

```

```

        cmdeliminar.Enabled = False
        cmdcerrar.Enabled = False
        cmdbuscar.Enabled = False
    Else
        ado_asignatura.Recordset.CancelUpdate
        blanco
        cmdmodificar.Caption = "&Modificar"
        cmdagregar.Caption = "&Agregar"
        cmdagregar.ToolTipText = "Agregar una nueva asignatura"
        cmdmodificar.ToolTipText = "Modificar una asignatura existente"
        ado_asignatura.Enabled = True
        cmdeliminar.Enabled = True
        cmdcerrar.Enabled = True
        cmdbuscar.Enabled = True
        bloquear
    End If
End Sub
Private Sub cmdcerrar_Click()
X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
If X = vbYes Then
    Unload Me
End If
End Sub

```

```

Private Sub Form_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
    SendKeys "{Tab}"
End If
End Sub

```

```

Private Sub Form_Load()
    blanco
    bloquear
End Sub

```

```

Private Sub txtcodigo_Change()
Dim i As Integer
txtcodigo.Text = UCase(txtcodigo.Text)
i = Len(txtcodigo.Text)
txtcodigo.SelStart = i
End Sub
Private Sub txtnombre_Change()
Dim i As Integer
txtnombre.Text = UCase(txtnombre.Text)
i = Len(txtnombre.Text)
txtnombre.SelStart = i
End Sub
Sub blanco()
    txtcodigo.Text = " "
    txtnombre.Text = " "
    txtsemestre.Text = " "
End Sub

```

Frmbackup

```

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "seguridad.chm", &O0, &O0, SW_NORMAL
End Sub

Private Sub cmdcerrar_Click()
X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
If X = vbYes Then
    Unload Me
End If
End Sub

Private Sub Command1_Click()

```

```

On Error Resume Next
Me.CommonDialog1.FileName = "backup"
Me.CommonDialog1.ShowSave
Me.Label1.Caption = Me.CommonDialog1.FileName
Dim l As Integer
l = Len(Me.Label1.Caption)
While l > 1 And Mid(Me.Label1.Caption, l, 1) <> "\"
    l = l - 1
Wend
Me.CommonDialog1.Tag = Mid(Me.Label1.Caption, 1, l)
Me.Label1.Caption = Mid(Me.Label1.Caption, 1, l) & "codetec" & Format(Date, "yyyymmdd") & "\"
End Sub

```

```

Private Sub Command2_Click()
On Error GoTo e
Dim a As Object
Dim fso As Object

Set fso = CreateObject("Scripting.FileSystemObject")
Set a = fso.GetFolder("C:\mysql\data\codetec")
MkDir Me.Label1.Caption

a.Copy (Me.Label1.Caption)
MsgBox "El proceso de Backup se realizo con exito".vbInformation, "Backup Exitoso"
SaveSetting "CODETEC", "HOME", "BACKUP", Me.CommonDialog1.Tag
Exit Sub
e:
If Err.Number = 75 Then
    MsgBox "El backup de este dia ya fue generado"
End If
End Sub

```

```

Private Sub Form_Load()
Me.Label1.Caption = GetSetting("CODETEC", "HOME", "BACKUP", "C:\mysql\data") & "codetec" & Format(Date, "yyyymmdd") & "\"
Me.CommonDialog1.Tag = GetSetting("CODETEC", "HOME", "BACKUP", "C:\mysql\data")
End Sub

```

Frmcambio

```

Private Sub cmdcambio_Click()

If txtusuario.Text = "" And txtcontrasena.Text = "" And txtcontrasena2.Text = "" Then
    MsgBox "Para poder llevar a cabo el proceso de Cambio de contraseña debe diligenciar toda la información Solicitada", vbCritical, "Error de Cambio de Contraseña"
txtusuario.SetFocus
Else
    c = "select usu_nomusu,usu_contrasena from usuario where" + txtusuario.Text + "= usu_nomusu and" + txtcontrasena.Text + "= usu_contrasena"
    c = "select usu_nomusu from usuario where" + txtusuario.Text + "= usu_nomusu"
    ado_clave.RecordSource = c
    ado_clave.Refresh

    If txtusuario.Text = txtusu.Text Then
        If txtcontrasena.Text = txtclave.Text And txtcontrasena.Text <> "" Then
            If txtcontrasena2.Text <> "" And txtcontrasena2.Text <> txtcontrasena.Text Then
                txtusuario.Text = txtusu.Text
                txtclave.Text = txtcontrasena2.Text
                ado_clave.Recordset.Update

                Dim Db As New ADODB.Connection
                Db.Open "Provider=MSDASQL.1;Persist Security Info=False;Extended Properties='DRIVER={MySQL ODBC 3.51 Driver};SERVER=localhost;DATABASE=Codetec ;UID=root ;OPTION=3'"
                Db.Execute "update usuario set usu_contrasena = password('" + txtcontrasena2.Text + "') where usu_nomusu =" & txtusu & ""
                Db.Close

                ado_clave.Recordset.MoveFirst
                MsgBox "El proceso de Cambio de Contraseña fue realizado Exitosamente", vbInformation, "Cambio de Contraseña Exitoso"
                txtusuario.Text = ""
                txtcontrasena.Text = ""
                txtcontrasena2.Text = ""
                cmdcambio.Enabled = False
            End If
        End If
    End If
End If

```

```

Else
    MsgBox "Nueva Contraseña de Usuario Erróneo, verifique la información", vbCritical, "Error Nueva Contraseña de usuario"
    txtcontrasena2.SetFocus
End If
End If
Else
    MsgBox "Nombre de Usuario ó Contraseña Erróneo, verifique la información", vbCritical, "Error nombre ó contraseña de usuario"
    txtusuario.SetFocus
End If
End If
End Sub

```

```

Private Sub cmdcerrar_Click()
X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
If X = vbYes Then
    Unload Me
End If
End Sub

```

```

Private Sub Form_Load()
If KeyAscii = 13 Then
    SendKeys "{Tab}"
End If
End Sub

```

```

Private Sub txtusuario_Change()
Dim i As Integer
txtusuario.Text = UCase(txtusuario.Text)
i = Len(txtusuario.Text)
txtusuario.SelStart = i
End Sub

```

Frmcandidato

```

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "parametrizacion.chm", &O0, &O0, SW_NORMAL
End Sub

```

```

Sub bloquear()
    txtnom.Enabled = False
    txtapellidos.Enabled = False
    txtcedula.Enabled = False
    txtfijo.Enabled = False
    txtcelular.Enabled = False
    txtdir.Enabled = False
    txtemail.Enabled = False
    txtnorganizacion.Enabled = False
    txtorgdir.Enabled = False
    txttelorg.Enabled = False
    txttipo.Enabled = False
    optmasculino.Enabled = False
    optfemenino.Enabled = False
    optegres.Enabled = False
    optegren.Enabled = False
End Sub

```

```

Sub desbloquear()
    txtnom.Enabled = True
    txtapellidos.Enabled = True
    txtcedula.Enabled = True
    txtfijo.Enabled = True
    txtcelular.Enabled = True
    txtdir.Enabled = True
    txtemail.Enabled = True
    txtnorganizacion.Enabled = True
    txtorgdir.Enabled = True
    txttelorg.Enabled = True
    txttipo.Enabled = True
    optmasculino.Enabled = True
    optfemenino.Enabled = True

```



```

    optegres.Enabled = True
    optegren.Enabled = True
End Sub

Sub blanco()
    txtnom.Text = " "
    txtapellidos.Text = " "
    txtcedula.Text = " "
    txtfijo.Text = " "
    txtcelular.Text = " "
    txtdir.Text = " "
    txtemail.Text = " "
    txtnorganizacion.Text = " "
    txtorgdir.Text = " "
    txttelorg.Text = " "
    txttipo.Text = " "
    optmasculino.Text = " "
    optfemenino.Text = " "
    optegres.Text = " "
    optegren.Text = " "
End Sub

Private Sub ado_candidatos_WillMove(ByVal adReason As ADODB.EventReasonEnum, adStatus As
ADODB.EventStatusEnum, ByVal pRecordset As ADODB.Recordset)
    If optsex.Text = "M" Then
        optmasculino.Value = True
    End If
    If optsex.Text = "F" Then
        optfemenino.Value = True
    End If
    If optegresado.Text = "S" Then
        optegres.Value = True
    End If
    If optegresado.Text = "N" Then
        optegren.Value = True
    End If
End Sub

Private Sub cmdagregar_Click()

    If cmdagregar.Caption = "&Agregar" Then
        txtsw.Text = 0
        ado_candidatos.Refresh
        cmdagregar.Caption = "&Guardar"
        cmdagregar.ToolTipText = "Guardar datos del candidato"
        cmdmodificar.Caption = "&Cancelar"
        cmdmodificar.ToolTipText = "Cancelar proceso"
        ado_candidatos.Recordset.AddNew
        desbloquear
        txtcedula.SetFocus
        Label13.Visible = False
        Label14.Visible = False
        Label15.Visible = False
        cmdeliminar.Enabled = False
        cmdcerrar.Enabled = False
        cmdbuscar.Enabled = False
        cmdinscripcion.Enabled = False

    Else
        cmdagregar.Caption = "&Agregar"
        cmdmodificar.Caption = "&Modificar"

        Do
            If Not IsNumeric(txtcedula.Text) Then
                X = InputBox("Digite su documento en numeros", "Error de tipo de Caracter")
                txtcedula.Text = X
            End If

            If txtnom.Text = "" Then
                X = InputBox("Debe escribir los nombres del candidato", "Nombres del candidato")
                txtnom.Text = X
            End If
            If txtapellidos.Text = "" Then
                X = InputBox("Debe escribir los apellidos del candidato", "Apellidos del candidato")
                txtapellidos.Text = X
            End If

```

```

If txttipo.Text = "" Then
    MsgBox "Debe escoger el tipo de documento", vbCritical, "Error tipo de documento"
    v1 = InputBox("Tipo documento: 1-Cedula, 2-Tarjeta de identidad, 3-Cedula de extranjeria", "Escoger tipo de documento")
    If v1 = 1 Then
        txttipo.Text = "Cedula"
    End If
    If v1 = 2 Then
        txttipo.Text = "Tarjeta de identidad"
    End If
    If v1 = 3 Then
        txttipo.Text = "Cedula de extranjeria"
    End If
End If
If txtcedula.Text = "" Then
    X = InputBox("Debe escribir la Cedula del candidato", "Cedula del Candidato")
    txtcedula.Text = X
End If
If txtidir.Text = "" Then
    X = InputBox("Debe escribir la Dirección del candidato", "Direccion del candidato")
    txtidir.Text = X
End If
Loop Until IsNumeric(txtcedula.Text) And Not (txtnom.Text = "") And Not (txtapellidos.Text = "") And Not (txtcedula.Text = "") And Not (txtidir.Text = "") And Not (txttipo.Text = "")
ado_auditor.Recordset.AddNew
If txtsw.Text = 0 Then
    txtidenti.Text = txtcedula.Text
    txtoperacion.Text = "Inserción registro"
    txtusu.Text = frmprincipal.Label4
    txtfecha.Text = Format(Date, "dd-mm-yyyy")
    txthora.Text = Format(Now, "h:nn:ss")
    txtmodulo.Text = "Candidato"
Else
    txtantes.Text = txtantes1.Text
    txtdespues.Text = txtcedula.Text & ", " & txttipo.Text & ", " & txtnom.Text & ", " & txtapellidos.Text & ", " & txtcelular.Text & ", " & txtfijo & ", " & optsex.Text & ", " & txtemail.Text & ", " & txtidir.Text & ", " & txtnorganizacion.Text & ", " & txtorgdir.Text & ", " & txttelorg
    txtidenti.Text = txtcedula.Text
    txtoperacion.Text = "Actualización registro"
    txtusu.Text = frmprincipal.Label4
    txtfecha.Text = Format(Date, "dd-mm-yyyy")
    txthora.Text = Format(Now, "h:nn:ss")
    txtmodulo.Text = "Candidato"
End If
ado_auditor.Recordset.Update
ado_candidatos.Recordset.Update
cmdinscripcion.Enabled = False
bloquear
blanco
ado_candidatos.Enabled = True
cmdeliminar.Enabled = True
cmdcerrar.Enabled = True
cmdbuscar.Enabled = True
cmdinscripcion.Enabled = True
Label14.Visible = True
End If
End Sub
Private Sub cmdbuscar_Click()
adobuscarv.Refresh
Dim bus As String
Dim bus2 As String
bus = InputBox("Digite La cedula del candidato", "Buscar Candidato")
If bus <> vbNullString Then
    ado_candidatos.Refresh
    ado_candidatos.Recordset.MoveFirst
    ado_candidatos.Recordset.Find ("can_cedula=" & bus & "")
    If txttestadov.Text <> "Tramite" Then
        Label14.Visible = True
        cmdinscripcion.Enabled = True
    End If
    If txttestadov.Text <> "" Then
        txtvolante.Text = txtcedula.Text
        bus2 = bus
        adobuscarv.Recordset.MoveFirst
        adobuscarv.Recordset.Find ("vol_cedula=" & bus2 & "")
    End If
End If

```

```

If txttestadov.Text = "Tramite" Then
    Label15.Visible = False
    Label14.Visible = False
    Label13.Visible = True
    cmdinscripcion.Enabled = True
    'cmdinscripcion.Caption = "Legalizar Inscripcion"
    cmdinscripcion.Caption = "Aplicar pago"
Else
    If txttestadov.Text = "Cancelado" Then
        Label14.Visible = False
        Label15.Visible = True
        Label13.Visible = False
        cmdinscripcion.Enabled = True
        cmdinscripcion.Caption = "Diligenciar Matricula"
    Else
        Label14.Visible = True
        cmdinscripcion.Enabled = True
    End If
End If
If adobuscarv.Recordset.EOF Then
    adobuscarv.Recordset.MoveFirst
End If
End If
If ado_candidatos.Recordset.EOF Then
    MsgBox "Verifique La cedula del candidato", vbInformation + vbOKOnly, "Cedula Incorrecta"
    ado_candidatos.Recordset.MoveFirst
    Label13.Visible = False
    Label14.Visible = False
    Label15.Visible = False
    cmdinscripcion.Enabled = False
    blanco
End If
Else
    MsgBox "Vuelva a intentar la busqueda - Cedula Nula", vbInformation + vbOKCancel, "Error de Busqueda"
    Label13.Visible = False
    Label14.Visible = False
    Label15.Visible = False
    cmdinscripcion.Enabled = False
    blanco
End If
End Sub
Private Sub cmdeliminar_Click()
    Dim resp As String
    resp = MsgBox("¿Seguro que desea eliminar este registro?", vbYesNo + vbInformation, "Eliminar registro")
    If resp = vbYes Then
        ado_auditor.Recordset.AddNew
        txtidenti.Text = txtcedula.Text
        ado_candidatos.Recordset.Delete
        ado_candidatos.Recordset.MovePrevious
        txtoperacion.Text = "Eliminación registro"
        txtusu.Text = frmprincipal.Label4
        txtfecha.Text = Format(Date, "dd-mm-yyyy")
        txthora.Text = Format(Now, "h:nn:ss")
        txtmodulo.Text = "Candidato"
        ado_auditor.Recordset.Update
    End If
    ado_candidatos.Recordset.MoveFirst
    blanco
End Sub

Private Sub cmdinscripcion_Click()
    Dim s As String
    If cmdinscripcion.Caption = "Diligenciar Matricula" Then
        frmmatricula.txtcedula.Text = txtcedula.Text
        s = "select c.can_nombres,c.can_apellidos,p.pro_nombre,p.pro_valor,i.ins_codprograma from inscripcion i,programa p,candidato c where " + frmmatricula.txtcedula.Text + " = i.ins_idcandidato and i.ins_idcandidato = c.can_cedula and i.ins_codprograma = p.pro_codigo "
        frmmatricula.ado_inscripcion.RecordSource = s
        frmmatricula.ado_inscripcion.Refresh
        frmmatricula.Label1.Caption = frmmatricula.txtnombre.Text + " " + frmmatricula.txtapellidos.Text
        If frmmatricula.Label1.Caption = " " Then
            frmmatricula.cmdvolante.Enabled = False
            frmmatricula.cmdnuevo.Enabled = False
            MsgBox "Registro no encontrado", vbCritical
        End If
    End If

```

```

Else
    frmmatricula.cmdvolante.Enabled = True
    frmmatricula.cmdnuevo.Enabled = True
End If
Unload Me
frmmatricula.Show vbModal
Else
    If cmdinscripcion.Caption = "Aplicar pago" Then
        Unload Me
        frmaplicar.Show vbModal
    Else
        frminscripcion.txtnom = txtnom.Text
        frminscripcion.txtapellidos = txtapellidos.Text
        frminscripcion.txtcedula1.Text = txtcedula.Text
        frminscripcion.txttipo = txttipo.Text
        Unload Me
        frminscripcion.Show vbModal
    End If
End If
End Sub

Private Sub cmdModificar_Click()
    If cmdmodificar.Caption = "&Modificar" Then

        'estas instrucciones dan como resultado el temporal del registro actual que despues se guardan en la tabla auditor
        txtantes1.Text = txtcedula.Text & ", " & txttipo.Text & ", " & txtnom.Text & ", " & txtapellidos.Text & ", " &
        txtcelular.Text & ", " & txtfijo & ", " & optsex.Text & ", " & txtemail.Text & ", " & txtmdir.Text & ", " & txtnorganizacion.Text &
        ", " & txtorgdir.Text & ", " & txttelorg
        txtsw.Text = 1
        cmdmodificar.Caption = "&Cancelar"
        cmdagregar.Caption = "&Guardar"
        cmdagregar.ToolTipText = "Guardar datos del candidato"
        cmdmodificar.ToolTipText = "Cancelar proceso"
        desbloquear
        txtcedula.SetFocus
        Label13.Visible = False
        Label14.Visible = False
        Label15.Visible = False
        ado_candidatos.Enabled = False
        cmdeliminar.Enabled = False
        cmdcerrar.Enabled = False
        cmdbuscar.Enabled = False
        cmdinscripcion.Enabled = False
        sw.Text = "1"
    Else
        sw.Text = "0"
        ado_candidatos.Recordset.CancelUpdate
        blanco
        cmdmodificar.Caption = "&Modificar"
        cmdagregar.Caption = "&Agregar"
        cmdagregar.ToolTipText = "Agregar un nuevo candidato"
        cmdmodificar.ToolTipText = "Modificar un candidato existente"
        ado_candidatos.Enabled = True
        cmdeliminar.Enabled = True
        cmdcerrar.Enabled = True
        cmdbuscar.Enabled = True
        cmdinscripcion.Enabled = True
        bloquear
    End If
End Sub

Private Sub cmdcerrar_Click()
    X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
    If X = vbYes Then
        Unload Me
    End If
End Sub

Private Sub cmdnueva_Click()
    cmdinscripcion.Caption = ""
End Sub

Private Sub Form_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then
        SendKeys "{Tab}"
    End If
End Sub

Private Sub Form_Load()

```

```

blanco
Label14.Visible = False
bloquear
End Sub

Private Sub optegren_Click()
    If optegren = True Then
        optegresado.Text = "N"
    End If
End Sub
Private Sub optegres_Click()
    If optegres = True Then
        optegresado.Text = "S"
    End If
End Sub

Private Sub optfemenino_Click()
    If optfemenino = True Then
        optsex.Text = "F"
    End If
End Sub

Private Sub Option2_Click()

End Sub

Private Sub optmasculino_Click()
    If optmasculino = True Then
        optsex.Text = "M"
    End If
End Sub

Private Sub Text1_Change()

End Sub

Private Sub txtapellidos_Change()
    Dim i As Integer
    txtapellidos.Text = UCase(txtapellidos.Text)
    i = Len(txtapellidos.Text)
    txtapellidos.SelStart = i
End Sub

Private Sub txtcedula_Click()
    If txtcedula.Text <> "" Then
        Do
            If Not IsNumeric(txtcedula.Text) Then
                X = InputBox("Digite su documento en numeros", "Error de tipo de Caracter")
                txtcedula.Text = X
            End If
            con1 = "select can_cedula from candidato where " & txtcedula.Text & " = can_cedula"
            ado_val.RecordSource = con1
            ado_val.Refresh
            If txtval.Text = txtcedula.Text Then
                MsgBox "Registro ingresado ya existe", vbCritical, "Registro Duplicado"
                ado_candidatos.Recordset.CancelUpdate
                cmdmodificar.Caption = "&Modificar"
                cmdagregar.Caption = "&Agregar"
                cmdagregar.ToolTipText = "Agregar un nuevo candidato"
                cmdmodificar.ToolTipText = "Modificar un candidato existente"
                ado_candidatos.Enabled = True
                cmdeliminar.Enabled = True
                cmdcerrar.Enabled = True
                cmdbuscar.Enabled = True
                cmdinscripcion.Enabled = True
                bloquear
            End If
        Loop Until IsNumeric(txtcedula.Text)
    End If
End Sub
Private Sub txtcedula_KeyPress(KeyAscii As Integer)
    If txtcedula.Text <> "" Then
        If KeyAscii = 13 Or KeyAscii = 9 Then
            Do
                If Not IsNumeric(txtcedula.Text) Then

```

```

        X = InputBox("Digite su documento en numeros", "Error de tipo de Caracter")
        txtcedula.Text = X
    End If
    con1 = "select can_cedula from candidato where '" + txtcedula.Text + "' = can_cedula"
    ado_val.RecordSource = con1
    ado_val.Refresh
    If txtval.Text = txtcedula.Text Then
        MsgBox "Registro ingresado ya existe", vbCritical, "Registro Duplicado"
        ado_candidatos.Recordset.CancelUpdate
        cmdmodificar.Caption = "&Modificar"
        cmdagregar.Caption = "&Agregar"
        cmdagregar.ToolTipText = "Agregar un nuevo candidato"
        cmdmodificar.ToolTipText = "Modificar un candidato existente"
        ado_candidatos.Enabled = True
        cmdeliminar.Enabled = True
        cmdcerrar.Enabled = True
        cmdbuscar.Enabled = True
        cmdinscripcion.Enabled = True
        bloquear
    End If
    Loop Until IsNumeric(txtcedula.Text)
End If
End Sub

Private Sub txtcedula_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If txtcedula.Text <> "" Then
        Do
            con1 = "select can_cedula from candidato where '" + txtcedula.Text + "' = can_cedula"
            ado_val.RecordSource = con1
            ado_val.Refresh
            If txtval.Text = txtcedula.Text And sw.Text = "0" Then
                MsgBox "Registro ingresado ya existe", vbCritical, "Registro Duplicado"
                ado_candidatos.Recordset.CancelUpdate
                cmdmodificar.Caption = "&Modificar"
                cmdagregar.Caption = "&Agregar"
                cmdagregar.ToolTipText = "Agregar un nuevo candidato"
                cmdmodificar.ToolTipText = "Modificar un candidato existente"
                ado_candidatos.Enabled = True
                cmdeliminar.Enabled = True
                cmdcerrar.Enabled = True
                cmdbuscar.Enabled = True
                cmdinscripcion.Enabled = True
                bloquear
            End If
            Loop Until IsNumeric(txtcedula.Text)
        End If
    End Sub

Private Sub txtdir_Change()
    Dim i As Integer
    txtdir.Text = UCase(txtdir.Text)
    i = Len(txtdir.Text)
    txtdir.SelStart = i
End Sub

Private Sub txtemail_Change()
    Dim i As Integer
    txtemail.Text = LCase(txtemail.Text)
    i = Len(txtemail.Text)
    txtemail.SelStart = i
End Sub

Private Sub txtnom_Change()
    Dim i As Integer
    txtnom.Text = UCase(txtnom.Text)
    i = Len(txtnom.Text)
    txtnom.SelStart = i
End Sub

Private Sub txtnorganizacion_Change()
    Dim i As Integer
    txtnorganizacion.Text = UCase(txtnorganizacion.Text)
    i = Len(txtnorganizacion.Text)

```



```
txtnorganizacion.SelStart = i
End Sub
```

```
Private Sub txtorgdir_Change()
Dim i As Integer
txtorgdir.Text = UCase(txtorgdir.Text)
i = Len(txtorgdir.Text)
txtorgdir.SelStart = i
End Sub
```

```
Private Sub txttelorg_Change()
Dim i As Integer
txttelorg.Text = UCase(txttelorg.Text)
i = Len(txttelorg.Text)
txttelorg.SelStart = i
End Sub
```

```
Private Sub txttipo_KeyPress(KeyAscii As Integer)
If txttipo.Text = "" Then
MsgBox "Debe escoger el tipo de documento", vbCritical, "Error tipo de documento"
txttipo.SetFocus
End If
End Sub
```

Frmconasig

```
Private Sub cmdasignar_Click()
frmnotdoc.txtcodasi.Text = txtcodigo.Text
frmnotdoc.Label6.Caption = txtnombre.Text
'frmnotdoc.txtcodgrupo.Text = txtinte.Text
Unload Me
End Sub
```

```
Private Sub DataGrid1_Click()
txtcodigo.Text = DataGrid1.Columns("Codigo").Text
txtnombre.Text = DataGrid1.Columns("Nombre").Text
txtinte.Text = DataGrid1.Columns("Grupo").Text
End Sub
```

```
Private Sub DataGrid1_RowColChange(LastRow As Variant, ByVal LastCol As Integer)
txtcodigo.Text = DataGrid1.Columns("Codigo").Text
txtnombre.Text = DataGrid1.Columns("Nombre").Text
txtinte.Text = DataGrid1.Columns("Grupo").Text
End Sub
```

Frmconpen

```
Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "reportes.chm", &O0, &O0, SW_NORMAL
End Sub
```

```
Private Sub cmdimprimir_Click()
c = "Select p.pro_nombre,a.asi_nombre,a.asi_semestre from asignatura a,pensum pe, programa p where pe.pen_codasignatura = a.asi_codigo and pe.pen_codprograma = '" & txtx.Text & "' and p.pro_codigo= '" & txtx.Text & "' order by a.asi_semestre"
rpt_asignaturas.Commands(1).CommandText = c
rpt_asignatura.Show vbModal
End Sub
```

```
Private Sub Form_Load()
Dim X As String
X = txtx.Text
End Sub
```

```
Private Sub Limpiar_Click()
'ado_consulta.Recordset.Delete
'ado_consulta.Recordset.MovePrevious
X = MsgBox("¿Desea cerrar este módulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
```

```

If X = vbYes Then
    Unload Me
End If

```

```

End Sub

```

Frmconsulta

```

Private Sub cmdasignar_Click()
    frmestudiante.txtmatricula.Text = txtmatricula.Text
    frmestudiante.txtsemestre.Text = txtsemestre.Text
    frmestudiante.txtprograma.Text = txtprograma.Text
    frmestudiante.cmdvolante.Enabled = True
    MsgBox "Presionar el botón - Generar Volante - Para Nuevo volante de Matrícula", vbInformation, "Generación Nuevo Volante de Matrícula"
    Unload Me
End Sub

```

```

Private Sub DataGrid1_Click()
    txtmatricula.Text = DataGrid1.Columns("Matricula").Text
    txtsemestre.Text = DataGrid1.Columns("Semestre").Text
    txtprograma.Text = DataGrid1.Columns("Programa2").Text
End Sub
Private Sub DataGrid1_RowColChange(LastRow As Variant, ByVal LastCol As Integer)
    txtmatricula.Text = DataGrid1.Columns("Matricula").Text
    txtsemestre.Text = DataGrid1.Columns("Semestre").Text
End Sub

```

Frmconvol

```

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
    ShellExecute frmprincipal.hwnd, "Open", "reportes.chm", &O0, &O0, SW_NORMAL
End Sub

```

```

Private Sub cmdbuscar_Click()
    Dim s As String
    s = "select v.vol_inte,c.can_nombres,c.can_apellidos,v.vol_tipo,v.vol_estado,v.vol_valor,p.pro_nombre from candidato c,programa p,volantes v where " & txtcedula.Text & " = v.vol_cedula and " & txtcedula.Text & " = c.can_cedula and v.vol_programa = p.pro_codigo"
    ado_consultas.RecordSource = s
    ado_consultas.Refresh

```

```

Label2.Caption = txtnombre.Text & " " & txtapellido.Text
End Sub

```

```

Private Sub cmdcerrar_Click()
    X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
    If X = vbYes Then
        Unload Me
    End If
End Sub

```

```

Private Sub cmdvolante_Click()
    If txtcedula.Text <> "" Then
        txtnum.Text = DataGrid1.Columns("Numero").Text
        MsgBox "Volante Duplicado Generado - Número: " & txtnum.Text, vbInformation, "Volantes Duplicados"
    Else
        MsgBox "No ha seleccionado ningun volante", vbCritical, "Impresión Errada"
    End If
End Sub

```

```

Private Sub DataGrid1_RowColChange(LastRow As Variant, ByVal LastCol As Integer)
    txtnum.Text = DataGrid1.Columns("Numero").Text
    cmdvolante.Enabled = True
End Sub

```



```

Private Sub txtcedula_KeyPress(KeyAscii As Integer)
Dim s As String
If KeyAscii = 13 Then
    s = "select v.vol_inte,c.can_nombres,c.can_apellidos,v.vol_tipo,v.vol_estado,v.vol_valor,p.pro_nombre from
candidato c,programa p,volantes v where " + txtcedula.Text + " = v.vol_cedula and " + txtcedula.Text + " = c.can_cedula
and v.vol_programa = p.pro_codigo"
    ado_consultas.RecordSource = s
    ado_consultas.Refresh
    If DataGrid1.Bookmark = 1 Then
        DataGrid1.Enabled = True
        Label2.Caption = txtnombre.Text + " " + txtapellido.Text
    Else
        MsgBox "La cedula no posee volantes asociados", vbCritical, "Error busqueda"
    End If
    Unload Me
    frmconvol.Show vbModal
End If

```

End If

End Sub

Frmdoc_asig

```

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation
As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As
Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "parametrizacion.chm", &O0, &O0, SW_NORMAL
End Sub

```

```

Private Sub cmdcancelar_Click()
txtcedula.Text = ""
txtnom.Text = ""
txtapellidos.Text = ""
txtcedula.Enabled = True
cmdregistrar.SetFocus
cmdcancelar.Enabled = False
ado_guardar.Recordset.CancelUpdate
cmdregistrar.Caption = "&Registrar Docente - Asignatura"
End Sub
Private Sub cmdcerrar_Click()
X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
If X = vbYes Then
    Unload Me
End If
End Sub
Private Sub cmdregistrar_Click()
txtcedula.Enabled = True
txtcedula.SetFocus
cmdcancelar.Enabled = True
s = txtcodigo.Text
If cmdregistrar.Caption = "&Guardar" Then
    If txtcedula.Text = "" Or txtcodigo.Text = "" Then
        MsgBox "No se puede guardar registro", vbCritical, "Cédula Inválida / Asignatua Inválida"
        txtcedula.SetFocus
    End If
    If txtcedula.Text <> "" And txtcodigo.Text <> "" Then
        txtcedula2.Text = txtcedula.Text
        txtcodigo.Text = s
        'esta instruccion se utiliza para crear el registro en la tabla auditor
        ado_auditor.Recordset.AddNew
        txtiden2.Text = txtcodigo.Text
        txtidenti.Text = txtcedula.Text
        txtoperacion.Text = "Registro docente asignatura"
        txtusu.Text = frmprincipal.Label4
        txtfecha.Text = Format(Date, "dd-mm-yyyy")
        txthora.Text = Format(Now, "h:nn:ss")
        txtmodulo.Text = "Docente_asignatura "
        ado_auditor.Recordset.Update
    End If
    ado_guardar.Recordset.Update
    MsgBox "El proceso de se llevo a cabo con éxito", vbInformation, "Proceso Exitoso"
    cmdregistrar.Caption = "&Registrar Docente - Asignatura"
End Sub

```

```

        DataGrid1.Enabled = False
        cmdcancelar.Enabled = False
        txtcedula.Text = ""
        txtnom.Text = ""
        txtapellidos.Text = ""
        txtcedula.Enabled = False
    End If

Else

    MsgBox "Diligenciar datos del Docente", vbInformation, "Docente"
    cmdregistrar.Caption = "&Guardar"
    DataGrid1.Enabled = True
    End If
End Sub

Private Sub DataGrid1_RowColChange(LastRow As Variant, ByVal LastCol As Integer)
txtcodigo.Text = DataGrid1.Columns("Código").Text
X = "Se asignó la asignatura " + txtcodigo.Text + " " + DataGrid1.Columns("Nombre Asignatura").Text
MsgBox X
MsgBox "Proceda a Guardar la Asignación", vbInformation, "Asignación exitosa"
cmdregistrar.SetFocus
End Sub

Private Sub txtcedula_KeyPress(KeyAscii As Integer)
Dim bus As String
If KeyAscii = 13 Then
    bus = "Select doc_identifi,doc_nombres,doc_apellidos from docente where doc_identifi = " + txtcedula.Text + ""
    ado_docentes.RecordSource = bus
    ado_docentes.Refresh
    ado_guardar.Recordset.AddNew
    If txtnom.Text = "" Then
        MsgBox "Verifique la identificación del Docente", vbCritical, "Error de identificación"
        ado_guardar.Recordset.CancelUpdate
    Else
        MsgBox "Escoja la asignatura correspondiente", vbInformation, "Asignaturas"
    End If
End If
End Sub
End Sub

```

Frmdocente

```

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation
As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As
Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "parametrizacion.chm", &O0, &O0, SW_NORMAL
End Sub

Sub bloquear()
    txtnom.Enabled = False
    txtapellidos.Enabled = False
    txtcedula.Enabled = False
    txtfijo.Enabled = False
    txtcelular.Enabled = False
    txtdir.Enabled = False
    txtemail.Enabled = False
    txtinfo.Enabled = False
    txttipo.Enabled = False
    optmascu.Enabled = False
    optfemenino.Enabled = False
End Sub

Sub desbloquear()
    txtnom.Enabled = True
    txtapellidos.Enabled = True
    txtcedula.Enabled = True
    txtfijo.Enabled = True
    txtcelular.Enabled = True
    txtdir.Enabled = True
    txtemail.Enabled = True
    txtinfo.Enabled = True
    txttipo.Enabled = True
    optmascu.Enabled = True
    optfemenino.Enabled = True

```

```

txtidenti.Text = txtcedula.Text
txtoperacion.Text = "Actualización registro"
txtusu.Text = frmprincipal.Label4
txtfecha.Text = Format(Date, "dd-mm-yyyy")
txthora.Text = Format(Now, "h:nn:ss")
txtmodulo.Text = "Docente"
End If
ado_auditor.Recordset.Update

ado_docentes.Recordset.Update
blanco
bloquear
ado_docentes.Enabled = True
cmdeliminar.Enabled = True
cmdcerrar.Enabled = True
cmdbuscar.Enabled = True

End If
End Sub
Private Sub cmdbuscar_Click()
Dim bus As String
bus = InputBox("Digite La cedula del docente", "Buscar docente")
If bus <> vbNullString Then
ado_docentes.Refresh
ado_docentes.Recordset.MoveFirst
ado_docentes.Recordset.Find ("doc_identi=" & bus & "")
If ado_docentes.Recordset.EOF Then
MsgBox "Verifique La cedula del docente", vbInformation + vbOKOnly, "Cedula Incorrecta"
blanco
ado_docentes.Refresh

ado_docentes.Recordset.MoveFirst

End If
Else
ado_docentes.Refresh
MsgBox "Vuelva a intentar la busqueda - Cedula Nula", vbInformation + vbOKCancel, "Error de Busqueda"
blanco
End If
End Sub
Private Sub cmdeliminar_Click()
Dim resp As String
resp = MsgBox("¿Seguro que desea eliminar este registro?", vbYesNo + vbInformation, "Eliminar registro")
If resp = vbYes Then
ado_auditor.Recordset.AddNew
txtidenti.Text = txtcedula.Text
ado_docentes.Recordset.Delete
ado_docentes.Recordset.MovePrevious
txtoperacion.Text = "Eliminación registro"
txtusu.Text = frmprincipal.Label4
txtfecha.Text = Format(Date, "dd-mm-yyyy")
txthora.Text = Format(Now, "h:nn:ss")
txtmodulo.Text = "Docente"
ado_auditor.Recordset.Update
End If
ado_docentes.Recordset.MoveFirst
blanco
End Sub
Private Sub cmdModificar_Click()
If cmdmodificar.Caption = "&Modificar" Then
'estas instrucciones dan como resultado el temporal del registro actual que despues se guardan en la tabla auditor
txtantes1.Text = txtcedula.Text & ", " & txttipo.Text & ", " & txtnom.Text & ", " & txtapellidos.Text & ", " &
txtcelular.Text & ", " & txtfijo & ", " & txtsexo.Text & ", " & txtemail.Text & ", " & txtdir.Text & ", " & txtinfo.Text
sw.Text = 1
txtsw.Text = 1
cmdmodificar.Caption = "&Cancelar"
cmdagregar.Caption = "&Guardar"
cmdagregar.ToolTipText = "Guardar datos del docente"
cmdmodificar.ToolTipText = "Cancelar proceso"
desbloquear
txtcedula.SetFocus
ado_docentes.Enabled = False
cmdeliminar.Enabled = False
cmdcerrar.Enabled = False

```

```

        cmdbuscar.Enabled = False
    Else
        sw.Text = "0"
        ado_docentes.Recordset.CancelUpdate
        blanco
        cmdmodificar.Caption = "&Modificar"
        cmdagregar.Caption = "&Agregar"
        cmdagregar.ToolTipText = "Agregar un nuevo docente"
        cmdmodificar.ToolTipText = "Modificar un docente existente"
        ado_docentes.Enabled = True
        cmdeliminar.Enabled = True
        cmdcerrar.Enabled = True
        cmdbuscar.Enabled = True
        bloquear
    End If
End Sub
Private Sub cmdcerrar_Click()
X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
If X = vbYes Then
    Unload Me
End If
End Sub
Private Sub Form_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
    SendKeys "{Tab}"
End If
End Sub

Private Sub Form_Load()
blanco
bloquear
End Sub
Private Sub optfemenino_Click()
If optfemenino.Value = True Then
    txtsexo.Text = "F"
End If
End Sub
Private Sub optmascul_Click()
If optmascul.Value = True Then
    txtsexo.Text = "M"
End If
End Sub
Private Sub txtapellidos_Change()
Dim i As Integer
txtapellidos.Text = UCase(txtapellidos.Text)
i = Len(txtapellidos.Text)
txtapellidos.SelStart = i
End Sub

Private Sub txtcedula_Click()
If txtcedula.Text <> "" Then
Do
    If Not IsNumeric(txtcedula.Text) Then
        X = InputBox("Digite su documento en numeros", "Error de tipo de Caracter")
        txtcedula.Text = X
    End If
    con2 = "select doc_identi from docente where " & txtcedula.Text & " = doc_identi"
    ado_val.RecordSource = con2
    ado_val.Refresh
    If txtval.Text = txtcedula.Text Then
        MsgBox "Registro ingresado ya existe", vbCritical, "Registro Duplicado"
        ado_docentes.Recordset.CancelUpdate
        cmdmodificar.Caption = "&Modificar"
        cmdagregar.Caption = "&Agregar"
        cmdagregar.ToolTipText = "Agregar un nuevo docente"
        cmdmodificar.ToolTipText = "Modificar un docente existente"
        ado_docentes.Enabled = True
        cmdeliminar.Enabled = True
        cmdcerrar.Enabled = True
        cmdbuscar.Enabled = True
        bloquear
    End If
    Loop Until IsNumeric(txtcedula.Text)
End If
End Sub

```

```

Private Sub txtcedula_KeyPress(KeyAscii As Integer)
If txtcedula.Text <> "" Then
If KeyAscii = 13 Then
Do
If Not IsNumeric(txtcedula.Text) Then
X = InputBox("Digite su documento en numeros", "Error de tipo de Caracter")
txtcedula.Text = X
End If
con2 = "select doc_identi from docente where "" + txtcedula.Text + "" = doc_identi"
ado_val.RecordSource = con2
ado_val.Refresh
If txtval.Text = txtcedula.Text Then
MsgBox "Registro ingresado ya existe", vbCritical, "Registro Duplicado"
ado_docentes.Recordset.CancelUpdate
cmdmodificar.Caption = "&Modificar"
cmdagregar.Caption = "&Agregar"
cmdagregar.ToolTipText = "Agregar un nuevo docente"
cmdmodificar.ToolTipText = "Modificar un docente existente"
ado_docentes.Enabled = True
cmdeliminar.Enabled = True
cmdcerrar.Enabled = True
cmdbuscar.Enabled = True
bloquear
End If
Loop Until IsNumeric(txtcedula.Text)
End If
End If
End Sub

Private Sub txtcedula_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
If txtcedula.Text <> "" Then
Do
con2 = "select doc_identi from docente where "" + txtcedula.Text + "" = doc_identi"
ado_val.RecordSource = con2
ado_val.Refresh
If txtval.Text = txtcedula.Text And sw.Text = "0" Then
MsgBox "Registro ingresado ya existe", vbCritical, "Registro Duplicado"
ado_docentes.Recordset.CancelUpdate
cmdmodificar.Caption = "&Modificar"
cmdagregar.Caption = "&Agregar"
cmdagregar.ToolTipText = "Agregar un nuevo docente"
cmdmodificar.ToolTipText = "Modificar un docente existente"
ado_docentes.Enabled = True
cmdeliminar.Enabled = True
cmdcerrar.Enabled = True
cmdbuscar.Enabled = True
bloquear
End If
Loop Until IsNumeric(txtcedula.Text)
End If
End Sub

Private Sub txtmdir_Change()
Dim i As Integer
txtmdir.Text = UCase(txtmdir.Text)
i = Len(txtmdir.Text)
txtmdir.SelStart = i
End Sub

Private Sub txtemail_Change()
Dim i As Integer
txtemail.Text = LCase(txtemail.Text)
i = Len(txtemail.Text)
txtemail.SelStart = i
End Sub

Private Sub txtnom_Change()
Dim i As Integer
txtnom.Text = UCase(txtnom.Text)
i = Len(txtnom.Text)
txtnom.SelStart = i
End Sub

Private Sub txtinfo_Change()
Dim i As Integer
txtinfo.Text = UCase(txtinfo.Text)
i = Len(txtinfo.Text)
txtinfo.SelStart = i
End Sub

```

```

End Sub
Private Sub txttipo_KeyPress(KeyAscii As Integer)
If txttipo.Text = "" Then
    MsgBox "Debe escoger el tipo de documento", vbCritical, "Error tipo de documento"
    txttipo.SetFocus
End If
End Sub

```

```

Sub blanco()
    txtcedula.Text = " "
    txttipo.Text = " "
    txtnom.Text = " "
    txtapellidos.Text = " "
    txtcelular.Text = " "
    txtfijo.Text = " "
    txtemail.Text = " "
    txtdir.Text = " "
    txtinfo.Text = " "
End Sub

```

Frimestudiante

```

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "parametrizacion.chm", &O0, &O0, SW_NORMAL
End Sub
Sub bloquear()
txttipo.Enabled = False
'txtcedula.Enabled = False
txtnombres.Enabled = False
txtapellidos.Enabled = False
txtdir.Enabled = False
txtfijo.Enabled = False
txtcelular.Enabled = False
txtemail.Enabled = False
txtnaci.Enabled = False
txtcivil.Enabled = False
optmasculino.Enabled = False
optfemenino.Enabled = False
End Sub
Sub desbloquear()
txttipo.Enabled = True
'txtcedula.Enabled = True
txtnombres.Enabled = True
txtapellidos.Enabled = True
txtdir.Enabled = True
txtfijo.Enabled = True
txtcelular.Enabled = True
txtemail.Enabled = True
txtnaci.Enabled = True
txtcivil.Enabled = True

optmasculino.Enabled = True
optfemenino.Enabled = True
End Sub

Private Sub ado_estudiantes_WillMove(ByVal adReason As ADODB.EventReasonEnum, adStatus As ADODB.EventStatusEnum, ByVal pRecordset As ADODB.Recordset)
If optsex.Text = "M" Then
    optmasculino.Value = True
End If
If optsex.Text = "F" Then
    optfemenino.Value = True
End If
End Sub

Private Sub cmdagregar_Click()
If cmdagregar.Caption = "&Agregar" Then
    cmdagregar.Caption = "&Guardar"
    cmdagregar.ToolTipText = "Guardar datos del estudiante"
    cmdmodificar.Caption = "&Cancelar"
    cmdmodificar.ToolTipText = "Cancelar proceso"

```

```

ado_estudiantes.Recordset.AddNew
desbloquear
txtcedula.SetFocus
ado_estudiantes.Enabled = False
cmdeliminar.Enabled = False
cmdcerrar.Enabled = False
cmdbuscar.Enabled = False

Else
    cmdagregar.Caption = "&Agregar"
    cmdmodificar.Caption = "&Modificar"

    Do
        If txtnombres.Text = "" Then
            X = InputBox("Debe escribir los nombres del estudiante", "Nombres del estudiante")
            txtnombres.Text = X
        End If
        If txtapellidos.Text = "" Then
            X = InputBox("Debe escribir los apellidos del estudiante", "Apellidos del estudiante")
            txtapellidos.Text = X
        End If
        If txttipo.Text = "" Then
            MsgBox "Debe escoger el tipo de documento", vbCritical, "Error tipo de documento"
            v1 = InputBox("Tipo documento: 1-Cedula, 2-Tarjeta de identidad, 3-Cedula de extranjeria", "Escoger tipo de documento")
            If v1 = 1 Then
                txttipo.Text = "Cedula"
            End If
            If v1 = 2 Then
                txttipo.Text = "Tarjeta de identidad"
            End If
            If v1 = 3 Then
                txttipo.Text = "Cedula de extranjeria"
            End If
        End If
        If txtcedula.Text = "" Then
            X = InputBox("Debe escribir la Cedula del estudiante", "Cedula del estudiante")
            txtcedula.Text = X
        End If
        If txtdir.Text = "" Then
            X = InputBox("Debe escribir la Dirección del estudiante", "Direccion del estudiante")
            txtdir.Text = X
        End If
        Loop Until Not (txtnombres.Text = "") And Not (txtapellidos.Text = "") And Not (txtcedula.Text = "") And Not (txtdir.Text = "") And Not (txttipo.Text = "")
        txtusuario.Text = frmlogin.txtusuario.Text
        ado_estudiantes.Recordset.Update
        bloquear
        ado_estudiantes.Enabled = True
        cmdeliminar.Enabled = True
        cmdcerrar.Enabled = True
        cmdbuscar.Enabled = True
    End If
End Sub

Private Sub cmdbuscar_Click()
    Dim bus As String
    bus = InputBox("Digite La cedula del estudiante", "Buscar estudiante")
    If bus <> vbNullString Then
        ado_estudiantes.Refresh
        ado_estudiantes.Recordset.MoveFirst
        ado_estudiantes.Recordset.Find ("est_identi=" & bus & "")
        If ado_estudiantes.Recordset.EOF Then
            MsgBox "Verifique La cedula del estudiante", vbInformation + vbOKOnly, "Cedula Incorrecta"
            ado_estudiantes.Refresh
            ado_estudiantes.Recordset.MoveFirst
        End If
    Else
        MsgBox "Vuelva a intentar la busqueda - Cedula Nula", vbInformation + vbOKCancel, "Error de Busqueda"
    End If
End Sub

Private Sub cmdconsultar_Click()

End Sub

```

```

Private Sub cmdeliminar_Click()
    Dim resp As String
    resp = MsgBox("¿Seguro que desea eliminar este registro?", vbYesNo + vbInformation, "Eliminar registro")
    If resp = vbYes Then
        ado_estudiantes.Recordset.Delete
        ado_estudiantes.Recordset.MovePrevious
    End If
    ado_estudiantes.Recordset.MoveFirst
End Sub
Private Sub cmdModificar_Click()
    If cmdmodificar.Caption = "&Modificar" Then
        cmdmodificar.Caption = "&Cancelar"
        cmdagregar.Caption = "&Guardar"
        cmdagregar.ToolTipText = "Guardar datos del estudiante"
        cmdmodificar.ToolTipText = "Cancelar proceso"
        desbloquear
        txtcedula.SetFocus
        ado_estudiantes.Enabled = False
        cmdeliminar.Enabled = False
        cmdcerrar.Enabled = False
        cmdbuscar.Enabled = False
    Else
        ado_estudiantes.Recordset.CancelUpdate
        cmdmodificar.Caption = "&Modificar"
        cmdagregar.Caption = "&Agregar"
        cmdagregar.ToolTipText = "Agregar un nuevo estudiante"
        cmdmodificar.ToolTipText = "Modificar un estudiante existente"
        ado_estudiantes.Enabled = True
        cmdeliminar.Enabled = True
        cmdcerrar.Enabled = True
        cmdbuscar.Enabled = True
        bloquear
    End If
End Sub
Private Sub cmdcerrar_Click()
    X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
    If X = vbYes Then
        Unload Me
    End If
End Sub

Private Sub cmdnueva_Click()
    frmconsulta.txtcedula.Text = txtcedula.Text
    frmconsulta.Label1.Caption = txtnombres.Text + " " + txtapellidos.Text
    s = "select mat_inte,mat_semestre,mat_grupo,mat_codgrupo,pro_nombre,mat_programa from matricula,programa"
    where "" + frmconsulta.txtcedula.Text + "" = mat_idestudiante and mat_programa = pro_codigo"
    frmconsulta.ado_consulta.RecordSource = s
    frmconsulta.ado_consulta.Refresh
    MsgBox "Consultar Matricula para generar nuevo volante", vbInformation, "Consulta Matricula"
    frmconsulta.Show vbModal
End Sub

Private Sub cmdvolante_Click()

    If txtsemestre.Text = "Primero" Then
        frmmatricula.txtsemestre.Text = "Segundo"
    End If
    If txtsemestre.Text = "Segundo" Then
        frmmatricula.txtsemestre.Text = "Tercero"
    End If
    If txtsemestre.Text = "Tercero" Then
        frmmatricula.txtsemestre.Text = "Cuarto"
    End If
    If txtsemestre.Text = "Cuarto" Then
        frmmatricula.txtsemestre.Text = "Quinto"
    End If
    If txtsemestre.Text = "Quinto" Then
        frmmatricula.txtsemestre.Text = "Sexto"
    End If
    If txtsemestre.Text = "Sexto" Then
        frmmatricula.txtsemestre.Text = "Septimo"
    End If
    frmmatricula.txtmatricula = txtmatricula.Text
    frmmatricula.txtprograma.Text = txtprograma.Text
    frmmatricula.txtcedula.Text = txtcedula.Text
    frmmatricula.cmdvolante.Enabled = True

```



```

s = "select pro_nombre,pro_valor from Programa where " + frmmatricula.txtprograma.Text + "= pro_codigo"
frmmatricula.ado_vol.RecordSource = s
frmmatricula.ado_vol.Refresh
frmmatricula.DataGrid2.Columns("Semestre").Text = frmmatricula.txtsemestre.Text

```

```

frmmatricula.DataGrid2.Visible = False
frmmatricula.txtsemestre.Visible = True
frmmatricula.Show vbModal
End Sub

```

```

Private Sub Form_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
    SendKeys "{Tab}"
End If
End Sub
Private Sub Form_Load()
    bloquear
End Sub

```

```

Private Sub optfemenino_Click()
If optfemenino = True Then
    optsex.Text = "F"
End If
End Sub
Private Sub optmasculino_Click()
If optmasculino = True Then
    optsex.Text = "M"
End If
End Sub
Private Sub txtapellidos_Change()
Dim i As Integer
txtapellidos.Text = UCase(txtapellidos.Text)
i = Len(txtapellidos.Text)
txtapellidos.SelStart = i
End Sub

```

```

Private Sub txtcedula_KeyPress(KeyAscii As Integer)

```

```

If KeyAscii = 13 Then
' If Not IsNumeric(txtcedula.Text) Then
*   X = InputBox("Digite su documento en numeros", "Error de tipo de Caracter")
*   txtcedula.Text = X
s = "select
est_identi,est_tipodi,est_nombres,est_apellidos,est_direccion,est_telefijo,est_telemovil,est_email,est_sexo,est_fnacimie
nto,est_ecivil,est_usuario,est_estado from estudiante where " + txtcedula.Text + "= est_identi"
ado_estudiantes.RecordSource = s
ado_estudiantes.Refresh
cmdnueva.Enabled = True
' End If
End If
End Sub

```

```

Private Sub txtdir_Change()
Dim i As Integer
txtdir.Text = UCase(txtdir.Text)
i = Len(txtdir.Text)
txtdir.SelStart = i
End Sub

```

```

Private Sub txtemail_Change()
Dim i As Integer
txtemail.Text = LCase(txtemail.Text)
i = Len(txtemail.Text)
txtemail.SelStart = i

```

```

End Sub
Private Sub txtestudios_KeyPress(KeyAscii As Integer)
Dim i As Integer
txtestudios.Text = UCase(txtestudios.Text)
i = Len(txtestudios.Text)
txtestudios.SelStart = i
End Sub

```

```

Private Sub txtnombres_Change()
Dim i As Integer
txtnombres.Text = UCase(txtnombres.Text)
i = Len(txtnombres.Text)
txtnombres.SelStart = i
End Sub

Private Sub txttipo_KeyPress(KeyAscii As Integer)
If txttipo.Text = "" Then
MsgBox "Debe escoger el tipo de documento", vbCritical, "Error tipo de documento"
txttipo.SetFocus
End If
End Sub

```

Frmexport

```

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "seguridad.chm", &O0, &O0, SW_NORMAL
End Sub

```

```

Private Sub cmdcerrar_Click()
X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
If X = vbYes Then
Unload Me
End If
End Sub

```

```

Private Sub Command1_Click()
On Error Resume Next
Me.CommonDialog1.FileName = "export"
Me.CommonDialog1.ShowSave
Me.Label1.Caption = Me.CommonDialog1.FileName
Dim l As Integer
l = Len(Me.Label1.Caption)
While l > 1 And Mid(Me.Label1.Caption, l, 1) <> "\"
l = l - 1
Wend
Me.CommonDialog1.Tag = Mid(Me.Label1.Caption, 1, l)
Me.Label1.Caption = Mid(Me.Label1.Caption, 1, l) & "codetec" & Format(Date, "yyyymmdd") & "\"
End Sub

```

```

Private Sub Command2_Click()
On Error Resume Next
Dim i As Integer
Dim Db As New ADODB.Connection
Db.Open "Provider=MSDASQL.1;Persist Security Info=False;Extended Properties='DRIVER={MySQL ODBC 3.51 Driver};SERVER=localhost;DATABASE=Codetec;UID=root;OPTION=3'"

```

```

Dim j As Integer
For i = 1 To Len(Me.Label1.Caption)
Replace
Next j
MkDir Me.Label1.Caption
Dim dirarc As String
dirarc = Replace(Me.Label1.Caption, "\", "/")

For i = 0 To Me.List1.ListCount - 1
If Me.List1.Selected(i) = True Then
Db.Execute "SELECT * INTO OUTFILE '" & dirarc & Me.List1.List(i) & ".txt' FROM " & Me.List1.List(i)
End If
Next i
MsgBox "El proceso de Exportación se realizó exitosamente", vbInformation, "Exportación Exitosa"
SaveSetting "CODETEC", "HOME", "EXPORT", Me.CommonDialog1.Tag
Db.Close
End Sub

```

```

Private Sub Form_Load()
Me.Label1.Caption = GetSetting("CODETEC", "HOME", "EXPORT", "C:\mysql\data") & "codetec" & Format(Date, "yyyymmdd") & "\"
Me.CommonDialog1.Tag = GetSetting("CODETEC", "HOME", "EXPORT", "C:\mysql\data")

```

```

Dim rs As New ADODB.Recordset

rs.Open "select rel_origen as name from relaciones Union select rel_destino as name from relaciones",
"Provider=MSDASQL.1;Persist Security Info=False;Extended Properties='DRIVER={MySQL ODBC 3.51
Driver};SERVER=localhost;DATABASE=Codetec ;UID=root ;OPTION=3'"

Me.List1.Clear
While Not rs.EOF
    Me.List1.AddItem rs!Name
    rs.MoveNext
Wend
rs.Close

End Sub

```

Frmgenerator

```

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation
As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As
Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "reportes.chm", &00, &00, SW_NORMAL
End Sub

```

```

Private Sub cmdcerrar_Click()
X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
If X = vbYes Then
    Unload Me
End If
End Sub

```

```

Private Sub Command1_Click()
On Error GoTo e
Dim ds As New FlexADO
Dim rs As New ADODB.Recordset

rs.Open Me.Text1.Text, "Provider=MSDASQL.1;Persist Security Info=False;Extended Properties='DRIVER={MySQL
ODBC 3.51 Driver};SERVER=localhost;DATABASE=Codetec ;UID=root ;OPTION=3'"

```

```

ds.OpenRecordset Me.Text1.Text, "Provider=MSDASQL.1;Persist Security Info=False;Extended
Properties='DRIVER={MySQL ODBC 3.51 Driver};SERVER=localhost;DATABASE=Codetec ;UID=root ;OPTION=3'"

```

```

Me.fg_res.FlexDataSource = ds

```

```

Dim c%, r%, w%, mw%
For c = 0 To fg_res.Cols - 1
    mw = 0
    For r = 0 To 5
        w = TextWidth(fg_res.TextMatrix(r, c))
        If w > mw Then mw = w
    Next
    fg_res.ColWidth(c) = mw + 100
    fg_res.TextMatrix(0, c) = rs.Fields(c).Name
Next

```

```

Exit Sub
e:
MsgBox Err.Description
End Sub

```

```

Private Sub Command2_Click()
Me.fg_res.PrintGrid
Me.fg_res.SaveGrid "c:\xx.xls", flexFileExcel
End Sub

```

```

Private Sub fg_cri_AfterEdit(ByVal Row As Long, ByVal Col As Long)
Call GenerarSql

```

End Sub

Private Sub fg_cri_BeforeEdit(ByVal Row As Long, ByVal Col As Long, Cancel As Boolean)

If Col = 2 Then

If fg_cri.TextMatrix(Row, 1) <> "" Then

Dim rs As New ADODB.Recordset

Dim i As Integer

Dim Cad As String

Cad = ""

rs.Open "desc " & fg_cri.TextMatrix(Row, Col - 1), "Provider=MSDASQL.1;Persist Security Info=False;Extended Properties='DRIVER={MySQL ODBC 3.51 Driver};SERVER=localhost;DATABASE=Codetec ;UID=root ;OPTION=3'"

While Not rs.EOF

Cad = Cad & rs!Field & "|"

rs.MoveNext

Wend

rs.Close

Me.fg_cri.ColComboList(Col) = Mid(Cad, 1, Len(Cad) - 1)

Else

Cancel = True

End If

End If

If Col > 2 Then

If fg_cri.TextMatrix(Row, 1) = "" Or fg_cri.TextMatrix(Row, 2) = "" Then

Cancel = True

ElseIf fg_cri.TextMatrix(Row, 2) = "" And Col <> 4 Then

Cancel = True

End If

End If

End Sub

Private Sub fg_cri_KeyDown(KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyDelete Then

fg_cri.RemoveItem Me.fg_cri.Row

Call GenerarSql

End If

End Sub

Private Sub fg_rel_AfterEdit(ByVal Row As Long, ByVal Col As Long)

Call GenerarSql

End Sub

Private Function Registro(Operador As String, Criterio As String, Optional Entre As String = "") As String

Select Case Operador

Case "Igual": Registro = "=" & Criterio & ""

Case "Diferente": Registro = "<>" & Criterio & ""

Case "Entre"

If Entre <> "" Then

Registro = " between " & Criterio & " and " & Entre & ""

End If

Case "Mayor o igual que": Registro = ">=" & Criterio & ""

Case "Mayor que": Registro = ">" & Criterio & ""

Case "Menor o igual que": Registro = "<=" & Criterio & ""

Case "Menor que": Registro = "<" & Criterio & ""

Case "Que comience por": Registro = " like " & Criterio & "%"

Case "Que contenga": Registro = " like '%" & Criterio & "%"

Case "Que termine en": Registro = " like '%" & Criterio & ""

End Select

End Function

Private Sub GenerarSql()

Dim c As Integer

Dim i As Integer

Dim Cad As String

Dim Ord As String

Dim Whe As String

c = fg_cri.Aggregate(flexSTCount, 1, 1, fg_cri.Rows - 1, 1)

Me.Text1.Text = "select "

Cad = ""

Ord = ""

```

Whe = ""

For i = 1 To c
    If Me.fg_cri.ValueMatrix(i, 4) = -1 Then
        Cad = Cad & Me.fg_cri.TextMatrix(i, 1) & "." & Me.fg_cri.TextMatrix(i, 2) & If(Me.fg_cri.TextMatrix(i, 3) <> "", " as" & Me.fg_cri.TextMatrix(i, 3), "") & ","
    End If

    If Trim(Me.fg_cri.TextMatrix(i, 5)) <> "" Then
        Ord = Ord & Me.fg_cri.TextMatrix(i, 1) & "." & Me.fg_cri.TextMatrix(i, 2) & If(Me.fg_cri.TextMatrix(i, 5) = "Ascendente", " asc", " desc") & ","
    End If

    If Trim(Me.fg_cri.TextMatrix(i, 6)) <> "" And Trim(Me.fg_cri.TextMatrix(i, 7)) <> "" Then
        Whe = Whe & "(" & Me.fg_cri.TextMatrix(i, 1) & "." & Me.fg_cri.TextMatrix(i, 2) & _
        Registro(Trim(Me.fg_cri.TextMatrix(i, 6)), Trim(Me.fg_cri.TextMatrix(i, 7)), Trim(Me.fg_cri.TextMatrix(i, 8)))

        If Trim(Me.fg_cri.TextMatrix(i, 8)) <> "" And Trim(Me.fg_cri.TextMatrix(i, 6)) <> "Entre" Then
            Whe = Whe & " or " & Me.fg_cri.TextMatrix(i, 1) & "." & Me.fg_cri.TextMatrix(i, 2) & _
            Registro(Trim(Me.fg_cri.TextMatrix(i, 6)), Trim(Me.fg_cri.TextMatrix(i, 8)))
        End If

        If Trim(Me.fg_cri.TextMatrix(i, 9)) <> "" And Trim(Me.fg_cri.TextMatrix(i, 6)) <> "Entre" Then
            Whe = Whe & " or " & Me.fg_cri.TextMatrix(i, 1) & "." & Me.fg_cri.TextMatrix(i, 2) & _
            Registro(Trim(Me.fg_cri.TextMatrix(i, 6)), Trim(Me.fg_cri.TextMatrix(i, 9)))
        End If

        If Trim(Me.fg_cri.TextMatrix(i, 10)) <> "" And Trim(Me.fg_cri.TextMatrix(i, 6)) <> "Entre" Then
            Whe = Whe & " or " & Me.fg_cri.TextMatrix(i, 1) & "." & Me.fg_cri.TextMatrix(i, 2) & _
            Registro(Trim(Me.fg_cri.TextMatrix(i, 6)), Trim(Me.fg_cri.TextMatrix(i, 10)))
        End If

        Whe = Whe & ")" and "
    End If

Next i

If Whe <> "" Then
    Whe = " where " & Mid(Whe, 1, Len(Whe) - 5)
End If

If Ord <> "" Then
    Ord = " order by " & Mid(Ord, 1, Len(Ord) - 1)
End If

If Cad <> "" Then
    Cad = Mid(Cad, 1, Len(Cad) - 1)
    Me.Text1.Text = Me.Text1.Text & Cad
End If

Me.Text1.Text = Me.Text1.Text & " from "
Cad = ""
c = fg_rel.Aggregate(flexSTCount, 1, 1, fg_rel.Rows - 1, 1)
If c = 0 Then
    For i = 0 To Me.List1.ListCount - 1
        If Me.List1.Selected(i) Then
            Cad = Cad & " " & Me.List1.List(i) & ","
        End If
    Next i
    If Cad <> "" Then
        Cad = Mid(Cad, 1, Len(Cad) - 1)
    End If
Else
    For i = 1 To c
        If Me.fg_rel.TextMatrix(i, 1) <> "" And Me.fg_rel.TextMatrix(i, 3) <> "" And Me.fg_rel.TextMatrix(i, 4) <> "" And Me.fg_rel.TextMatrix(i, 2) <> "" And Me.fg_rel.TextMatrix(i, 5) <> "" Then
            Cad = Cad & " " & Me.fg_rel.TextMatrix(i, 1) & " " & Me.fg_rel.TextMatrix(i, 3) & " " & Me.fg_rel.TextMatrix(i, 4) & " on " & Me.fg_rel.TextMatrix(i, 2) & "=" & Me.fg_rel.TextMatrix(i, 5)
        End If
    Next i
    If Cad = "" Then
        For i = 0 To Me.List1.ListCount - 1
            If Me.List1.Selected(i) Then
                Cad = Cad & " " & Me.List1.List(i) & ","
            End If
        Next i
    End If
End If

```

```

        If Cad <> "" Then
            Cad = Mid(Cad, 1, Len(Cad) - 1)
        End If
    End If
End If

Me.Text1.Text = Me.Text1 & Cad & Whe & Ord
End Sub

Private Sub fg_rel_BeforeEdit(ByVal Row As Long, ByVal Col As Long, Cancel As Boolean)
    Select Case Col
        Case 2, 5
            If fg_rel.TextMatrix(Row, Col - 1) <> "" Then
                Dim rs As New ADODB.Recordset
                Dim i As Integer
                Dim Cad As String

                Cad = ""

                rs.Open "desc " & fg_rel.TextMatrix(Row, Col - 1), "Provider=MSDASQL.1;Persist Security
Info=False;Extended Properties='DRIVER={MySQL ODBC 3.51 Driver};SERVER=localhost;DATABASE=Codetec
;UID=root;OPTION=3'"

                While Not rs.EOF
                    Cad = Cad & rs!Field & "|"
                    rs.MoveNext
                Wend

                rs.Close

                Me.fg_rel.ColComboList(Col) = Mid(Cad, 1, Len(Cad) - 1)
            End If
        End Select
    End Sub

Private Sub fg_rel_KeyDown(KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyDelete Then
        fg_rel.RemoveItem Me.fg_rel.Row
        Call GenerarSql
    End If
End Sub

Private Sub Form_Load()
    Dim rs As New ADODB.Recordset

    rs.Open "select rel_origen as name from relaciones Union select rel_destino as name from relaciones",
    "Provider=MSDASQL.1;Persist Security Info=False;Extended Properties='DRIVER={MySQL ODBC 3.51
Driver};SERVER=localhost;DATABASE=Codetec;UID=root;OPTION=3'"

    Me.List1.Clear
    While Not rs.EOF
        Me.List1.AddItem rs!Name
        rs.MoveNext
    Wend
    rs.Close

End Sub

Private Sub List1_ItemCheck(Item As Integer)
    Dim i As Integer
    Dim Cad As String
    Dim rs As New ADODB.Recordset
    Dim c As Integer

    Cad = ""
    If Me.List1.Selected(Item) Then
        For i = 0 To Me.List1.ListCount - 1
            If Me.List1.Selected(i) And i <> Item Then
                Cad = Cad & "" & Me.List1.List(i) & ","
            End If
        Next i
        If Cad <> "" Then
            Cad = Mid(Cad, 1, Len(Cad) - 1)
            rs.Open "select rel_destino as name from relaciones where rel_origen = '" & Me.List1.List(Item) & _

```

```

        "" and rel_destino in(" & Cad & ") Union select rel_origen as name from relaciones where rel_destino = "" & _
        Me.List1.List(Item) & "" and rel_origen in(" & Cad & ")", "Provider=MSDASQL.1;Persist Security
Info=False;Extended Properties='DRIVER={MySQL ODBC 3.51 Driver};SERVER=localhost;DATABASE=Codetec
;UID=root ;OPTION=3'"
    If rs.EOF Then
        MsgBox "la tabla seleccionada no posee relacion"
        Me.List1.Selected(Item) = False
    Else
        Cad = ""
        For i = 0 To Me.List1.ListCount - 1
            If Me.List1.Selected(i) And i <> Item Then
                Cad = Cad & "" & Me.List1.List(i) & " "
            End If
        Next i

        If Cad <> "" Then
            Cad = Mid(Cad, 1, Len(Cad) - 1)

            rs.Close

            rs.Open "select * from relaciones where (rel_destino = "" & _
            Me.List1.List(Item) & "" and rel_origen in(" & Cad & ")) or (rel_origen = "" & _
            Me.List1.List(Item) & "" and rel_destino in(" & Cad & "))", "Provider=MSDASQL.1;Persist Security
Info=False;Extended Properties='DRIVER={MySQL ODBC 3.51 Driver};SERVER=localhost;DATABASE=Codetec
;UID=root ;OPTION=3'"
            c = fg_rel.Aggregate(flexSTCount, 1, 1, fg_rel.Rows - 1, 1)
            While Not rs.EOF
                c = c + 1
                Me.fg_rel.TextMatrix(c, 1) = rs!rel_origen
                Me.fg_rel.TextMatrix(c, 2) = rs!rel_llaveorigen
                Me.fg_rel.TextMatrix(c, 3) = "Inner Join"
                Me.fg_rel.TextMatrix(c, 4) = rs!rel_destino
                Me.fg_rel.TextMatrix(c, 5) = rs!rel_llavedestino

                rs.MoveNext
            Wend
        End If
    End If

    c = fg_rel.Aggregate(flexSTCount, 1, 1, fg_rel.Rows - 1, 1)
    i = 1
    While i <= c
        If Me.fg_rel.TextMatrix(i, 1) = Me.List1.List(Item) Or Me.fg_rel.TextMatrix(i, 4) = Me.List1.List(Item) Then
            fg_rel.RemoveItem i
        Else
            i = i + 1
        End If
    Wend
End If

Cad = ""
For i = 0 To Me.List1.ListCount - 1
    If Me.List1.Selected(i) Then
        Cad = Cad & Me.List1.List(i) & "|"
    End If
Next i

If Cad <> "" Then
    Me.fg_rel.ColComboList(1) = Mid(Cad, 1, Len(Cad) - 1)
    Me.fg_rel.ColComboList(4) = Mid(Cad, 1, Len(Cad) - 1)
    Me.fg_cri.ColComboList(1) = Mid(Cad, 1, Len(Cad) - 1)
End If
Call GenerarSql
End Sub

```

Frmgrupos

```
Private Sub cmdasignar_Click()
```

```

If txtcodigo.Text = "" Then
    MsgBox "por favor Escoger Programa", vbInformation
End If
frmofcmatricula.txtcodgrupo = txtcodigo.Text
frmofcmatricula.txtnivel = txtnivel.Text
Unload Me
End Sub

```

```

Private Sub DataGrid1_Click()
txtcodigo.Text = DataGrid1.Columns("Codigo").Text
txtnivel.Text = DataGrid1.Columns("Grupo").Text
End Sub

```

```

Private Sub DataGrid1_RowColChange(LastRow As Variant, ByVal LastCol As Integer)
txtcodigo.Text = DataGrid1.Columns("Codigo").Text
txtnivel.Text = DataGrid1.Columns("Grupo").Text
End Sub

```

Frmimport

```

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "seguridad.chm", &O0, &O0, SW_NORMAL
End Sub

```

```

Private Sub cmdcerrar_Click()
X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
If X = vbYes Then
    Unload Me
End If
End Sub

```

```

Private Sub Command1_Click()
On Error Resume Next
Me.CommonDialog1.Filter = "Datos (*.txt;*.dat)|*.txt;*.dat"
Me.CommonDialog1.FileName = Me.Text1.Text
Me.CommonDialog1.ShowOpen
Me.Text1.Text = Me.CommonDialog1.FileName
End Sub

```

```

Private Sub Command2_Click()
On Error Resume Next
Dim i As Integer
Dim Db As New ADODB.Connection
Db.Open "Provider=MSDASQL.1;Persist Security Info=False:Extended Properties='DRIVER={MySQL ODBC 3.51 Driver};SERVER=localhost;DATABASE=Codetec ;UID=root ;OPTION=3'"

```

```

Dim dirarc As String
dirarc = Replace(Me.Text1.Text, "\", "/")

```

```

For i = 0 To Me.List1.ListCount - 1
    If Me.List1.Selected(i) = True Then
        Db.Execute "LOAD DATA INFILE '" & dirarc & "'ignore INTO TABLE " & Me.List1.List(List1.ListIndex)
    End If
Next i
MsgBox "El proceso de Importación se realizó exitosamente", vbInformation, "Importación Exitosa"
SaveSetting "CODETEC", "HOME", "IMPORT", Me.Text1.Text
Db.Close
End Sub

```

```

Private Sub Form_Load()
Me.Text1.Text = GetSetting("CODETEC", "HOME", "IMPORT", "C:\mysql\data\import.txt")

```

```

Dim rs As New ADODB.Recordset

```

```

rs.Open "select rel_origen as name from relaciones Union select rel_destino as name from relaciones",
"Provider=MSDASQL.1;Persist Security Info=False:Extended Properties='DRIVER={MySQL ODBC 3.51 Driver};SERVER=localhost;DATABASE=Codetec ;UID=root ;OPTION=3'"

```

```

Me.List1.Clear

```



```

While Not rs.EOF
    Me.List1.AddItem rs!Name
    rs.MoveNext
Wend
rs.Close

```

```
End Sub
```

Frminicio

```

Dim i As Integer
Private Sub Form_Load()
    i = 0
    Timer1.Enabled = True
End Sub
Private Sub Timer1_Timer()
    i = i + 1
    ProgressBar1.Value = i
    If i = 150 Then
        Unload Me
        frmlogin.Show vbModal
        Timer1.Enabled = False
    End If
End Sub

```

Frminscripcion

```

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
    ShellExecute frmprincipal.hwnd, "Open", "C:\Documents and Settings\Xpsp2\Escritorio\Proyecto Completo\Backup_completp_codetec-12-11-2007-1019PM\Aplicacion_CODETEC\ayuda_web\parametrizacion.mht", &O0, &O0, SW_NORMAL
End Sub

```

```

Private Sub cdmlegalizar_Click()
    Dim bus As String
    bus = InputBox("Digite el NUMero del Volante", "Buscar Volante")
    If bus <> vbNullString Then
        adolega.Recordset.MoveFirst
        adolega.Recordset.Find ("vol_numero=" & bus & "")
        If adolega.Recordset.EOF Then
            MsgBox "Verifique La cedula del candidato", vbInformation + vbOKOnly, "Cedula Incorrecta"
            adolega.Recordset.MoveFirst
        End If
    Else
        MsgBox "Vuelva a intentar la busqueda - Cedula Nula", vbInformation + vbOKCancel, "Error de Busqueda"
    End If
End Sub

```

```
End Sub
```

```

Private Sub cmdbuspro_Click()
    frmminiprograma.Show vbModal
    cmdvolante.Enabled = True
End Sub

```

```

Private Sub cmdcerrar_Click()
    X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
    If X = vbYes Then
        Unload Me
    End If
End Sub

```

```

Public Sub LimpiarControles(frmminscripcion As Form)
    Dim ctlControl As Object

```

```

On Error Resume Next
For Each ctlControl In frminscripcion.Controls
    ctlControl.Text = ""
    ctlControl.ListIndex = -1
DoEvents
Next ctlControl
End Sub

```

```

Private Sub cmdvolante_Click()
Do
    If txtvalor.Text = "" Then
        txtvalor.Text = InputBox("Debe Asignar un valor a la inscripcion", "Valor inscripcion")
    End If
    Loop Until Not (txtvalor.Text = "")
    ado_volante.Recordset.AddNew
    If txtnumero.Text = "" Then
        txtvolante.Text = 1000
    Else
        txtvolante.Text = txtnumero.Text
    End If
    txtinte.Text = txtcodigo.Text + "INS" + txtvolante.Text
    txtprog.Text = txtcodigo.Text
    vol_estado.Text = "Tramite"
    Vol_tipo.Text = "Inscripcion"
    Text3.Text = txtcedula1.Text
    txtvalor2.Text = txtvalor.Text
    X = "El numero de Volante de inscripción generado es " + txtinte.Text
    MsgBox X

```

```

p = txtprog.Text
i = txtinte.Text
v = txtvalor2.Text
s = "Primero"
c = Text3.Text
ado_volante.Recordset.Update

```

```

vol = "select c.can_cedula,c.can_nombres,c.can_apellidos.vol_inte,vol_semestre,vol_valor,p.pro_nombre from
volantes,candidato c,programa p where '" & c & "' = can_cedula and vol_tipo = 'Inscripción' and vol_inte = '" & i & "' and
vol_semestre = '" & s & "' and vol_valor = '" & v & "' and vol_programa = '" & p & "' and vol_programa = p.pro_codigo"
rpt_volantes.Commands(1).CommandText = vol
rpt_ins.Show vbModal

```

```

'ado_volante.Recordset.MoveFirst
txtvalor.Enabled = False
cmdbuspro.Enabled = False
cmdvolante.Enabled = False
cmdcerrar.SetFocus
End Sub

```

Frmlistadep

```

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation
As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As
Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "reportes.chm", &O0, &O0, SW_NORMAL
End Sub

```

```

Private Sub cmdcerrar_Click()
X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
If X = vbYes Then
    Unload Me
End If
End Sub

```

```

Private Sub DataGrid1_Click()
Dim s As String
If txtprograma.Text <> "" Then
    s = "select pro_nombre from programa where pro_codigo = " + txtprograma.Text
    ado_programa.RecordSource = s
    ado_programa.Refresh
    Label2.Caption = txtnom.Text

```

```

Label2.Visible = True
Label3.Visible = True
Else
Label2.Visible = False
Label3.Visible = False
End If
End Sub

```

Frmlogin

```

Private Sub cmdcancelar_Click()
txtusuario.Text = ""
txtcontrasena.Text = ""
txtusuario.SetFocus
End Sub

```

```

Private Sub cmdcontrasena_Click()
frmambio.Show vbModal
End Sub

```

```

Private Sub cmdingresar_Click()

```

```

'c = "select usu_nomusu,usu_contrasena from usuario where " + txtusuario.Text + "= usu_nomusu and" +
txtcontrasena.Text + "= usu_contrasena"
'ado_clave.RecordSource = c
'ado_clave.Refresh
Dim Db As ADODB.Connection
Set Db = New ADODB.Connection
Dim rs As ADODB.Recordset
Set rs = New ADODB.Recordset
Db.Open "Provider=MSDASQL.1;Persist Security Info=False;Extended Properties='DRIVER={MySQL ODBC 3.51
Driver};SERVER=localhost;DATABASE=Codetec ;UID=root ;OPTION=3'"
rs.Open "select usu_contrasena as name from usuario where " & txtusuario.Text & "= usu_nomusu and password (" &
txtcontrasena.Text & ") = usu_contrasena ", Db, adOpenDynamic, adLockOptimistic
If Not rs.EOF Then
List1.AddItem rs!Name
End If
rs.Close
Db.Close
'If txtusu.Text = "" Or txtclave.Text = "" Then
If List1.List(0) = "" Then
MsgBox "por favor verifique su nombre de usuario y contraseña"
txtusuario.SetFocus
Else

```

```

c2 = "select * from perfil_permiso,usuario where " + txtusu.Text + "= usu_nomusu and usu_perfil = cod_perfil "
ado_bus.RecordSource = c2
ado_bus.Refresh
'este grupo de instrucciones me validan cuando tengo acceso a todos los permisos o solo a algunos
If (txtca.Text = "0") And (txtes.Text = "0") And (txtdo.Text = "0") And (txtpro.Text = "0") And (txtasi.Text = "0") And
(txtpen.Text = "0") And (xtasig.Text = "0") Then
frmprincipal.macademica.Visible = False
Else
If txtca.Text = "0" Then
frmprincipal.smcandidato.Visible = False
End If
If txtes.Text = "0" Then
frmprincipal.smestudiante.Visible = False
End If
If txtdo.Text = "0" Then
frmprincipal.mmdoc.Visible = False
End If
If txtpro.Text = "0" Then
frmprincipal.smprog.Visible = False
End If
If txtasi.Text = "0" Then
frmprincipal.massignaturam.Visible = False
End If
If txtpen.Text = "0" Then
frmprincipal.smpensum.Visible = False
End If
If xtasig.Text = "0" Then
frmprincipal.mnugru.Visible = False

```

```

    End If
End If
If (txtmat.Text = "0") And (txtn.Text = "0") Then
    frmprincipal.mprocesos.Visible = False
Else
    If txtmat.Text = "0" Then
        frmprincipal.mmatri.Visible = False
    End If
    If txtn.Text = "0" Then
        frmprincipal.mmmmnot.Visible = False
    End If
End If
If (txtins.Text = "0") And (txtmatri.Text = "0") Then
    frmprincipal.mpagos.Visible = False
Else
    If txtins.Text = "0" Then
        frmprincipal.mnupa.Visible = False
    End If
    If txtmatri.Text = "0" Then
        frmprincipal.smpago.Visible = False
    End If
End If
If (txtlisd.Text = "0") And (txtpensu.Text = "0") And (txtvol.Text = "0") And (txtasigna.Text = "0") And (txtestud.Text = "0") And (txtcand.Text = "0") And (txtgenerador.Text = "0") Then
    frmprincipal.mreportes.Visible = False
Else
    If txtlisd.Text = "0" Then
        frmprincipal.mnudepl.Enabled = False
    End If
    If txtpensu.Text = "0" Then
        frmprincipal.smpensum.Visible = False
    End If
    If txtvol.Text = "0" Then
        frmprincipal.nmuvolcan.Visible = False
    End If
    If txtasigna.Text = "0" Then
        frmprincipal.pa.Visible = False
    End If
    If txtestud.Text = "0" Then
        frmprincipal.smestu.Visible = False
    End If
    If txtcand.Text = "0" Then
        frmprincipal.smcandi.Visible = False
    End If
    If txtgenerador.Text = "0" Then
        frmprincipal.smdina.Visible = False
    End If
End If

If txtusua.Text = "0" Then
    frmprincipal.madmin.Visible = False
End If

If (txtlog.Text = "0") And (txtbackup.Text = "0") And (txtoperaciones.Text = "0") Then
    frmprincipal.msegu.Visible = False
Else
    If txtlog.Text = "0" Then
        frmprincipal.msegu.Visible = False
    End If
    If txtbackup.Text = "0" Then
        frmprincipal.mbackup.Visible = False
    End If
    If txtoperaciones.Text = "0" Then
        frmprincipal.smoper.Visible = False
    End If
End If
frmprincipal.Label4.Caption = txtusuario.Text
ado_auditor.Recordset.AddNew
txtidenti.Text = txtusuario.Text
txtoperacion.Text = "Ingreso al sistema"
txtuser.Text = txtusuario.Text
txtfecha.Text = Format(Date, "dd-mm-yyyy")
txthora.Text = Format(Now, "h:nn:ss")
txtmodulo.Text = "Inicio de Sesión"
ado_auditor.Recordset.Update

```

```

        Unload Me
        frmprincipal.Show vbModal
    End If
End Sub

Private Sub Cmdsalir_Click()
    c = MsgBox("¿Seguro que desea Salir del sistema?", vbYesNo + vbInformation, "Salir del Sistema")
    If c = vbYes Then
        End
    End If
End Sub

Private Sub Form_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then
        SendKeys "{Tab}"
    End If
End Sub

Private Sub Form_Load()
    If txtcedula.Text = "" Then
        ado_perfil.Recordset.AddNew
        txtperfil.Text = "100"
        txtnombres.Text = "ADMINISTRADOR"
        ado_perfil.Recordset.Update
        ado_perfil.Recordset.MoveFirst

        ado_bus.Recordset.AddNew
        txtperfil2.Text = "100"
        txtca.Text = "1"
        txtes.Text = "1"
        txtdo.Text = "1"
        txtpro.Text = "1"
        txtasi.Text = "1"
        txtpen.Text = "1"
        txtasig.Text = "1"
        txtmat.Text = "1"
        txtn.Text = "1"
        txtins.Text = "1"
        txtmatri.Text = "1"
        txtlisd.Text = "1"
        txtpensu.Text = "1"
        txtvol.Text = "1"
        txtasigna.Text = "1"
        txtstud.Text = "1"
        txtcand.Text = "1"
        txtusua.Text = "1"
        txtlog.Text = "1"
        ado_bus.Recordset.Update
        ado_bus.Refresh

        Dim con As New ADODB.Connection
        con.Open "Provider=MSDASQL.1;Persist Security Info=False;Extended Properties='DRIVER={MySQL ODBC 3.51 Driver};SERVER=localhost;DATABASE=Codetec ;UID=root ;OPTION=3'"

        con.Execute "insert into usuario values('99999',' ','100', password(123456),'',ADMINISTRADOR)"
        con.Close
        *   ado_clave.Recordset.AddNew
        *   txtusu.Text = "ADMINISTRADOR"
        *   txtclave.Text = "123456"
        *   txtper.Text = "100"

        *   ado_clave.Recordset.Update
        *   ado_clave.Refresh

        MsgBox "No existen usuarios registrados - Su usuario por defecto es: ADMINISTRADOR - Contraseña: 123456",
        vbInformation, "Usuario Inicial"
        Unload Me
        frmlogin.Show vbModal
    End If

End Sub

Private Sub txtusuario_Change()
    Dim i As Integer
    txtusuario.Text = UCase(txtusuario.Text)

```

```

i = Len(txtusuario.Text)
txtusuario.SelStart = i
End Sub

```

```

Private Sub txtusuario_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
    If txtusuario.Text = "" Then
        MsgBox "Debe ingresar un nombre de usuario", vbCritical, "Error de entrada"
        txtusuario.SetFocus
    End If
End If
End Sub

```

```

Private Sub txtcontrasena_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
    If txtcontrasena.Text = "" Then
        MsgBox "Debe ingresar una contraseña de usuario", vbCritical, "Error de entrada"
        txtcontrasena.SetFocus
    End If
End If
End Sub

```

Frmlogs

```

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation
As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As
Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "seguridad.chm", &O0, &O0, SW_NORMAL
End Sub

```

```

Private Sub Check1_Click()
If Check1.Value = 0 Then
    txtfecha1.Enabled = False
    txtfecha2.Enabled = False
Else
    txtfecha1.Enabled = True
    txtfecha2.Enabled = True
End If
End Sub

```

```

Private Sub Check2_Click()
If Check2.Value = 0 Then
    combo_usuarios.Enabled = False
Else
    combo_usuarios.Enabled = True
End If
End Sub
Private Sub Check3_Click()
If Check3.Value = 0 Then
    Me.txtoperacion.Enabled = False
Else
    Me.txtoperacion.Enabled = True
End If
End Sub

```

```

Private Sub cmdbuscar_Click()
If Check1.Value = 1 And Check2.Value = 0 And Check3.Value = 0 Then
    con1 = "Select * from auditor where aud_fecha between '" + txtfecha1.Text + "' and '" + txtfecha2.Text + "'"
    ado_auditoria.RecordSource = con1
    ado_auditoria.Refresh
End If

```

```

If Check1.Value = 1 And Check2.Value = 1 And Check3.Value = 0 Then
    con2 = "Select * from auditor where aud_fecha between '" + txtfecha1.Text + "' and '" + txtfecha2.Text + "' and
aud_usuario = '" + combo_usuarios.Text + "'"
    ado_auditoria.RecordSource = con2
    ado_auditoria.Refresh
End If

```

```

If Check1.Value = 1 And Check2.Value = 0 And Check3.Value = 1 Then

```

```

con3 = "Select * from auditor where aud_fecha between '" + txtfecha1.Text + "'and'" + txtfecha2.Text + "' and
aud_operacion = '" + txtoperacion.Text + "'"
ado_auditoria.RecordSource = con3
ado_auditoria.Refresh
End If

```

```

If Check1.Value = 0 And Check2.Value = 1 And Check3.Value = 0 Then
con4 = "Select * from auditor where '" + combo_usuarios.Text + "' = aud_usuario"
ado_auditoria.RecordSource = con4
ado_auditoria.Refresh
End If

```

```

If Check1.Value = 0 And Check2.Value = 1 And Check3.Value = 1 Then
con5 = "Select * from auditor where '" + combo_usuarios.Text + "' = aud_usuario and aud_operacion = '" +
txtoperacion.Text + "'"
ado_auditoria.RecordSource = con5
ado_auditoria.Refresh
End If

```

```

If Check1.Value = 0 And Check2.Value = 0 And Check3.Value = 1 Then
con6 = "Select * from auditor where aud_operacion = '" + txtoperacion.Text + "'"
ado_auditoria.RecordSource = con6
ado_auditoria.Refresh
End If

```

```

If Check1.Value = 1 And Check2.Value = 1 And Check3.Value = 1 Then
con7 = "Select * from auditor where aud_fecha between '" + txtfecha1.Text + "'and'" + txtfecha2.Text + "' and
aud_operacion = '" + txtoperacion.Text + "' and '" + combo_usuarios.Text + "' = aud_usuario and aud_operacion = '" +
txtoperacion.Text + "'"
ado_auditoria.RecordSource = con7
ado_auditoria.Refresh
End If

```

```

If Check1.Value = 0 And Check2.Value = 0 And Check3.Value = 0 Then
MsgBox "Debe seleccionar un Criterio de Búsqueda", vbInformation + vbCritical, "Error información inválida"
End If

```

End Sub

```

Private Sub cmdcerrar_Click()
X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
If X = vbYes Then
Unload Me
End If
End Sub

```

Frmmatricula

```

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation
As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As
Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "academicos.chm", &O0, &O0, SW_NORMAL
End Sub

```

```

Private Sub cmdbuspro_Click()
frmminiprograma.cmdasignar.Caption = "&Asignar a matricula"
frmminiprograma.Show vbModal
End Sub

```

```

Private Sub cmdbuscar_Click()
Dim s As String
s = "select c.can_nombres,c.can_apellidos,p.pro_nombre,p.pro_valor,i.ins_codprograma from inscripcion i,programa
p,candidato c where '" + txtcedula.Text + "' = i.ins_idcandidato and i.ins_idcandidato = c.can_cedula and
i.ins_codprograma = p.pro_codigo "

```



```

ado_inscripcion.RecordSource = s
ado_inscripcion.Refresh
Label1.Caption = txtnombre.Text + " " + txtapellidos.Text
If Label1.Caption = " " Then
    cmdvolante.Enabled = False
    cmdnuevo.Enabled = False
    MsgBox "Registro no encontrado", vbCritical
Else
    cmdvolante.Enabled = True
    cmdnuevo.Enabled = True
End If
End Sub

```

```

Private Sub cmdcerrar_Click()
X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
If X = vbYes Then
    Unload Me
End If
End Sub

```

```

Private Sub cmdnuevo_Click()
frmminiprograma.txtcedula = txtcedula.Text
frmminiprograma.cmdasignar.Caption = "Registrar Nuevo Programa"
frmminiprograma.Show vbModal
End Sub

```

```

Private Sub cmdvolante_Click()
Dim consul As String
If txtsemestre.Text = "" Then
    ado_volante.Recordset.AddNew
    txtcodpro2.Text = txtcodpro.Text
    txtcedula2.Text = txtcedula.Text
    txtvalor2.Text = txtvalor.Text
    txttipo.Text = "Matricula"
    txtestado.Text = "Tramite"
    If txtnumero.Text = "" Then
        txtvolante.Text = 10000
    Else
        ado_numero.Refresh
        txtvolante.Text = txtnumero.Text
    End If

    consul = "Select vol_cedula,vol_programa,vol_semestre,vol_tipo from volantes where "" + txtcedula2.Text + "" = " & _
vol_cedula and "" + txtcodpro2.Text + "" = vol_programa and "" + txttipo.Text + "" = vol_tipo"
    ado.RecordSource = consul
    ado.Refresh

    If txtcedula2.Text = tced.Text And txtcodpro2.Text = tpro.Text And txttipo.Text = ttipo.Text Then
        MsgBox "El volante para este Programa ya fue generado", vbCritical, "Error de Generación de Volante"
        ado_volante.Recordset.CancelUpdate
    Else
        txtinte.Text = txtcodpro2.Text + "MAT" + txtvolante.Text
        X = "El numero de Volante de matricula generado es " + txtinte.Text
        MsgBox X

        ado_auditor.Recordset.AddNew
        txtiden2.Text = txtinte.Text
        txtidenti.Text = txtcedula.Text
        txtident3.Text = txtcodpro.Text
        txtoperacion.Text = "Volante Generado"
        txtusu.Text = frmprincipal.Label4
        txtfecha.Text = Format(Date, "dd-mm-yyyy")
        txthora.Text = Format(Now, "h:nn:ss")
        txtmodulo.Text = "Generar Volante Matricula"
        ado_auditor.Recordset.Update

        i = txtinte.Text
        p = txtcodpro2.Text
        c = txtcedula2.Text
        v = txtvalor2.Text
        s = "Primero"

        ado_volante.Recordset.Update
    End If
End Sub

```



```

        vol = "select c.can_cedula,c.can_nombres,c.can_apellidos,vol_inte,vol_semestre,vol_valor,p.pro_nombre from
volantes,candidato c,programa p where can_cedula = '" & c & "' and vol_tipo = 'Matricula' and vol_inte = '" & i & "' and
vol_semestre = '" & s & "' and vol_valor = '" & v & "' and vol_programa = '" & p & "' and vol_programa = p.pro_codigo"
        rpt_volantes.Commands(1).CommandText = vol
        rpt_volante.Show vbModal

```

```

        Unload Me
        frmmatricula.Show vbModal
        v = MsgBox("Desea Generar un nuevo volante para otro programa", vbYesNo, "Nueva Matrícula")
        If v = vbNo Then
            ado_volante.Recordset.Update
            Unload Me
            frmmatricula.Show vbModal
        Else
            ado_volante.Recordset.Update
        End If
    End If
End If

```

```

Else
    ado_volante.Recordset.AddNew
    txtcodpro2.Text = txtprograma.Text
    txtcedula2.Text = txtcedula.Text
    txtvalor2.Text = DataGrid2.Columns("Valor").Text
    txtsem.Text = txtsemestre.Text
    txttipo.Text = "Matricula"
    txtestado.Text = "Tramite"
    If txtnumero.Text = "" Then
        txtvolante.Text = 10000
    Else
        ado_numero.Refresh
        txtvolante.Text = txtnumero.Text
    End If
    txtinte.Text = txtcodpro2.Text + "MAT" + txtvolante.Text
    X = "El numero de Volante de matricula generado es " + txtinte.Text
    MsgBox X

```

```

        ado_auditor.Recordset.AddNew
        txtiden2.Text = txtinte.Text
        txtidenti.Text = txtcedula.Text
        txtident3.Text = txtcodpro.Text
        txtoperacion.Text = "Volante Generado"
        txtusu.Text = frmprincipal.Label4
        txtfecha.Text = Format(Date, "dd-mm-yyyy")
        txthora.Text = Format(Now, "h:nn:ss")
        txtmodulo.Text = "Generar Volante Matricula"
        ado_auditor.Recordset.Update

```

```

        ado_volante.Recordset.Update

```

```

        Unload Me
        frmmatricula.Show vbModal
    End If
End Sub

```

```

Private Sub Command1_Click()
    Text1.Text = Format(Date, "dd-mm-yyyy")
    Text2.Text = Format(Now, "h:nn:ss AM/PM")
End Sub

```

```

Private Sub txtcedula_KeyPress(KeyAscii As Integer)
    Dim s As String
    If KeyAscii = 13 Then

```

```

s = "select c.can_nombres,c.can_apellidos,p.pro_nombre,p.pro_valor,i.ins_codprograma from inscripcion
i,programa p,candidato c where " + txtcedula.Text + " = i.ins_idcandidato and i.ins_idcandidato = c.can_cedula and
i.ins_codprograma = p.pro_codigo "
ado_inscripcion.RecordSource = s
ado_inscripcion.Refresh
Label1.Caption = txtnombre.Text + " " + txtapellidos.Text
If Label1.Caption = " " Then
cmdvolante.Enabled = False
cmdnuevo.Enabled = False
MsgBox "Registro no encontrado", vbCritical
Else
cmdvolante.Enabled = True
cmdnuevo.Enabled = True
End If
End If
End Sub

```

Frmminiprograma

```

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation
As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As
Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "C:\Documents and Settings\Windows Xpsp2\Escritorio\Proyecto
Completo\Backup_completo_codetec-12-11-2007-1019PM\Aplicacion_CODETEC\ayuda_web\parametrizacion.mht",
&O0, &O0, SW_NORMAL
End Sub
Private Sub cmdasignar_Click()
Dim a, s As String
If cmdasignar.Caption = "Registrar Nuevo Programa" Then
a = txtcedula.Text
ado_ins.Recordset.AddNew
txtprog.Text = txtcodigo.Text
txtcedula.Text = a
ado_ins.Recordset.Update
frmmatricula.txtcedula.Text = txtcedula.Text
s = "select c.can_nombres,c.can_apellidos,p.pro_nombre,p.pro_valor,i.ins_codprograma from inscripcion i,programa
p,candidato c where " + frmmatricula.txtcedula.Text + " = i.ins_idcandidato and i.ins_idcandidato = c.can_cedula and
i.ins_codprograma = p.pro_codigo"
frmmatricula.ado_inscripcion.RecordSource = s
frmmatricula.ado_inscripcion.Refresh
frmmatricula.Label1.Caption = frmmatricula.txtnombre.Text + " " + frmmatricula.txtapellidos.Text
Unload Me
End If
If cmdasignar.Caption = "&Asignar a Inscripción" Then
frminscripcion.txtcodigo = txtcodigo.Text
frminscripcion.txtnompro = txtnombre.Text
Unload Me
frminscripcion.cmdbuspro.Enabled = True
frminscripcion.cmdcerrar.Enabled = True
frminscripcion.txtvalor.Text = 20000
frminscripcion.txtvalor.Enabled = False
End If
If cmdasignar.Caption = "&Asignar" Then
frmpensum.txtprogama.Text = txtnombre.Text
frmpensum.tem.Text = txtcodigo.Text
frmpensum.txtcodpro.Text = txtcodigo.Text
frmpensum.DataGrid1.Enabled = True
frmpensum.cmdagregar.Enabled = True
frmpensum.DataGrid1.Enabled = False

' s = "select pen_codprograma from pensum where "" + txtcodigo.Text + "" = pen_codprograma"
' bus.RecordSource = s
' bus.Refresh

' If txtbus.Text <> "" Then
' s = "select distinct a.asi_codigo,a.asi_nombre,a.asi_semestre from asignatura a,pensum p where "" +
txtcodigo.Text + "" = p.pen_codprograma and a.asi_codigo <> p.pen_codasignatura"
' frmpensum.ado_asignatura.RecordSource = s
' frmpensum.ado_asignatura.Refresh
' Else
s = "select a.asi_codigo,a.asi_nombre,a.asi_semestre from asignatura a"

```

```

    frmpensum.ado_asignatura.RecordSource = s
    frmpensum.ado_asignatura.Refresh
End If
MsgBox "Al descargar esta ventana debe activar la asignacion de asignaturas"
Unload Me
End If
If cmdasignar.Caption = "&Listar Pensum" Then
    If frmconpen.txtx.Text = "" Then
        frmconpen.ado_consulta.Recordset.AddNew
    End If
    frmconpen.txtx = txtcodigo.Text
    frmconpen.Label1.Visible = True
    frmconpen.Label1.Caption = txtnombre.Text
    frmconpen.ado_consulta.Recordset.Update
    frmconpen.ado_consulta.Recordset.MoveFirst
    frmconpen.ado_pensum.Refresh
    Unload Me
    frmconpen.Show vbModal
End If
If cmdasignar.Caption = "Asignacion dependencia" Then
    frmassignacion.txtcodigo.Text = txtcodigo.Text
    frmassignacion.txtprograma.Text = txtnombre.Text
    Unload Me
End If
If cmdasignar.Caption = "&Asignar a matricula" Then
    Unload Me
End If
If cmdasignar.Caption = "Matricula" Then
    frmofcmatricula.txtprograma2.Text = txtnombre.Text
    frmofcmatricula.txtcodprograma2.Text = txtcodigo.Text
    frmofcmatricula.txtsemestre.Text = frmofcmatricula.Combo1.Text
    s = "select p.pen_inteligente,a.asi_nombre,a.asi_semestre from asignatura a.pensum p where" +
    frmofcmatricula.txtcodprograma2.Text + "=" + p.pen_codprograma and p.pen_codasignatura = a.asi_codigo and
    a.asi_semestre =" + frmofcmatricula.txtsemestre.Text + ""
    frmofcmatricula.ado_pensum.RecordSource = s
    frmofcmatricula.ado_pensum.Refresh
    frmofcmatricula.Limpiar
    p = frmofcmatricula.DataGrid1.VisibleRows
    frmofcmatricula.txtconse.Text = "MAT" + frmofcmatricula.txtcodprograma2.Text + frmofcmatricula.txtnumero.Text
For i = 0 To p
    N = frmofcmatricula.DataGrid1.Bookmark
    If N = 1 Then
        frmofcmatricula.m1.Text = frmofcmatricula.DataGrid1.Columns("CODIGO").Text
    End If
    If N = 2 Then
        frmofcmatricula.m2.Text = frmofcmatricula.DataGrid1.Columns("CODIGO").Text
    End If
    If N = 3 Then
        frmofcmatricula.m3.Text = frmofcmatricula.DataGrid1.Columns("CODIGO").Text
    End If
    If N = 4 Then
        frmofcmatricula.m4.Text = frmofcmatricula.DataGrid1.Columns("CODIGO").Text
    End If
    If N = 5 Then
        frmofcmatricula.m5.Text = frmofcmatricula.DataGrid1.Columns("CODIGO").Text
    End If
    If N = 6 Then
        frmofcmatricula.m6.Text = frmofcmatricula.DataGrid1.Columns("CODIGO").Text
    End If
    If N = 7 Then
        frmofcmatricula.m7.Text = frmofcmatricula.DataGrid1.Columns("CODIGO").Text
    End If
    If N = 8 Then
        frmofcmatricula.m8.Text = frmofcmatricula.DataGrid1.Columns("CODIGO").Text
    End If
    If N = 9 Then
        frmofcmatricula.m9.Text = frmofcmatricula.DataGrid1.Columns("CODIGO").Text
    End If
    If N = 10 Then
        frmofcmatricula.m10.Text = frmofcmatricula.DataGrid1.Columns("CODIGO").Text
    End If
    If frmofcmatricula.DataGrid1.Bookmark < p Then
        frmofcmatricula.DataGrid1.Bookmark = frmofcmatricula.DataGrid1.Bookmark + 1
    End If
Next
Unload Me

```

```

End If
If cmdasignar.Caption = "Asignar Apertura" Then
    frmapartar.Label2.Caption = txtnombre.Text
    frmapartar.txtcodigo.Text = txtcodigo.Text
    s = "select p.pen_inteligente,a.asi_nombre from asignatura a,pensum p where '" + txtcodigo.Text + "' =
p.pen_codoprograma and p.pen_codasignatura = a.asi_codigo and '" + frmapartar.txtsemestre.Text + "' =
a.asi_semestre"
    frmapartar.ado_consulta.RecordSource = s
    frmapartar.ado_consulta.Refresh
    frmapartar.txtinteligente.Text = frmapartar.txtcodigo.Text + frmapartar.txtsemestre.Text
    s2 = "select doc_identi,doc_nombres,doc_apellidos from docente"
    frmapartar.ado_profesores.RecordSource = s2
    frmapartar.ado_profesores.Refresh
    MsgBox "Se ha asignado el Programa Académico, culmine el proceso de Apartar grupos y presione la opción
Asignación", vbInformation, "Asignación Exitosa"
    Unload Me
End If
If cmdasignar.Caption = "Asignar notas" Then
    frmnotdoc.txtcodigo.Text = txtcodigo.Text
    frmnotdoc.Label3.Caption = txtnombre.Text
    Unload Me
End If
If cmdasignar.Caption = "&Asignar Matriculados" Then
    frmmatriculados.txtcodigo.Text = txtcodigo.Text
    frmmatriculados.Label3.Caption = txtnombre.Text
    Unload Me
End If
End Sub

Private Sub cmdbuscar_Click()
Dim bus As String
bus = InputBox("Digite el codigo del programa", "Buscar programa")
If bus <> vbNullString Then
    ado_programa.Recordset.MoveFirst
    ado_programa.Recordset.Find ("pro_codigo='" & bus & "'")
    If ado_programa.Recordset.EOF Then
        MsgBox "Verifique el codigo del programa", vbInformation + vbOKOnly, "Código Incorrecto"
        ado_programa.Recordset.MoveFirst
    End If
Else
    MsgBox "Vuelva a intentar la busqueda - Código Nula", vbInformation + vbOKCancel, "Error de Busqueda"
End If
End Sub

Private Sub cmdcerrar_Click()
frmpensum.DataGrid1.Enabled = False
frmpensum.ado_pensum.Recordset.CancelUpdate
Unload Me
End Sub

```

Frmnotdoc

```

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation
As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As
Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "academicos.chm", &O0, &O0, SW_NORMAL
End Sub

Private Sub cmdasignatura_Click()
s = "select a.asi_nombre,p.pen_inteligente,g.gru_inteligente from grupos g,pensum p,asignatura a where'" +
txtcedula.Text + "'=g.gru_docente and '" + txtcodigo.Text + "'= g.gru_programa and g.gru_asignatura =
p.pen_inteligente and p.pen_codasignatura = a.asi_codigo and'" + Combo1.Text + "' = g.gru_semestre and'" +
Combo2.Text + "' = gru_grupo"
frmconasig.ado_asignatura.RecordSource = s
frmconasig.ado_asignatura.Refresh
frmconasig.Show vbModal
End Sub

Private Sub cmdcerrar_Click()
X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")

```

```

If X = vbYes Then
    Unload Me
End If
End Sub

```

```

Private Sub cmdnotas_Click()
'On Error Resume Next
' Dim Rs As New ADODB.Recordset
' Rs.Open "insert into
asignatura_estudiante(ases_codasignatura,ases_idestudiante,ases_nombre,ases_apellido,ases_nota1,ases_nota2,ases_nota3,ases_notafinal,ases_usuario,ases_matricula) " & _
' "select "" & txtcodasi.Text & "" as aa, e.est_identi,e.est_nombres,e.est_apellidos,0 as bb,0 as cc,0 as dd,0 as ee," as
ff,m.mat_inte from estudiante e,matricula m,mat_asignatura ma where "" & _
' txtcodgrupo.Text & ""= m.mat_codgrupo and m.mat_inte = ma.Cod_mat and "" & txtcodigo.Text & ""= mat_programa
and m.mat_idestudiante = e.est_identi and (" & txtcodasi.Text & _
' "" = ma.Cod_m1 or "" & txtcodasi.Text & ""= ma.Cod_m2 or "" & txtcodasi.Text & ""= ma.Cod_m3 or "" & txtcodasi.Text
+ _
' ""=ma.Cod_m4 or "" & txtcodasi.Text & ""=ma.Cod_m5 or "" & txtcodasi.Text & ""=ma.Cod_m6 or "" & txtcodasi.Text & _
' ""=ma.Cod_m7 or "" & txtcodasi.Text & ""=ma.Cod_m8 or "" & txtcodasi.Text & ""=ma.Cod_m9 or "" & txtcodasi.Text & _
' ""= ma.Cod_m10)" & _
' "select * from asignatura_estudiante as es where es.ases_codasignatura = "" & txtcodasi.Text & "" and
es.ases_matricula=ma.Cod_mat)", "Provider=MSDASQL.1;Persist Security Info=False;Extended
Properties='DRIVER={MySQL ODBC 3.51 Driver};SERVER=localhost;DATABASE=Codetec;UID=root ;OPTION=3'"
' s1 = "select e.est_identi,e.est_nombres,e.est_apellidos,m.mat_inte,asi.ases_nota1 from asignatura_estudiante
asi,estudiante e,matricula m,mat_asignatura ma where "" & txtcodgrupo.Text & ""= m.mat_codgrupo and m.mat_inte =
ma.Cod_mat and "" & txtcodigo.Text & ""= mat_programa and m.mat_idestudiante = e.est_identi and (" & txtcodasi.Text
+ "" = ma.Cod_m1 or "" & txtcodasi.Text & ""= ma.Cod_m2 or "" & txtcodasi.Text & ""= ma.Cod_m3 or "" & txtcodasi.Text &
""=ma.Cod_m4 or "" & txtcodasi.Text & ""=ma.Cod_m5 or "" & txtcodasi.Text & ""=ma.Cod_m6 or "" & txtcodasi.Text &
""=ma.Cod_m7 or "" & txtcodasi.Text & ""=ma.Cod_m8 or "" & txtcodasi.Text & ""=ma.Cod_m9 or "" & txtcodasi.Text & ""=
ma.Cod_m10)" & _
' "select * from asignatura_estudiante a
' a.ases_nota1,a.ases_nota2,a.ases_nota3
' "" & txtcodgrupo.Text & ""= mat_codgrupo and
' s1 = "select e.est_nombres,e.est_apellidos from estudiante e,matricula m,mat_asignatura ma where m.mat_inte =
ma.Cod_mat and (" & txtcodasi.Text & "" = ma.Cod_m1 or "" & txtcodasi.Text & ""= ma.Cod_m2 or "" & txtcodasi.Text &
"" = ma.Cod_m3 or "" & txtcodasi.Text & ""= ma.Cod_m4 or "" & txtcodasi.Text & ""= ma.Cod_m5 or "" & txtcodasi.Text &
""=ma.Cod_m6 or "" & txtcodasi.Text & ""=ma.Cod_m7 or "" & txtcodasi.Text & ""=ma.Cod_m8 or "" & txtcodasi.Text &
""=ma.Cod_m9 or "" & txtcodasi.Text & ""=ma.Cod_m10)"
' adon.RecordSource = s1
' adon.Refresh
'
' s3 = "select ae.* from asignatura_estudiante ae where "" & txtcodgrupo.Text & ""= m.mat_codgrupo and m.mat_inte =
ma.Cod_mat and "" & txtcodigo.Text & ""= mat_programa and m.mat_idestudiante = e.est_identi and (" & txtcodasi.Text
+ "" = ma.Cod_m1 or "" & txtcodasi.Text & ""= ma.Cod_m2 or "" & txtcodasi.Text & ""= ma.Cod_m3 or "" & txtcodasi.Text &
""=ma.Cod_m4 or "" & txtcodasi.Text & ""=ma.Cod_m5 or "" & txtcodasi.Text & ""=ma.Cod_m6 or "" & txtcodasi.Text &
""=ma.Cod_m7 or "" & txtcodasi.Text & ""=ma.Cod_m8 or "" & txtcodasi.Text & ""=ma.Cod_m9 or "" & txtcodasi.Text & ""=
ma.Cod_m10)" & _
' adon.RecordSource = "select * from asignatura_estudiante where ases_codasignatura="" & Me.txtcodasi.Text & "" "
' adon.Refresh
'
l = "select distinct
a.ases_idestudiante,e.est_nombres,e.est_apellidos,a.ases_nota1,a.ases_nota2,a.ases_nota3,a.ases_notafinal from
grupos g,asignatura_estudiante a,estudiante e where"" & txtced.Text & ""= g.gru_docente and"" & txtcodigo.Text & ""=
g.gru_programa and"" & txtsemestre.Text & ""= g.gru_semestre and "" & txtcodgrupo.Text & ""= g.gru_grupo and
g.gru_asignatura = a.ases_codasignatura and a.ases_idestudiante = e.est_identi"
ado_no.RecordSource = l
ado_no.Refresh

```

```

End Sub

```

```

Private Sub cmdprograma_Click()
frmminiprograma.cmdasignar.Caption = "Asignar notas"
frmminiprograma.Show vbModal
txtsemestre.Text = Combo1.Text
txtced.Text = txtcedula.Text
txtcodgrupo.Text = Combo2.Text
End Sub

```

```

Private Sub Combo1_Click()
txtsemestre.Text = Combo1.Text
End Sub

```

```

Private Sub Combo2_Click()

```

```
txtcodgrupo.Text = Combo2.Text
End Sub
```

```
Private Sub Command2_Click()
i2 = DataGrid1.ApproxCount
For i = 1 To i2
```

```
End Sub
```

```
Private Sub DataGrid1_Click()
txtid.Text = DataGrid1.Columns("Identificacion").Text
frmnot.txt1.Text = DataGrid1.Columns("Nota1").Text
frmnot.Text2.Text = DataGrid1.Columns("Nota2").Text
frmnot.Text3.Text = DataGrid1.Columns("Nota3").Text
frmnot.Text3.Text = Replace(frmnot.Text3.Text, ",", ".")
frmnot.Text2.Text = Replace(frmnot.Text2.Text, ",", ".")
frmnot.txt1.Text = Replace(frmnot.txt1.Text, ",", ".")
frmnot.Label1.Caption = DataGrid1.Columns("Nombres").Text + " " + DataGrid1.Columns("Apellidos").Text
frmnot.txtasi.Text = txtcodasi.Text
frmnot.txtid.Text = Me.txtid.Text
frmnot.txtantes1.Text = frmnot.txtid.Text & " " & frmnot.txtasi.Text & " " & frmnot.txt1.Text & " " & frmnot.Text2.Text & " " & frmnot.Text3.Text
frmnot.Show vbModal
End Sub
```

```
'Private Sub DataGrid1_AfterUpdate()
'DataGrid1.AllowUpdate = True
'
'End Sub
```

```
'Private Sub DataGrid1_Click()
'txtnota1.Text = DataGrid1.Columns("Nota1").Text
'End Sub
```

```
'Private Sub DataGrid2_AfterUpdate()
' DataGrid2.Columns(7).Text = (Val(DataGrid2.Columns(4).Text) + Val(DataGrid2.Columns(5).Text) + Val(DataGrid2.Columns(6).Text)) / 3
'End Sub
```

```
Private Sub txtcedula_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
m = "select doc_nombres,doc_apellidos from docente where doc_identi = " & txtcedula.Text & ""
ado_docente.RecordSource = m
ado_docente.Refresh
Combo1.SetFocus
End If
End Sub
```

Frmofoctmatricula

```
Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "academicos.chm", &O0, &O0, SW_NORMAL
End Sub
```

```
Private Sub cmdbuscar_Click()
Dim bus, bus2 As String
Dim s, X As String
If KeyAscii = 13 Then
'registro de matricula
ado_matricula.Recordset.AddNew
'registro de estudiante
ado_estudiante.Recordset.AddNew
```



```

'registro de asignatura
ado_Asimat.Recordset.AddNew
ado_notas.Recordset.AddNew
ado_numero.Refresh

con2 = "select mat_volante from matricula where '" + txtvolante.Text + "' = mat_volante"
ado_val.RecordSource = con2
ado_val.Refresh
If txtval.Text = txtvolante.Text Then
    ado_matricula.Recordset.CancelUpdate
    ado_estudiante.Recordset.CancelUpdate
    ado_Asimat.Recordset.CancelUpdate
    ado_notas.Recordset.CancelUpdate
    MsgBox "Esta Matricula ya fue realizada", vbCritical, "Error de registro de Matricula"
    txtvolante.SetFocus
Else
    txtestado.Text = "REGULAR"
    bus = "select v.vol_cedula,p.pro_nombre,p.pro_codigo,v.vol_inte,v.vol_tipo,v.vol_estado from volantes v,programa p
where v.vol_inte = '" + txtvolante.Text + "' and vol_tipo = 'Matricula' and vol_estado = 'Cancelado' and v.vol_programa
=p.pro_codigo"
    ado_volante.RecordSource = bus
    ado_volante.Refresh
    txtcedula.Text = txtvolcedula.Text
    txtprograma2.Text = txtprograma.Text
    txtcodprograma2.Text = txtcodprograma.Text
    If txtvolestado.Text = "Cancelado" And txtvoltipo.Text = "Matricula" Then
        bus2 = "select
can_tipo,can_nombres,can_apellidos,can_direccion,can_telefijo,can_telemovil,can_email,can_sexo,vol_semestre from
candidato,volantes where can_cedula = '" + txtcedula.Text + "' and can_cedula = vol_cedula and '" + txtvolante.Text +
"'= vol_inte"
        ado_volante.RecordSource = bus2
        ado_volante.Refresh
        txttipo.Text = txtcan_tipo.Text
        txtnom.Text = txtcan_nombres.Text
        txtapellidos.Text = txtcan_apellidos.Text
        txtfijo.Text = txtcan_telefijo.Text
        txtcelular.Text = txtcan_telemovil.Text
        xtemail.Text = txtcan_email.Text
        optsex.Text = txtcan_sexo.Text
        txtdir.Text = txtcan_dir.Text
        txthora.Text = Format(Now, "hh:mm:ss")
        Combo1.Text = txtcansemestre.Text
        Combo1.Enabled = False
        txtsemestre.Text = Combo1.Text

        s = "select p.pen_inteligente,a.asi_nombre,a.asi_semestre from asignatura a,pensum p where '" +
txtcodprograma2.Text + "' = p.pen_codprograma and p.pen_codasignatura = a.asi_codigo and a.asi_semestre = '" +
txtsemestre + "'"
        ado_pensum.RecordSource = s
        ado_pensum.Refresh

'crea el consecutivo del codigo inteligente
If txtnumero2.Text = "" Then
    txtnumero.Text = 1000
Else
    txtnumero.Text = txtnumero2.Text
    ado_numero.Refresh
End If
txtconse.Text = "MAT" + txtcodprograma2.Text + txtnumero.Text
txtmat2.Text = txtconse.Text
txtmat.Text = txtmat2.Text
txtced.Text = txtcedula.Text

If optsex.Text = "M" Then
    optmasculino.Value = True
Else
    optfemenino.Value = False
End If
txtemail.SetFocus
Limpiar
p = DataGrid1.VisibleRows
For i = 0 To p
    N = DataGrid1.Bookmark
    If N = 1 Then
        m1.Text = DataGrid1.Columns("CODIGO").Text
    End If

```

```

If N = 2 Then
    m2.Text = DataGrid1.Columns("CODIGO").Text
End If
If N = 3 Then
    m3.Text = DataGrid1.Columns("CODIGO").Text
End If
If N = 4 Then
    m4.Text = DataGrid1.Columns("CODIGO").Text
End If
If N = 5 Then
    m5.Text = DataGrid1.Columns("CODIGO").Text
End If
If N = 6 Then
    m6.Text = DataGrid1.Columns("CODIGO").Text
End If
If N = 7 Then
    m7.Text = DataGrid1.Columns("CODIGO").Text
End If
If N = 8 Then
    m8.Text = DataGrid1.Columns("CODIGO").Text
End If
If N = 9 Then
    m9.Text = DataGrid1.Columns("CODIGO").Text
End If
If N = 10 Then
    m10.Text = DataGrid1.Columns("CODIGO").Text
End If
If DataGrid1.Bookmark < p Then
    DataGrid1.Bookmark = DataGrid1.Bookmark + 1
End If
Next
cmdguardar.Enabled = True
cmdcancelar.Enabled = True
If txtcansestremestre.Text = "Segundo" Then
    s = "select * from asignatura_estudiante where " + txtcedula.Text + " = ases_idestudiante and ases_matricula
= mat_inte and mat_semestre = Primero "
    ado_c.RecordSource = s
    ado_c.Refresh
End If
Else
    MsgBox "Registro no encontrado", vbCritical
    ado_matricula.Recordset.CancelUpdate
    ado_estudiante.Recordset.CancelUpdate
    ado_Asimat.Recordset.CancelUpdate
    ado_notas.Recordset.CancelUpdate
    txttipo.Text = ""
    txtnom.Text = ""
    txtapellidos.Text = ""
    txtfijo.Text = ""
    txtcelular.Text = ""
    txtemail.Text = ""
    optsex.Text = ""
    txtdir.Text = ""
    txtvolante.Text = ""
    Unload Me
    frmofcmatricula.Show vbModal
End If
End If
'End If
End Sub

Private Sub cmdcancelar_Click()
    ado_matricula.Recordset.CancelUpdate
    ado_estudiante.Recordset.CancelUpdate
    ado_Asimat.Recordset.CancelUpdate
    ado_notas.Recordset.CancelUpdate
    cmdcancelar.Enabled = False
    cmdguardar.Enabled = False
    Unload Me
    frmofcmatricula.Show vbModal
End Sub

Private Sub cmdcerrar_Click()
    X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
    If X = vbYes Then
        Unload Me
    End If
End Sub

```



```

End If
End Sub
Private Sub cmdguardar_Click()
s = "select est_identi from estudiante where '" + txtcedula.Text + "'= est_identi"
ado_c.RecordSource = s
ado_c.Refresh
'ado_notas.Recordset.Update
txtn.Text = DataGrid1.ApproxCount
For i = 1 To txtn.Text
txtmt.Text = txtmat.Text
txtced100.Text = txtced.Text
txtmat100.Text = DataGrid1.Columns("Codigo").Text
ado_notas.Recordset.Update
ado_notas.Recordset.AddNew
DataGrid1.Bookmark = i
Next
'Matricula
txtvol.Text = txtvolante.Text
txtfecham.Text = tfecha.Value
txtsemestre.Text = Combo1.Text
txthora2.Text = txthora.Text
'cod_m = txtconse.Text
txtcedula3.Text = txtcedula.Text
'ced = txtcedula3.Text
'fecha = txtfecham.Text
'hora = txthora2.Text
'prog = txtcodprograma2.Text
'semestre = txtsemestre.Text
'N = txtnumero.Text

ado_matricula.Recordset.Update

'txtnumero.Text = N
'txtconse.Text = cod_m
'txtcedula3.Text = ced
'txtfecham.Text = fecha
'txthora2.Text = hora
'txtcodprograma2.Text = prog
'txtsemestre.Text = semestre

'estudiante
If optmasculino = True Then
    optsex.Text = "M"
End If
If optfemenino = True Then
    optsex.Text = "F"
End If
txtcedula2.Text = txtcedula.Text
txttipo2.Text = txttipo.Text
txtnombres2.Text = txtnom.Text
txtapellido.Text = txtapellidos.Text
txtdireccion2.Text = txtdir.Text
txttelef2.Text = txtfijo.Text
txtmovil2.Text = txtcelular.Text
txtemail2.Text = txtemail.Text
txtsexo.Text = optsex.Text
txtfe2.Text = txtfecha.Value
txtestc.Text = txtcivil.Text

'a1 = txtcedula2.Text
'a2 = txttipo2.Text
'a3 = txtnombres2.Text
'a4 = txtapellido.Text
'a5 = txtdireccion2.Text
'a6 = txttelef2.Text
'a7 = txtmovil2.Text
'a8 = txtemail2.Text
'a9 = txtsexo.Text
'a10 = txtfe2.Text
'a11 = txtestc.Text

'txtcedula2.Text = a1
'txttipo2.Text = a2
'txtnombres2.Text = a3
'txtapellido.Text = a4
'txtdireccion2.Text = a5

```

```

'txttelef2.Text = a6
'txtmovil2.Text = a7
'txtemail2.Text = a8
'txtsexo.Text = a9
'txtfe2.Text = a10
'txtesc.Text = a11
If txtc.Text = "" Then
    ado_estudiante.Recordset.Update
Else
    ado_estudiante.Recordset.CancelUpdate
End If
'Asignaturas matriculadas
't1 = m1.Text
't2 = m2.Text
't3 = m3.Text
't4 = m4.Text
't5 = m5.Text
't6 = m6.Text
't7 = m7.Text
't8 = m8.Text
't9 = m9.Text
't10 = m10.Text
MAT = txtmat2.Text

```

```

txtmat2.Text = MAT
'm1.Text = t1
'm2.Text = t2
'm3.Text = t3
'm4.Text = t4
'm5.Text = t5
'm6.Text = t6
'm7.Text = t7
'm8.Text = t8
'm9.Text = t9
'm10.Text = t10

```

```

ado_auditor.Recordset.AddNew
txtiden2.Text = txtvolante.Text
txtidenti.Text = txtcedula.Text
txtident3.Text = txtconse.Text
txtoperacion.Text = "Matricula Académica"
txtusu.Text = frmprincipal.Label4
txtfec.Text = Format(Date, "dd-mm-yyyy")
txth.Text = Format(Now, "h:nn:ss")
txtmodulo.Text = "Oficializar Matricula"
ado_auditor.Recordset.Update

```

```

ado_Asimat.Recordset.Update
MsgBox "El proceso de Matricula se realizó con éxito", vbInformation, "Matricula Exitosa"
cmdcancelar.Enabled = False
cmdguardar.Enabled = False
Unload Me
frmofcmatricula.Show vbModal
'Unload Me
'frmofcmatricula.Show vbModal
End Sub
Private Sub cmdprograma_Click()
    frmminiprograma.cmdasignar.Caption = "Matricula"
    frmminiprograma.Show vbModal
End Sub
Private Sub Combo1_Click()
    Dim s, X As String
    txtsemestre.Text = Combo1.Text
    s = "select a.asi_codigo,a.asi_nombre,a.asi_semeslre from asignatura a,pensum p where" + txtcodprograma2.Text +
    "" = p.pen_codprograma and p.pen_codasignatura = a.asi_codigo and a.asi_semestre ="" + txtsemestre + ""
    ado_pensum.RecordSource = s
    ado_pensum.Refresh
End Sub
Sub Limpiar()
    m1.Text = ""
    m2.Text = ""
    m3.Text = ""
    m4.Text = ""
    m5.Text = ""

```

```

m6.Text = ""
m7.Text = ""
m8.Text = ""
m9.Text = ""
m10.Text = ""
End Sub

```

```

Private Sub Command1_Click()
s = "select DISTINCT g.gru_codigo,pro_nombre,g.gru_semestre,g.gru_grupo from grupos g,programa p where "" +
txtcodprograma2 + "" = g.gru_programa and "" + txtsemestre.Text + "" = g.gru_semestre and g.gru_programa =
p.pro_codigo"
frmgrupos.Adodc1.RecordSource = s
frmgrupos.Adodc1.Refresh
frmgrupos.Show vbModal
End Sub

```

```

Private Sub Form_Load()
tfecha.Refresh
txtconse.Text = ""
txtestado.Text = ""
txtcodgrupo.Text = ""
txtnivel.Text = ""
frmofcmatricula.Caption = "Oficializar Matricula - Gestión Académica 1.0 " & Format(Date, "dd-mm-yyyy") & " " &
Format(Now, "h:nn:ss AM/PM")
End Sub

```

```

Private Sub txtvolante_KeyPress(KeyAscii As Integer)
Dim bus, bus2 As String
Dim s, X As String
If KeyAscii = 13 Then
'registro de matricula
ado_matricula.Recordset.AddNew
'registro de estudiante
ado_estudiante.Recordset.AddNew
'registro de asignatura
ado_Asimat.Recordset.AddNew
ado_notas.Recordset.AddNew
ado_numero.Refresh

con2 = "select mat_volante from matricula where "" + txtvolante.Text + "" = mat_volante"
ado_val.RecordSource = con2
ado_val.Refresh
If txtval.Text = txtvolante.Text Then
ado_matricula.Recordset.CancelUpdate
ado_estudiante.Recordset.CancelUpdate
ado_Asimat.Recordset.CancelUpdate
ado_notas.Recordset.CancelUpdate
MsgBox "Esta Matricula ya fue realizada", vbCritical, "Error de registro de Matricula"
txtvolante.SetFocus
Else
txtestado.Text = "REGULAR"
bus = "select v.vol_cedula,p.pro_nombre,pro_codigo,v.vol_inte,v.vol_tipo,v.vol_estado from volantes v,programa p
where v.vol_inte="" + txtvolante.Text + ""and vol_tipo = 'Matricula' and vol_estado = 'Cancelado' and v.vol_programa
=p.pro_codigo"
ado_volante.RecordSource = bus
ado_volante.Refresh
txtcedula.Text = txtvolcedula.Text
txtprograma2.Text = txtprograma.Text
txtcodprograma2.Text = txtcodprograma.Text
If txtvolestado.Text = "Cancelado" And txtvoltipo.Text = "Matricula" Then
bus2 = "select
can_tipo,can_nombres,can_apellidos,can_direccion,can_telefijo,can_telemovil,can_email,can_sexo,vol_semestre from
candidato,volantes where can_cedula = "" + txtcedula.Text + "" and can_cedula = vol_cedula and "" + txtvolante.Text +
""= vol_inte"
ado_volante.RecordSource = bus2
ado_volante.Refresh
txttipo.Text = txtcan_tipo.Text
txtnom.Text = txtcan_nombres.Text
txtapellidos.Text = txtcan_apellidos.Text
txtfijo.Text = txtcan_telefijo.Text
txtcelular.Text = txtcan_telemovil.Text
xttemail.Text = txtcan_email.Text
optsex.Text = txtcan_sexo.Text
txtdir.Text = txtcan_dir.Text

```

```

txthora.Text = Format(Now, "hh:mm:ss")
Combo1.Text = txtcansestestre.Text
Combo1.Enabled = False
txtsestestre.Text = Combo1.Text

s = "select p.pen_inteligente,a.asi_nombre,a.asi_sestestre from asignatura a,pensum p where" +
txtcodprograma2.Text + "=" + p.pen_codprograma and p.pen_codasignatura = a.asi_codigo and a.asi_sestestre =" +
txtsestestre + """"
ado_pensum.RecordSource = s
ado_pensum.Refresh

'crea el consecutivo del codigo inteligente
If txtnumero2.Text = "" Then
    txtnumero.Text = 1000
Else
    txtnumero.Text = txtnumero2.Text
    ado_numero.Refresh
End If
txtconse.Text = "MAT" + txtcodprograma2.Text + txtnumero.Text
txtmat2.Text = txtconse.Text
txtmat.Text = txtmat2.Text
txtced.Text = txtcedula.Text

If optsex.Text = "M" Then
    optmasculino.Value = True
Else
    optfemenino.Value = False
End If
txtemail.SetFocus
Limpiar
p = DataGrid1.VisibleRows
For i = 0 To p
    N = DataGrid1.Bookmark
    If N = 1 Then
        m1.Text = DataGrid1.Columns("CODIGO").Text
    End If
    If N = 2 Then
        m2.Text = DataGrid1.Columns("CODIGO").Text
    End If
    If N = 3 Then
        m3.Text = DataGrid1.Columns("CODIGO").Text
    End If
    If N = 4 Then
        m4.Text = DataGrid1.Columns("CODIGO").Text
    End If
    If N = 5 Then
        m5.Text = DataGrid1.Columns("CODIGO").Text
    End If
    If N = 6 Then
        m6.Text = DataGrid1.Columns("CODIGO").Text
    End If
    If N = 7 Then
        m7.Text = DataGrid1.Columns("CODIGO").Text
    End If
    If N = 8 Then
        m8.Text = DataGrid1.Columns("CODIGO").Text
    End If
    If N = 9 Then
        m9.Text = DataGrid1.Columns("CODIGO").Text
    End If
    If N = 10 Then
        m10.Text = DataGrid1.Columns("CODIGO").Text
    End If
    If DataGrid1.Bookmark < p Then
        DataGrid1.Bookmark = DataGrid1.Bookmark + 1
    End If
Next
cmdguardar.Enabled = True
cmdcancelar.Enabled = True
* If txtcansestestre.Text = "Segundo" Then
    s = "select * from asignatura_estudiante where " + txtcedula.Text + "=" + ases_idestudiante and ases_matricula
= mat_inte and mat_sestestre = Primero "
*     ado_c.RecordSource = s
*     ado_c.Refresh
* End If
Else

```

```

        MsgBox "Registro no encontrado", vbCritical
        ado_matricula.Recordset.CancelUpdate
        ado_estudiante.Recordset.CancelUpdate
        ado_Asimat.Recordset.CancelUpdate
        ado_notas.Recordset.CancelUpdate
        txttipo.Text = ""
        txtnom.Text = ""
        txtapellidos.Text = ""
        txtfijo.Text = ""
        txtcelular.Text = ""
        txtemail.Text = ""
        optsex.Text = ""
        txtdir.Text = ""
        txtvolante.Text = ""
        Unload Me
        frmofcmatricula.Show vbModal
    End If
End If
End Sub
End Sub
Frmpenresul

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation
As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As
Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "reportes.chm", &O0, &O0, SW_NORMAL
End Sub

'Private Sub cmdresultado_Click()
'Dim bus As String
'bus = combo_programa.Text
'ado_programa.Recordset.MoveFirst
'ado_programa.Recordset.Find ("pro_nombre=" & bus & "")
'End Sub

Private Sub cmdcerrar_Click()
Unload Me
End Sub

Private Sub Form_Load()
'Adodc1.Recordset.MoveFirst
End Sub

Frmpensum

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation
As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As
Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "parametrizacion.chm", &O0, &O0, SW_NORMAL
End Sub

Private Sub cmdagregar_Click()
ado_pensum.Recordset.AddNew
MsgBox "Seleccione la asignatura", vbInformation, "Selección"
txtcodpro.Text = tem.Text
cmdguardar.Enabled = True
cmdsalir.Enabled = True
DataGrid1.Enabled = True
cmdagregar.Enabled = False
cmdcerrar.Enabled = False
End Sub

Private Sub cmdcerrar_Click()
X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
If X = vbYes Then
    Unload Me
End If
End Sub
Private Sub cmdguardar_Click()
Dim s As String

```

```

s = "select pen_codprograma,pen_codasignatura from pensum where '" + txtcodpro.Text + "' = pen_codprograma and '"
+ txtcodasig.Text + "' = pen_codasignatura"
bus.RecordSource = s
bus.Refresh
If txtbus.Text = txtcodasig.Text And bus1.Text = txtcodpro.Text Then
    MsgBox "La Asignatura ya hace parte del Pensum de este Programa", vbCritical, "Error de Asiganción"
    ado_pensum.Recordset.CancelUpdate
    cmdguardar.Enabled = False
    cmdagregar.Enabled = True
    cmdsalir.Enabled = False
    DataGrid1.Enabled = False
    cmdcerrar.Enabled = True
Else
    ' este grupo de intruccines permite crear una parte del codigo inteligente
    If txtsemestre.Text = "Primero" Then
        s = "PRI"
    End If
    If txtsemestre.Text = "Segundo" Then
        s = "SEG"
    End If
    If txtsemestre.Text = "Tercero" Then
        s = "TER"
    End If
    If txtsemestre.Text = "Cuarto" Then
        s = "CUA"
    End If
    If txtsemestre.Text = "Quinto" Then
        s = "QUI"
    End If
    If txtsemestre.Text = "Sexto" Then
        s = "SEX"
    End If
    If txtsemestre.Text = "Curso Basico" Then
        s = "CBA"
    End If
    If txtsemestre.Text = "Diplomado" Then
        s = "DIP"
    End If
    If txtsemestre.Text = "Especializacion" Then
        s = "ESP"
    End If
    'con esta instruccion se crea el codidgo inteligente
    txtinte.Text = txtcodpro.Text + s + txtcodasig.Text

    ado_auditor.Recordset.AddNew
    txtiden2.Text = txtcodasig.Text
    txtidenti.Text = txtcodpro.Text
    txtident3.Text = txtinte.Text
    txtoperacion.Text = "Asignacion Pensum"
    txtusu.Text = frmprincipal.Label4
    txtfecha.Text = Format(Date, "dd-mm-yyyy")
    txthora.Text = Format(Now, "h:nn:ss")
    txtmodulo.Text = "Asignacion Pensum"
    ado_auditor.Recordset.Update

    ado_pensum.Recordset.Update
    ado_pensum.Recordset.MoveFirst
    cmdguardar.Enabled = False
    cmdagregar.Enabled = True
    cmdsalir.Enabled = False
    DataGrid1.Enabled = False
    cmdcerrar.Enabled = True
    MsgBox "El registro ha sido guardado con exito", vbInformation
    cmdpoceso.Enabled = True
End If
End Sub

Private Sub cmdpoceso_Click()
MsgBox "Usted acaba de iniciar el proceso de asignacion de pensum, por favor escoja el programa con el que dara
inicio al procedimiento". vbInformation
cmdprograma.Enabled = True
cmdprograma.SetFocus
cmdpoceso.Enabled = False
'ado_asignatura.RecordSource = "select a.asi_codigo,a.asi_nombre,asi_semestre from asignatura a"

```

```
'ado_asignatura.Refresh
End Sub
```

```
Private Sub cmdprograma_Click()
MsgBox "Por favor escoja el programa al cual desea asignar el pemsun", vbInformation
'ado_pensum.Recordset.AddNew
frmminiprograma.cmdasignar.Caption = "&Asignar"
frmminiprograma.Show vbModal
End Sub
Private Sub Cmdsalir_Click()
ado_pensum.Recordset.CancelUpdate
cmdguardar.Enabled = False
cmdagregar.Enabled = True
cmdsalir.Enabled = False
DataGrid1.Enabled = False
cmdcerrar.Enabled = True
MsgBox "Usted acaba de cancelar el proceso de asignacion, para restablecer debe escoger nuevamente una
asignatura", vbInformation
cmdpoceso.Enabled = True
End Sub
```

```
Private Sub DataGrid1_Click()
MsgBox "Para completar la asignacion debe guardar el registro, para omitir el proceso debe cancelar la operacion",
vbInformation
txtcodasig.Text = DataGrid1.Columns("Codigo").Text
txtsemestre.Text = DataGrid1.Columns("Semestre").Text
End Sub
```

```
Private Sub DataGrid1_RowColChange(LastRow As Variant, ByVal LastCol As Integer)
txtcodasig.Text = DataGrid1.Columns("Codigo").Text
txtsemestre.Text = DataGrid1.Columns("Semestre").Text
End Sub
```

```
Private Sub Form_Load()
ado_asignatura.Refresh
End Sub
```

Frmpermisos

```
Private Sub asignacion_Click()
If asignacion.Value = Checked Then
txtasig.Text = 1
Else
txtasig.Text = 0
End If
End Sub
```

```
Private Sub Asignatura_Click()
If Asignatura.Value = Checked Then
txtasi.Text = 1
Else
txtasi.Text = 0
End If
End Sub
```

```
Private Sub backup_Click()
If backup.Value = Checked Then
txtbackup.Text = 1
Else
txtbackup.Text = 0
End If
End Sub
```

```
Private Sub Candidato_Click()
If Candidato.Value = Checked Then
txtca.Text = 1
Else
txtca.Text = 0
End If
```

```
End Sub
```

```

Private Sub Check1_Click()
If Check1.Value = Checked Then
    txtvol.Text = 1
Else
    txtvol.Text = 0
End If
End Sub

```

```

Private Sub Check2_Click()

End Sub

```

```

Private Sub Check5_Click()
If Check5.Value = Checked Then
    txtlog.Text = 1
Else
    txtlog.Text = 0
End If
End Sub

```

```

Private Sub cmdguardar_Click()
asignacion.Enabled = False
Asignatura.Enabled = False
Candidato.Enabled = False
Check1.Enabled = False
Check5.Enabled = False
Dependencias.Enabled = False
Docente.Enabled = False
Estudiante.Enabled = False
Inscripcion.Enabled = False
Matri.Enabled = False
Matricula.Enabled = False
Notas.Enabled = False
Pensum.Enabled = False
Pen.Enabled = False
Programa.Enabled = False
rAsignaturas.Enabled = False
rCandidato.Enabled = False
rEstudiantes.Enabled = False
Usuarios.Enabled = False
generador.Enabled = False
backup.Enabled = False
operacionesbd.Enabled = False
todos.Enabled = False
cmdguardar.Enabled = False
txtcod2.Text = txtcodigo.Text
ado_permisos.Recordset.Update
MsgBox "Permisos Guardados Satisfactoriamente a este Perfil", vbInformation, "Guardar Permisos"
ado_permisos.Recordset.MoveFirst
End Sub

```

```

Private Sub Cmdsalir_Click()
X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
If X = vbYes Then
    Unload Me
End If
End Sub

```

```

Private Sub Command1_Click()
Frame1.Enabled = True
Frame2.Enabled = True
Frame3.Enabled = True
Frame4.Enabled = True
Frame5.Enabled = True
Frame6.Enabled = True
Administracion.Enabled = True
If Val(frmusu.txtsw.Text) = 1 Then
    asignacion.Value = Unchecked
    Asignatura.Value = Unchecked
    Candidato.Value = Unchecked
    Check1.Value = Unchecked
    Check5.Value = Unchecked
    Dependencias.Value = Unchecked
    Docente.Value = Unchecked
    Estudiante.Value = Unchecked
    Inscripcion.Value = Unchecked
    Matri.Value = Unchecked

```

Matricula.Value = Unchecked
Notas.Value = Unchecked
Pensum.Value = Unchecked
Pen.Value = Unchecked
Programa.Value = Unchecked
rAsignaturas.Value = Unchecked
rCandidato.Value = Unchecked
rEstudiantes.Value = Unchecked
Usuarios.Value = Unchecked
generador.Value = Unchecked
backup.Value = Unchecked
operacionesbd.Value = Unchecked

Else
If txtca.Text = 1 Then
 Candidato.Value = Checked
End If
If txtes.Text = 1 Then
 Estudiante.Value = Checked
End If
If txtdo.Text = 1 Then
 Docente.Value = Checked
End If
If txtpro.Text = 1 Then
 Programa.Value = Checked
End If
If txtasi.Text = 1 Then
 Asignatura.Value = Checked
End If
If txtpen.Text = 1 Then
 Pensum.Value = Checked
End If
If txtasig.Text = 1 Then
 asignacion.Value = Checked
End If
If txtlisd.Text = 1 Then
 Dependencias.Value = Checked
End If
If txtpensu.Text = 1 Then
 Pen.Value = Checked
End If
If txtvol.Text = 1 Then
 Check1.Value = Checked
End If
If txtcand.Text = 1 Then
 rCandidato.Value = Checked
End If
If txtmat.Text = 1 Then
 Matricula.Value = Checked
End If
If txtn.Text = 1 Then
 Notas.Value = Checked
End If
If txtins.Text = 1 Then
 Inscripcion.Value = Checked
End If
If txtmatri.Text = 1 Then
 Matri.Value = Checked
End If
If txtusu.Text = 1 Then
 Usuarios.Value = Checked
End If
If txtasigna.Text = 1 Then
 rAsignaturas.Value = Checked
End If
If txttestud.Text = 1 Then
 rEstudiantes.Value = Checked
End If
If txtlog.Text = 1 Then
 Check5.Value = Checked
End If
If txtgenerador.Text = 1 Then
 generador.Value = Checked
End If
If txtbackup.Text = 1 Then

```

        backup.Value = Checked
    End If
    If txtoperaciones.Text = 1 Then
        operacionesbd.Value = Checked
    End If

End If
Command1.Enabled = False
End Sub

Private Sub Dependencias_Click()
If Dependencias.Value = Checked Then
    txtlisd.Text = 1
Else
    txtlisd.Text = 0
End If
End Sub

Private Sub Docente_Click()
If Docente.Value = Checked Then
    txtdo.Text = 1
Else
    txtdo.Text = 0
End If
End Sub

Private Sub Estudiante_Click()
If Estudiante.Value = Checked Then
    txtes.Text = 1
Else
    txtes.Text = 0
End If
End Sub

Private Sub Form_Load()
If Val(frmusu.btow.Text) = 1 Then
    Frame1.Enabled = True
    Frame2.Enabled = True
    Frame3.Enabled = True
    Frame4.Enabled = True
    Frame5.Enabled = True
    Frame6.Enabled = True
End If
End Sub

Private Sub generador_Click()
If generador.Value = Checked Then
    txtgenerador.Text = 1
Else
    txtgenerador.Text = 0
End If
End Sub

Private Sub Inscripcion_Click()
If Inscripcion.Value = Checked Then
    txtins.Text = 1
Else
    txtins.Text = 0
End If
End Sub

Private Sub Matri_Click()
If Matri.Value = Checked Then
    txtmatri.Text = 1
Else
    txtmatri.Text = 0
End If
End Sub

Private Sub Matricula_Click()
If Matricula.Value = Checked Then
    txtmat.Text = 1
Else

```

```
txtmat.Text = 0
```

```
End If
```

```
End Sub
```

```
Private Sub ninguno_Click()
```

```
If ninguno.Value = Unchecked Then
```

```
    asignacion.Value = Unchecked
```

```
    Asignatura.Value = Unchecked
```

```
    Candidato.Value = Unchecked
```

```
    Check1.Value = Unchecked
```

```
    Check5.Value = Unchecked
```

```
    Dependencias.Value = Unchecked
```

```
    Docente.Value = Unchecked
```

```
    Estudiante.Value = Unchecked
```

```
    Inscripcion.Value = Unchecked
```

```
    Matr.Value = Unchecked
```

```
    Matricula.Value = Unchecked
```

```
    Notas.Value = Unchecked
```

```
    Pensum.Value = Unchecked
```

```
    Pen.Value = Unchecked
```

```
    Programa.Value = Unchecked
```

```
    rAsignaturas.Value = Unchecked
```

```
    rCandidato.Value = Unchecked
```

```
    rEstudiantes.Value = Unchecked
```

```
    Usuarios.Value = Unchecked
```

```
    generador.Value = Unchecked
```

```
    backup.Value = Unchecked
```

```
    operacionesbd.Value = Unchecked
```

```
    asignacion.Enabled = False
```

```
    Asignatura.Enabled = False
```

```
    Candidato.Enabled = False
```

```
    Check1.Enabled = False
```

```
    Check5.Enabled = False
```

```
    Dependencias.Enabled = False
```

```
    Docente.Enabled = False
```

```
    Estudiante.Enabled = False
```

```
    Inscripcion.Enabled = False
```

```
    Matr.Enabled = False
```

```
    Matricula.Enabled = False
```

```
    Notas.Enabled = False
```

```
    Pensum.Enabled = False
```

```
    Pen.Enabled = False
```

```
    Programa.Enabled = False
```

```
    rAsignaturas.Enabled = False
```

```
    rCandidato.Enabled = False
```

```
    rEstudiantes.Enabled = False
```

```
    Usuarios.Enabled = False
```

```
    generador.Enabled = False
```

```
    backup.Enabled = False
```

```
    operacionesbd.Enabled = False
```

```
End If
```

```
End Sub
```

```
Private Sub Notas_Click()
```

```
If Notas.Value = Checked Then
```

```
    txtn.Text = 1
```

```
Else
```

```
    txtn.Text = 0
```

```
End If
```

```
End Sub
```

```
Private Sub operacionesbd_Click()
```

```
If operacionesbd.Value = Checked Then
```

```
    txtoperaciones.Text = 1
```

```
Else
```

```
    txtoperaciones.Text = 0
```

```
End If
```

```
End Sub
```

```
Private Sub Pen_Click()
```

```
If Pen.Value = Checked Then
```

```
    txtpensu.Text = 1
```

```
Else
```

```
    txtpensu.Text = 0
```

```
End If
End Sub
```

```
Private Sub Pensum_Click()
If Pensum.Value = Checked Then
    txtpen.Text = 1
Else
    txtpen.Text = 0
End If
End Sub
```

```
Private Sub Programa_Click()
If Programa.Value = Checked Then
    txtpro.Text = 1
Else
    txtpro.Text = 0
End If
End Sub
```

```
Private Sub rAsignaturas_Click()
If rAsignaturas.Value = Checked Then
    txtasigna.Text = 1
Else
    txtasigna.Text = 0
End If
End Sub
```

```
Private Sub rCandidato_Click()
If rCandidato.Value = Checked Then
    txtcand.Text = 1
Else
    txtcand.Text = 0
End If
End Sub
```

```
Private Sub rEstudiantes_Click()
If rEstudiantes.Value = Checked Then
    txttestud.Text = 1
Else
    txttestud.Text = 0
End If
End Sub
```

```
Private Sub todos_Click()
If todos.Value = Checked Then
    asignacion.Value = Checked
    Asignatura.Value = Checked
    Candidato.Value = Checked
    Check1.Value = Checked
    Check5.Value = Checked
    Dependencias.Value = Checked
    Docente.Value = Checked
    Estudiante.Value = Checked
    Inscripcion.Value = Checked
    Matri.Value = Checked
    Matricula.Value = Checked
    Notas.Value = Checked
    Pensum.Value = Checked
    Pen.Value = Checked
    Programa.Value = Checked
    rAsignaturas.Value = Checked
    rCandidato.Value = Checked
    rEstudiantes.Value = Checked
    Usuarios.Value = Checked
    generador.Value = Checked
    backup.Value = Checked
    operacionesbd.Value = Checked

    asignacion.Enabled = False
    Asignatura.Enabled = False
    Candidato.Enabled = False
    Check1.Enabled = False
    Check5.Enabled = False
```

```

Dependencias.Enabled = False
Docente.Enabled = False
Estudiante.Enabled = False
Inscripcion.Enabled = False
Matri.Enabled = False
Matricula.Enabled = False
Notas.Enabled = False
Pensum.Enabled = False
Pen.Enabled = False
Programa.Enabled = False
rAsignaturas.Enabled = False
rCandidato.Enabled = False
rEstudiantes.Enabled = False
Usuarios.Enabled = False
generador.Enabled = False
backup.Enabled = False
operacionesbd.Enabled = False
Else

```

```

    asignacion.Value = Unchecked
    Asignatura.Value = Unchecked
    Candidato.Value = Unchecked
    Check1.Value = Unchecked
    Check5.Value = Unchecked
    Dependencias.Value = Unchecked
    Docente.Value = Unchecked
    Estudiante.Value = Unchecked
    Inscripcion.Value = Unchecked
    Matri.Value = Unchecked
    Matricula.Value = Unchecked
    Notas.Value = Unchecked
    Pensum.Value = Unchecked
    Pen.Value = Unchecked
    Programa.Value = Unchecked
    rAsignaturas.Value = Unchecked
    rCandidato.Value = Unchecked
    rEstudiantes.Value = Unchecked
    Usuarios.Value = Unchecked
    generador.Value = Unchecked
    backup.Value = Unchecked
    operacionesbd.Value = Unchecked

```

```

    asignacion.Enabled = True
    Asignatura.Enabled = True
    Candidato.Enabled = True
    Check1.Enabled = True
    Check5.Enabled = True
    Dependencias.Enabled = True
    Docente.Enabled = True
    Estudiante.Enabled = True
    Inscripcion.Enabled = True
    Matri.Enabled = True
    Matricula.Enabled = True
    Notas.Enabled = True
    Pensum.Enabled = True
    Pen.Enabled = True
    Programa.Enabled = True
    rAsignaturas.Enabled = True
    rCandidato.Enabled = True
    rEstudiantes.Enabled = True
    Usuarios.Enabled = True
    generador.Enabled = True
    backup.Enabled = True
    operacionesbd.Enabled = True
End If

```

```

End Sub

```

```

Private Sub Usuarios_Click()
If Usuarios.Value = Checked Then
    txtusu.Text = 1
Else
    txtusu.Text = 0

```

```
End If
End Sub
```

Frmprincipal

```
Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As Long) As Long
Const SW_NORMAL = 1
Private Sub smayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "codetec2.chm", &O0, &O0, SW_NORMAL
End Sub
```

```
Private Sub Form_Load()
frmprincipal.Caption = "Menü Principal - Gestión Académica 1.0 - Fecha:" & Format(Date, "dd-mm-yyyy") & " - Hora:" & Format(Now, "h:nn:ss AM/PM")
Label4.Caption = frmlogin.txtusuario.Text
'tipo.Caption = frmlogin.txttipo.Text
'Inom.Caption = frmlogin.tnom.Text + " " + frmlogin.tapell.Text
End Sub
```

```
Private Sub masignaturas_Click()
frmasignatura.Show vbModal
End Sub
```

```
Private Sub mbackup_Click()
frmbackup.Show vbModal
End Sub
```

```
Private Sub mdocente_Click()
frmdocente.Show vbModal
End Sub
Private Sub minscripcion_Click()
frminscripcion.Show vbModal
frminscripcion.LimpiarControles Me
frminscripcion.cmdbuspro.Enabled = False
End Sub
Private Sub mmatricula_Click()
frmmatricula.Show vbModal
End Sub
Private Sub mnudep_Click()
frmasignacion.Show vbModal
End Sub
Private Sub mnudepl_Click()
frmasignatura.cmdagregar.Enabled = False
frmasignatura.cmdmodificar.Enabled = False
frmasignatura.cmdeliminar.Enabled = False
frmasignatura.cmdasignar.Visible = True
frmasignatura.txtsw.Text = 2
frmasignatura.Show vbModal
End Sub
Private Sub mnugru_Click()
frm apartar.Show vbModal
End Sub
Private Sub mnupa_Click()
frm aplicar.Frame1.Caption = "Aplicar Pago Inscripción"
frm aplicar.cmdlegalizar.Caption = "&Legalizar inscripción"
frm aplicar.Show vbModal
End Sub
Private Sub nmuvolcan_Click()
frmconvol.Show vbModal
End Sub
Private Sub pa_Click()
MsgBox "Por favor escoger El programa para La consulta", vbExclamation + vbInformation, "Consultar programas"
frmminiprograma.cmdasignar.Caption = "&Listar Pensum"
frmminiprograma.Show vbModal
End Sub
Private Sub rsasig_Click()
frmdoc_asig.Show vbModal
End Sub
Private Sub smacerca_Click()
frmacerca.Show vbModal
```

End Sub

```
Private Sub smcandidato_Click()  
    frm candidato.Show vbModal  
End Sub
```

```
Private Sub smcangenerales_Click()  
    frm inscritos.Show vbModal  
End Sub
```

```
Private Sub smcesion_Click()  
    g = MsgBox("¿Seguro que desea cerrar la sesión?", vbYesNo + vbInformation, "Cerrar Sesión")  
    If g = vbYes Then  
        Unload Me  
        frm login.Show vbModal  
    End If  
End Sub
```

```
Private Sub smdina_Click()  
    frm generador.Show vbModal  
End Sub
```

```
Private Sub smestudiante_Click()  
    frm estudiante.Show vbModal  
End Sub
```

```
Private Sub smgenerales_Click()  
    frm matriculados.Show vbModal  
End Sub
```

```
Private Sub smlogs_Click()  
    frm logs.Show vbModal  
End Sub
```

```
Private Sub smmpago_Click()  
    frm aplicar.Frame1.Caption = "Aplicar Pago Matricula"  
    frm aplicar.cmd legalizar.Visible = False  
    frm aplicar.Show vbModal  
End Sub
```

```
Private Sub smmpensum_Click()  
    frm pensum.Show vbModal  
End Sub
```

```
Private Sub smnotas2_Click()  
    frm reg_notas.Show vbModal  
End Sub
```

```
Private Sub smofic_Click()  
    frm ofc matricula.Show vbModal  
End Sub
```

```
Private Sub smpensum_Click()  
    frm PenResul.Show vbModal  
End Sub
```

```
Private Sub smper_Click()  
    frm Perfil.Show vbModal  
End Sub
```

```
Private Sub smpermisos_Click()  
    frm permisos.Show vbModal  
End Sub
```

```
Private Sub smpermi_Click()  
    frm permisos.Show vbModal  
End Sub
```

```
Private Sub smprog_Click()  
    frm programa.Show vbModal  
End Sub
```

```
Private Sub smregistro_Click()  
    frm not doc.Show vbModal  
End Sub
```

```
Private Sub smsalir_Click()
```

```

c = MsgBox("¿Seguro que desea Salir del sistema?", vbYesNo + vbInformation, "Salir del Sistema")
If c = vbYes Then
    End
End If
End Sub

```

```

Private Sub ssexportar_Click()
frmexport.Show vbModal
End Sub

```

```

Private Sub ssmimportar_Click()
frmimport.Show vbModal
End Sub

```

```

Private Sub ssoper_Click()
frmusuario.Show vbModal
End Sub

```

```

Private Sub ssper_Click()
frmusu.Show vbModal
End Sub

```

```

Private Sub sssinscritos_Click()
frminscritos.Show vbModal
End Sub

```

```

Private Sub WebBrowser1_NavigateComplete2(ByVal pDisp As Object, URL As Variant)
'rpt_asignatura.Show vbModal
h.html
End Sub

```

```

Private Sub sssatru_Click()
frmmatriculados.Show vbModal
End Sub

```

Frmprograma

```

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation
As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As
Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "parametrizacion.chm", &O0, &O0, SW_NORMAL
End Sub

```

```

Sub bloquear()
txtcodigo.Enabled = False
txtnombre.Enabled = False
txtricfes.Enabled = False
txtduracion.Enabled = False
txtvalor.Enabled = False
End Sub

```

```

Sub desbloquear()
txtnombre.Enabled = True
txtricfes.Enabled = True
txtduracion.Enabled = True
txtvalor.Enabled = True
End Sub

```

```

Private Sub cmdagregar_Click()
Dim bus As String
If cmdagregar.Caption = "&Agregar" Then
    ado_programa.Refresh
    txtsw.Text = 0
    cmdagregar.Caption = "&Guardar"
    cmdagregar.ToolTipText = "Guardar datos del programa"
    cmdmodificar.Caption = "&Cancelar"
    cmdmodificar.ToolTipText = "Cancelar proceso"
    bus = "select pro_codigo from programa where " + txtcodigo.Text + " = pro_codigo"
    frmapartar.ado_consulta.RecordSource = s
    frmapartar.ado_consulta.Refresh
    ado_programa.Recordset.AddNew
    desbloquear
    txtnombre.SetFocus

```



```

ado_programa.Enabled = False
cmdeliminar.Enabled = False
cmdcerrar.Enabled = False
cmdbuscar.Enabled = False
If txtnumero.Text = "" Then
    txtcodigo.Text = 100
Else
    ado_numero.Refresh
    txtcodigo.Text = txtnumero.Text
End If
Else
    cmdagregar.Caption = "&Agregar"
    cmdmodificar.Caption = "&Modificar"

Do
    If txtnombre.Text = "" Then
        X = InputBox("Debe escribir el nombre del programa", "Nombre del programa")
        txtnombre.Text = X
    End If
    If txtcodigo.Text = "" Then
        X = InputBox("Debe escribir el código del programa", "Código del programa")
        txtcodigo.Text = X
    End If
Loop Until Not (txtnombre.Text = "") And Not (txtcodigo.Text = "")

ado_auditor.Recordset.AddNew
If txtsw.Text = 0 Then
    txtidenti.Text = txtcodigo.Text
    txtoperacion.Text = "Inserción registro"
    txtusu.Text = frmprincipal.Label4
    txtfecha.Text = Format(Date, "dd-mm-yyyy")
    txthora.Text = Format(Now, "h:nn:ss")
    txtmodulo.Text = "Programa"
Else
    txtantes.Text = txtantes1.Text
    txtdespues.Text = txtcodigo.Text & ", " & txtnombre.Text & ", " & txturacion.Text & ", " & txticfes.Text & ", " &
txtvalor.Text
    txtidenti.Text = txtcodigo.Text
    txtoperacion.Text = "Actualización registro"
    txtusu.Text = frmprincipal.Label4
    txtfecha.Text = Format(Date, "dd-mm-yyyy")
    txthora.Text = Format(Now, "h:nn:ss")
    txtmodulo.Text = "Programa"
End If
ado_auditor.Recordset.Update

txtusuario.Text = frmlogin.txtusuario.Text
ado_programa.Recordset.Update
bloquear
ado_programa.Enabled = True
cmdeliminar.Enabled = True
cmdcerrar.Enabled = True
cmdbuscar.Enabled = True
End If
End Sub
Private Sub cmdbuscar_Click()
    Dim bus As String
    bus = InputBox("Digite el código del programa", "Buscar programa")
    If bus <> vbNullString Then
        ado_programa.Refresh
        ado_programa.Recordset.MoveFirst
        ado_programa.Recordset.Find ("pro_codigo=" & bus & "")
        If ado_programa.Recordset.EOF Then
            MsgBox "Verifique el código del programa", vbInformation + vbOKOnly, "Código Incorrecto"
            ado_programa.Recordset.MoveFirst
        End If
    Else
        MsgBox "Vuelva a intentar la búsqueda - Código Nula", vbInformation + vbOKCancel, "Error de Búsqueda"
    End If
End Sub
Private Sub cmdeliminar_Click()
    Dim resp As String
    resp = MsgBox("¿Seguro que desea eliminar este registro?", vbYesNo + vbInformation, "Eliminar registro")
    If resp = vbYes Then

```

```

ado_auditor.Recordset.AddNew
txtidenti.Text = txtcodigo.Text
ado_programa.Recordset.Delete
ado_programa.Recordset.MovePrevious
txtoperacion.Text = "Eliminación registro"
txtusu.Text = frmprincipal.Label4
txtfecha.Text = Format(Date, "dd-mm-yyyy")
txthora.Text = Format(Now, "h:nn:ss")
txtmodulo.Text = "Programa"
ado_auditor.Recordset.Update

End If
ado_programa.Recordset.MoveFirst
End Sub
Private Sub cmdModificar_Click()
If cmdmodificar.Caption = "&Modificar" Then
'estas instrucciones dan como resultado el temporal del registro actual que despues se guardan en la tabla auditor
txtantes1.Text = txtcodigo.Text & ", " & txtnombre.Text & ", " & txtduracion.Text & ", " & txtricfes.Text & ", " &
txtvalor.Text
txtsw.Text = 1
cmdmodificar.Caption = "&Cancelar"
cmdagregar.Caption = "&Guardar"
cmdagregar.ToolTipText = "Guardar datos del programa"
cmdmodificar.ToolTipText = "Cancelar proceso"
desbloquear
txtnombre.SetFocus
ado_programa.Enabled = False
cmdeliminar.Enabled = False
cmdcerrar.Enabled = False
cmdbuscar.Enabled = False
Else
ado_programa.Recordset.CancelUpdate
cmdmodificar.Caption = "&Modificar"
cmdagregar.Caption = "&Agregar"
cmdagregar.ToolTipText = "Agregar un nuevo programa"
cmdmodificar.ToolTipText = "Modificar un programa existente"
ado_programa.Enabled = True
cmdeliminar.Enabled = True
cmdcerrar.Enabled = True
cmdbuscar.Enabled = True
bloquear
End If
End Sub
Private Sub cmdcerrar_Click()
X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
If X = vbYes Then
Unload Me
End If
End Sub

Private Sub Form_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
SendKeys "{Tab}"
End If
End Sub

Private Sub Form_Load()
blanco
bloquear
End Sub

Private Sub txtricfes_Change()
Dim i As Integer
txtricfes.Text = UCase(txtricfes.Text)
i = Len(txtricfes.Text)
txtricfes.SelStart = i
End Sub
Private Sub txtnombre_Change()
Dim i As Integer
txtnombre.Text = UCase(txtnombre.Text)
i = Len(txtnombre.Text)
txtnombre.SelStart = i
End Sub
Sub blanco()
txtcodigo.Text = " "

```

```

txtduracion.Text = " "
txtnombre.Text = " "
txtricfes.Text = " "
txtvalor.Text = " "

```

```
End Sub
```

Frmusu

```

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation
As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As
Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "admin.chm", &O0, &O0, SW_NORMAL
End Sub

```

```

Private Sub cmdeliminar_Click()
Dim resp As String
resp = MsgBox("¿Seguro que desea eliminar este registro?", vbYesNo + vbInformation, "Eliminar registro")
If resp = vbYes Then
    ado_auditor.Recordset.AddNew
    txtidenti.Text = txtcodigo.Text
    txtoperacion.Text = "Eliminación de Perfil"
    txtusu.Text = frmprincipal.Label4
    txtfecha.Text = Format(Date, "dd-mm-yyyy")
    txthora.Text = Format(Now, "h:nn:ss")
    txtmodulo.Text = "Perfiles"
    ado_auditor.Recordset.Update

    ado_perfil.Recordset.Delete
    ado_perfil.Recordset.MovePrevious
End If
ado_perfil.Recordset.MoveFirst
Unload Me
frmusu.Show vbModal
End Sub

```

```

Private Sub cmdguardar_Click()
c = "select per_codigo from perfil where " + txtcodigo.Text + " = per_codigo"
Adodc1.RecordSource = c
Adodc1.Refresh
If txtbus.Text = txtcodigo.Text Then
    MsgBox "Por favor usar otro codigo para el nuevo perfil"
    ado_perfil.Recordset.CancelUpdate
    cmdnuevo.Caption = "&Nuevo"
    cmdguardar.Enabled = False
    Unload Me
    frmusu.Show vbModal
    ado_act.Recordset.MoveFirst

```

```

Else
    ado_auditor.Recordset.AddNew
    txtidenti.Text = txtcodigo.Text
    txtoperacion.Text = "Creación de Perfil"
    txtusu.Text = frmprincipal.Label4
    txtfecha.Text = Format(Date, "dd-mm-yyyy")
    txthora.Text = Format(Now, "h:nn:ss")
    txtmodulo.Text = "Perfiles"
    ado_auditor.Recordset.Update

    ado_perfil.Recordset.Update
    cmdnuevo.Caption = "&Nuevo"
    cmdguardar.Enabled = False
    Unload Me
    frmusu.Show vbModal
    ado_act.Recordset.MoveFirst

```

```

End If
End Sub

```

```

Private Sub cmdmodper_Click()
If cmdmodper.Caption = "&Modificar Permisos" Then
If t1.Text = "" And t2.Text = "" Then
    MsgBox "Debe seleccionar un perfil para poder modificar permisos", vbCritical, "Error de Asignación"

```

```

Else
    txtsw.Text = 0
    frmpermisos.txtcodigo = t1.Text
    frmpermisos.txtnombre = t2.Text
    bus = t1.Text
    frmpermisos.ado_permisos.Recordset.MoveFirst
    frmpermisos.ado_permisos.Recordset.Find ("cod_perfil=" & bus & "")
    MsgBox "Modifique los permisos de este perfil", vbInformation, "Modificar Permisos"
    frmpermisos.Command1.Visible = True
    frmpermisos.Show vbModal
End If
Else
    frmusuario.Label8.Caption = t2.Text
    frmusuario.txtcodper.Text = t1.Text
    cmdmodper.Caption = "&Modificar Permisos"
    frmusu.cmdnuevo.Enabled = True
    frmusu.cmdnper.Enabled = True
    Unload Me
End If
End Sub

Private Sub cmdnper_Click()
If t1.Text = "" And t2.Text = "" Then
    MsgBox "Debe seleccionar un perfil para poder asignarle permisos", vbCritical, "Error de Asignación"
Else
    c = "select cod_perfil from perfil_permiso where " & t1.Text & " = cod_perfil"
    Adodc2.RecordSource = c
    Adodc2.Refresh
    If txtbus2.Text = t1.Text Then
        MsgBox "Este perfil ya tiene permisos asignados, solo puede ser modificado", vbInformation + vbCritical, "Error de asignacion"
    Else
        txtsw.Text = 1
        Frame1.Enabled = True

        frmpermisos.ado_permisos.Recordset.AddNew
        frmpermisos.txtcodigo = t1.Text
        frmpermisos.txtnombre = t2.Text
        MsgBox "Seleccione los permisos de este perfil", vbInformation, "Asignación de Permisos"
        frmpermisos.Show vbModal
    End If
End If
End Sub

Private Sub cmdnuevo_Click()
If cmdnuevo.Caption = "&Nuevo Perfil" Then

    ado_perfil.Recordset.AddNew
    cmdnuevo.Caption = "Cancelar"
    cmdnper.Enabled = False
    cmdmodper.Enabled = False
    txtcodigo.Enabled = True
    txtnombre.Enabled = True
    txtcodigo.SetFocus
    cmdguardar.Enabled = True
    Cmdeliminar.Enabled = False

Else
    cmdnuevo.Caption = "&Nuevo Perfil"
    ado_perfil.Recordset.CancelUpdate
    cmdnper.Enabled = True
    cmdmodper.Enabled = True
    txtcodigo.Enabled = False
    txtnombre.Enabled = False
    cmdguardar.Enabled = False
    Cmdeliminar.Enabled = True
End If
End Sub
Private Sub Cmdsalir_Click()
X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
If X = vbYes Then
    Unload Me
End If
End Sub

```

```

Private Sub DataGrid1_Click()
t1.Text = DataGrid1.Columns("Código").Text
t2.Text = DataGrid1.Columns("Nombre Perfil").Text
bus = t1.Text
ado_perfil.Recordset.MoveFirst
ado_perfil.Recordset.Find ("per_codigo=" & bus & "")
End Sub

Private Sub DataGrid1_RowColChange(LastRow As Variant, ByVal LastCol As Integer)
t1.Text = DataGrid1.Columns("Código").Text
t2.Text = DataGrid1.Columns("Nombre Perfil").Text
End Sub

Private Sub Form_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
    SendKeys "{Tab}"
End If

End Sub

Private Sub txtnombre_Change()
Dim i As Integer
txtnombre.Text = UCase(txtnombre.Text)
i = Len(txtnombre.Text)
txtnombre.SelStart = i
End Sub

```

Frmusuario

```

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal lpOperation As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As Long) As Long
Const SW_NORMAL = 1
Private Sub cmdayuda_Click()
ShellExecute frmprincipal.hwnd, "Open", "admin.chm", &O0, &O0, SW_NORMAL
End Sub

Sub bloquear()
    txtnombres.Enabled = False
    txtapellidos.Enabled = False
    txtcedula.Enabled = False
    txtusuario.Enabled = False
    txtperfil.Enabled = False
    txtcontrasena.Enabled = False
    Text1.Enabled = False
End Sub

Sub desbloquear()
    Text1.Enabled = True
    txtnombres.Enabled = True
    txtapellidos.Enabled = True
    txtcedula.Enabled = True
    txtusuario.Enabled = True
    txtperfil.Enabled = True
    txtcontrasena.Enabled = True
End Sub

Private Sub cmdagregar_Click()
If cmdagregar.Caption = "&Agregar" Then
    txtsw.Text = 0
    cmdagregar.Caption = "&Guardar"
    cmdagregar.ToolTipText = "Guardar datos del usuario"
    cmdmodificar.Caption = "&Cancelar"
    cmdmodificar.ToolTipText = "Cancelar proceso"
    ado_usuario.Recordset.AddNew
    Label4.Caption = " "
    desbloquear
    Text1.SetFocus
    ado_usuario.Enabled = False
    cmdeliminar.Enabled = False
    cmdcerrar.Enabled = False
    cmdbuscar.Enabled = False
    cmdperfil.Enabled = True
Else
    If txtcodper.Text <> "" Then

```

```

cmdagregar.Caption = "&Agregar"
cmdmodificar.Caption = "&Modificar"

Do
If txtnombres.Text = "" Then
X = InputBox("Debe escribir los nombres del usuario", "Nombres del usuario")
txtnombres.Text = X
End If
If txtapellidos.Text = "" Then
X = InputBox("Debe escribir los apellidos del usuario", "Apellidos del usuario")
txtapellidos.Text = X
End If
If txtcedula.Text = "" Then
X = InputBox("Debe escribir la Cedula del usuario", "Cedula del usuario")
txtcedula.Text = X
End If
If txtcontrasena.Text = "" Then
X = InputBox("Debe escribir la contraseña del usuario en el sistema", "Contraseña del usuario")
txtcontrasena.Text = X
End If
Loop Until Not (txtnombres.Text = "") And Not (txtapellidos.Text = "") And Not (txtcedula.Text = "") And Not
(txtcontrasena.Text = "")
txtusuario.Text = frmlogin.txtusuario.Text
txtperfil.Text = txtper.Text

s = "select per_nombre from perfil where " & Label4.Caption & " = per_codigo"
ado_nom.RecordSource = s
ado_nom.Refresh
Label8.Caption = tnom.Text

ado_auditor.Recordset.AddNew
If txtsw.Text = 0 Then
txtidenti.Text = txtcedula.Text
txtiden2.Text = Text1.Text
txtoperacion.Text = "Inserción registro"
txtusu.Text = frmprincipal.Label4
txtfecha.Text = Format(Date, "dd-mm-yyyy")
txthora.Text = Format(Now, "h:nn:ss")
txtmodulo.Text = "Usuarios"
Else
txtantes.Text = txtantes1.Text
txtdespues.Text = Text1.Text & ", " & txtnombres.Text & ", " & txtapellidos.Text & ", " & txtcedula.Text
txtidenti.Text = txtcedula.Text
txtiden2.Text = Text1.Text
txtoperacion.Text = "Actualización registro"
txtusu.Text = frmprincipal.Label4
txtfecha.Text = Format(Date, "dd-mm-yyyy")
txthora.Text = Format(Now, "h:nn:ss")
txtmodulo.Text = "Usuarios"
End If
ado_auditor.Recordset.Update

MsgBox "La Contraseña inicial es el número de la cédula, para cambiarla presione la opción CAMBIAR
CONTRASEÑA en el inicio de sesión", vbInformation, "Asignación de Contraseña"
ado_usuario.Recordset.Update
Dim Db As New ADODB.Connection
Db.Open "Provider=MSDASQL.1;Persist Security Info=False;Extended Properties='DRIVER={MySQL ODBC 3.51
Driver};SERVER=localhost;DATABASE=Codetec ;UID=root ;OPTION=3'"
Db.Execute "update usuario set usu_contrasena = password('" & txtcontrasena.Text & "') where usu_cedula = " &
txtcedula.Text & ""
Db.Close
ado_usuario.Recordset.MoveFirst

cmdperfil.Enabled = False
Unload Me
frmusuario.Show vbModal
If txtperfil.Text = "Docente" Then

```

```

        ado_doce.Recordset.AddNew
        txtnombre2.Text = txtnombres.Text
        txtapellido2.Text = txtapellidos.Text
        txtcedula3.Text = txtcedula.Text
        ado_doce.Recordset.Update
        ado_doce.Recordset.MoveNext
    End If
    bloquear
    ado_usuario.Enabled = True
    cmdeliminar.Enabled = True
    cmdcerrar.Enabled = True
    cmbuscar.Enabled = True

Else
    MsgBox "Debe asignarle un perfil al usuario", vbInformation
    cmdperfil.SetFocus
End If
End If

End Sub

Private Sub cmbuscar_Click()
    Dim bus As String
    bus = InputBox("Digite La cedula del usuario", "Buscar usuario")
    If bus <> vbNullString Then
        ado_usuario.Recordset.MoveFirst
        ado_usuario.Recordset.Find ("usu_cedula=" & bus & "")
        s = "select per_nombre from perfil where " & Label4.Caption & " = per_codigo"
        ado_nom.RecordSource = s
        ado_nom.Refresh
        Label8.Caption = tnom.Text
    If ado_usuario.Recordset.EOF Then
        MsgBox "Verifique La cedula del usuario", vbInformation + vbOKOnly, "Cedula Incorrecta"
        ado_usuario.Recordset.MoveFirst
    End If
    Else
        MsgBox "Vuelva a intentar la busqueda - Cedula Nula", vbInformation + vbOKCancel, "Error de Busqueda"
    End If
End Sub

Private Sub cmdeliminar_Click()
    Dim resp As String
    resp = MsgBox("¿Seguro que desea eliminar este registro?", vbYesNo + vbInformation, "Eliminar registro")
    If resp = vbYes Then
        ado_auditor.Recordset.AddNew
        txtidenti.Text = txtcedula.Text
        txtiden2.Text = Text1.Text
        txtoperacion.Text = "Eliminación registro"
        txtusu.Text = frmprincipal.Label4
        txtfecha.Text = Format(Date, "dd-mm-yyyy")
        txthora.Text = Format(Now, "h:nn:ss")
        txtmodulo.Text = "Usuarios"
        ado_auditor.Recordset.Update
        ado_usuario.Recordset.Delete
        ado_usuario.Recordset.MovePrevious
    End If
    ado_usuario.Recordset.MoveFirst
End Sub

Private Sub cmdModificar_Click()
    If cmdmodificar.Caption = "&Modificar" Then
        'estas instrucciones dan como resultado el temporal del registro actual que despues se guardan en la tabla auditor
        txtantes1.Text = Text1.Text & ", " & txtnombres.Text & ", " & txtapellidos.Text & ", " & txtcedula.Text
        txtsw.Text = 1
        cmdmodificar.Caption = "&Cancelar"
        cmdagregar.Caption = "&Guardar"
        cmdagregar.ToolTipText = "Guardar datos del usuario"
        cmdmodificar.ToolTipText = "Cancelar proceso"
        desbloquear
        txtnombres.SetFocus
        ado_usuario.Enabled = False
        cmdeliminar.Enabled = False
        cmdcerrar.Enabled = False
        cmbuscar.Enabled = False
        cmdperfil.Enabled = True
        txtcontrasena.Enabled = False
    Else

```

```

        ado_usuario.Recordset.CancelUpdate
        cmdmodificar.Caption = "&Modificar"
        cmdagregar.Caption = "&Agregar"
        cmdagregar.ToolTipText = "Agregar un nuevo usuario"
        cmdmodificar.ToolTipText = "Modificar un usuario existente"
        ado_usuario.Enabled = True
        cmdeliminar.Enabled = True
        cmdcerrar.Enabled = True
        cmdbuscar.Enabled = True
        cmdperfil.Enabled = False
        bloquear
    End If
End Sub
Private Sub cmdcerrar_Click()
X = MsgBox("¿Desea cerrar este modulo de la aplicación?", vbYesNo + vbInformation, "Cerrar")
If X = vbYes Then
    Unload Me
End If
End Sub

Public Sub LimpiarControles(frmusuario As Form)
Dim ctlControl As Object
On Error Resume Next
For Each ctlControl In frmusuario.Controls
    ctlControl.Text = ""
    ctlControl.ListIndex = -1
    DoEvents
Next ctlControl
End Sub

Private Sub cmdperfil_Click()
frmusu.cmdmodper.Caption = "Asignar perfil"
frmusu.cmdnuevo.Enabled = False
frmusu.cmdnper.Enabled = False
frmusu.cmdeliminar.Enabled = False
frmusu.Show vbModal
End Sub

Private Sub Form_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
    SendKeys "{Tab}"
End If
End Sub

Private Sub Form_Load()
bloquear
End Sub

Private Sub Text1_Change()
Dim i As Integer
Text1.Text = UCase(Text1.Text)
i = Len(Text1.Text)
Text1.SelStart = i
End Sub

Private Sub txtapellidos_Change()
Dim i As Integer
txtapellidos.Text = UCase(txtapellidos.Text)
i = Len(txtapellidos.Text)
txtapellidos.SelStart = i
End Sub

Private Sub txtcedula_Change()
txtcontrasena.Text = txtcedula.Text
End Sub

Private Sub txtnombres_Change()
Dim i As Integer
txtnombres.Text = UCase(txtnombres.Text)
i = Len(txtnombres.Text)
txtnombres.SelStart = i
End Sub

```




**DISEÑO, DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA DE
INFORMACIÓN, PARA EL MANEJO DE LOS PROCESOS ACADÉMICOS
EN LA CORPORACIÓN TÉCNICA DE ESTUDIOS ESPECIALIZADOS DEL
CARIBE (CODETEC)**

MANUAL DEL USUARIO

JOSÉ G. DE LA CRUZ

NARCISO CANTILLO

WILLIAM CASTILLO

FREDY RENY

BARRANQUILLA

UNIVERSIDAD SIMÓN BOLÍVAR

INGENIERÍA DE SISTEMAS

XI SEMESTRE – JORNADA NOCTURNA

2007



OPERACIONES BASICAS

- **Agregar – Guardar**



Permite almacenar la información que usted ha digitado previamente en los campos de cualquier módulo del aplicativo.

¿Cómo utilizarlo?

Presione el botón **Agregar**, para el ingreso de datos, este cambia de estado a **Guardar**, permitiendo el almacenamiento de la información.

Haga un clic sobre el botón ilustrado, con la ayuda del Mouse, o pulse las teclas ALT + A ó ALT + G

- **Modificar – Cancelar**

Permite Modificar la información previamente almacenada.

¿Cómo utilizarlo?

Presione el botón **Modificar**, para la actualización de datos, este cambia de estado a **Cancelar**, para omitir la actualización.

Haga un clic sobre el botón ilustrado, con la ayuda del Mouse, o pulse las teclas ALT + M ó ALT + C



- **Eliminar** Eliminar

Permite eliminar un registro de la base de datos previamente seleccionado, por encontrarse con información errónea o desactualizado.

¿Cómo utilizarlo?

Seleccione el registro (Ver en la opción Buscar).

Haga clic en el botón de eliminar ilustrado anteriormente.

Confirme la aceptación de la eliminación en el cuadro de dialogo ilustrado por pantalla.

Haga un clic sobre el botón ilustrado, con la ayuda del Mouse, o pulse las teclas ALT + E.

- **Buscar** Buscar

Permite consultar un registro almacenado previamente.

¿Cómo utilizarlo?

Haga un clic sobre el botón ilustrado, con la ayuda del Mouse, o pulse las teclas ALT + B.

Digite la información a consultar y presione el botón aceptar como lo indica el cuadro de dialogo ilustrado por pantalla.



- **Cerrar** Cerrar

Permite cerrar un modulo de la aplicación, presione el botón Si / No en el cuadro de dialogo ilustrado por pantalla para confirmar el proceso.

Haga un clic sobre el botón ilustrado, con la ayuda del Mouse, o pulse las teclas ALT + C.



MÓDULOS DE LA APLICACIÓN

1. INICIO DE SESIÓN

En esta pantalla se realiza el ingreso al sistema

Procesos

- En el campo  digitamos el nombre de usuario que le fue asignado
- En el campo contraseña, digitamos la contraseña del usuario
- Para realizar el ingreso después de digitar los datos requeridos para el inicio

de sesión presionamos al botón  Ingresar

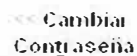


- Para cancelar el ingreso y blanquear los campos de la pantalla presionamos



el botón Cancelar

- Dado el caso que el usuario desee cambiar de contraseña , presionamos el

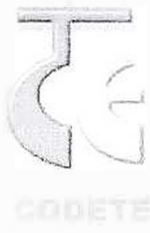


botón Cambiar Contraseña para que nos permita en cambio de contraseña

- En el campo **usuario**, digitamos el nombre del usuario
- En el campo **Contraseña anterior**, digitamos la contraseña que se esta utilizando en la actualidad
- Luego de digitar la contraseña actual, digitamos en el campo **Nueva contraseña**, la nueva contraseña
- Para guardar los cambios realizados al contraseña presionamos el



botón Aceptar



- Para salir del modulo de cambio de contraseña presionamos el botón



Cerrar

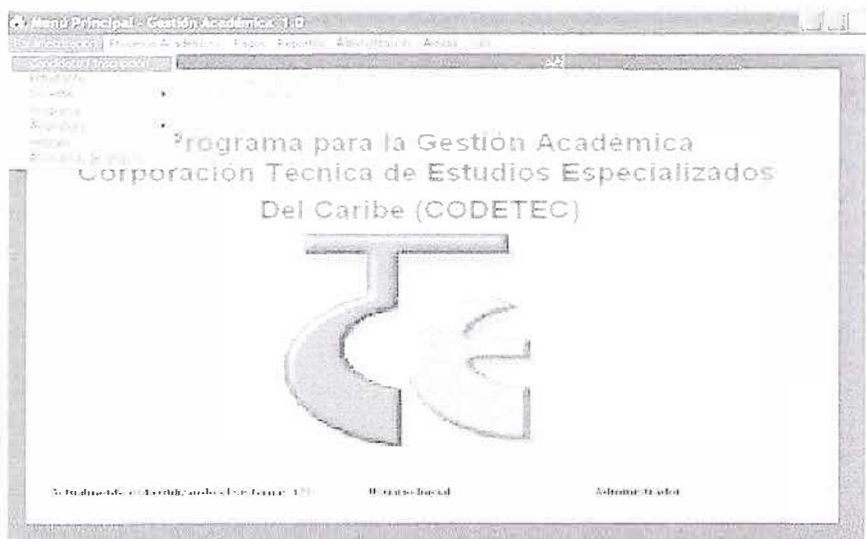


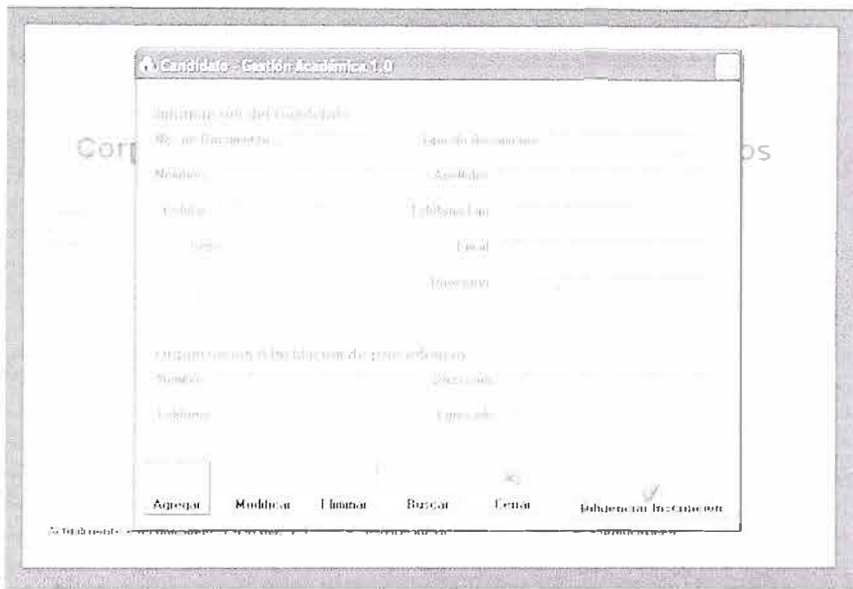
2. PARAMETRIZACIÓN

a. Candidato/Inscripción

Menú de acceso – Parametrización – Candidato / Inscripción

Permite realizar el proceso registro y actualización de la información de los candidatos, y el proceso de inscripción, que va desde la generación de volante de inscripción, hasta la oficialización de la inscripción.





Procedimiento

a. digite el documento del candidato

- Cedula
- Tarjeta de identidad
- Cedula de extranjería

b. Registre el numero de identificación de acuerdo al tipo de documento seleccionado, estos campos son obligatorios

c. En el campo nombre, digite el nombre del candidato, este es obligatorio diligenciarlo, su longitud máxima es de 40 caracteres



d. Luego en el campo apellidos, digite el apellido del candidato, este campo es obligatorio diligenciarlo

e. En el campo celular, se digita el celular del candidato

f. En el campo teléfono fijo, digite el teléfono fijo del candidato

g. En el campo sexo, seleccione el sexo del candidato

h. En el campo E-mail digite el correo electrónico del candidato

i. En el campo dirección, digite la dirección de residencia del candidato

j. En el recuadro de Organización o institución de procedencia se diligenciará datos básicos de la institución, de la cual es el candidato.

En el campo de nombre, digite el nombre de la institución de la cual viene el candidato

- En el campo dirección, digite la dirección de la institución de donde viene o es egresado el estudiante
- En el campo teléfono, digite el teléfono fijo de la institución
- Por último en el campo de selección, marcar (Si / No) es egresado el candidato de la institución.

k. Luego de haber diligenciado todos estos campos, damos click en el botón

Guardar para grabar la información



l. En caso de que el usuario desee cancelar la creación del nuevo registro,

debe presionar el botón Cancelar

m. Dado el caso que el registro sea guardado con éxito, se activara el botón



el cuál tiene como función mostrar en pantalla un nuevo formulario para diligenciar la inscripción del candidato como son:

- Numero de documento
- Tipo de documento
- Nombres
- Apellidos

Estos datos son llenados automáticamente por el sistema para agilizar el proceso de inscripción.



En cuanto al resto de los datos que se deben diligenciar como obligatorios están:

Código del programa, este código se diligencia con el botón

Buscar Programa

que al presionarlo mostrara en pantalla una ventana de Búsqueda de Programas, con los distintos programas existentes en la institución. En esta ventana se escogerá el programa en el que el candidato desea inscribirse.

- La ventana de Búsqueda de Programas esta conformada de tres botones y una barra para desplazarse de programa en programa Con el

botón  **Asignar a Inscripción** se selecciona el programa después de

buscarlo con el botón  **Buscar** por medio esta opción se buscan los



programas por el código; igualmente con esta la barra



Luego de esto, para terminar el proceso de inscripción se genera

volante de inscripción del candidato con el botón Generar Volante el cual será en formato PDF

Gestionar Inscripción del Candidato

No. de Documento: Tipo de documento:

Nombre: Apellido:

Código del Programa: Nombre del Programa:

Buscar Programa

Valor multiplica.:

Programas Académicos

Código:

Nombre:

Duración: Semestre:

Regreso H.F.E.S.:

Buscar programa

Digite el código del programa

Aceptar Cancelar

Programas Asignar a Inscripción ☒ Buscar Cerrar

Administrador



b. Estudiante

Menú de acceso – Parametrización – Estudiante

Programa para la Gestión Académica
Corporación Técnica de Estudios Especializados

Nombre del Estudiante: _____ Fecha de Nacimiento: _____
Sexo: ☐ Masculino ☐ Femenino Código: _____
Fecha de Registro: _____ Estado Civil: _____
Expediente: _____

Por medio de este modulo se registra todo el proceso registro y actualización del estudiante.

Procedimiento

a. digite el documento del estudiante

- Cedula
- Tarjeta de identidad
- Cedula de extranjería



- b. Registre el numero de identificación de acuerdo al tipo de documento seleccionado, estos campos son obligatorios
- c. En el campo nombre, digite el nombre del estudiante, este es obligatorio diligenciarlo, su longitud máxima es de 40 caracteres
- d. Luego en el campo apellidos, digite el apellido del estudiante, este campo es obligatorio diligenciarlo
- e. En el campo celular, se digita el celular del estudiante
- f. En el campo teléfono fijo, digite el teléfono fijo del estudiante
- g. En el campo sexo, seleccione el sexo del estudiante
- h. En el campo E-mail digite el correo electrónico del estudiante
- i. En el campo dirección, digite la dirección de residencia del estudiante
- j. En el campo fecha de nacimiento, digite la fecha del estudiante con el siguiente formato (aaaa/mm/dd).
- k. En el campo estado civil, escoja la opción que se despliega dándole clic en la pestaña.
- l. En el campo tipo de estudio, digite el nivel de escolaridad del estudiante.
- m. Luego de haber diligenciado todos estos campos, damos click en el botón

[Guardar](#) para grabar la información



n. En caso de que el usuario desee cancelar la creación del nuevo registro,

debe presionar el botón Cancelar

c. Docente

- **Registro / Actualización**

Menú de acceso – Parametrización – Docente – Registro / Actualización

Permite el registro y actualización de la información de los docentes.

Programa para la Gestión Académica
Corporación Técnica de Estudios Especializados

Información Básica del Docente

No. de Documento Tipo de documento

Nombres Apellidos

Celular Teléfono Fijo

Sexo:
☒ Masculino
☐ Femenino

Email

Dirección

Información Profesional

Actualmente:



Procedimiento

- digite el documento del Docente.
 - Cedula
 - Tarjeta de identidad
 - Cedula de extranjería
 - Registre el numero de identificación de acuerdo al tipo de documento seleccionado, estos campos son obligatorios
 - En el campo nombre, digite el nombre del docente, este es obligatorio diligenciarlo, su longitud máxima es de 40 caracteres
 - Luego en el campo apellidos, digite el apellido del docente, este campo es obligatorio diligenciarlo.
 - En el campo celular, se digita el celular del docente.
 - En el campo teléfono fijo, digite el teléfono fijo del docente.
 - En el campo genero, seleccione el sexo del docente.
 - En el campo E-mail digite el correo electrónico del docente.
 - En el campo dirección, digite la dirección de residencia del docente.
 - en el campo de información profesional, digite las observaciones de su carrera profesional del docente.
-



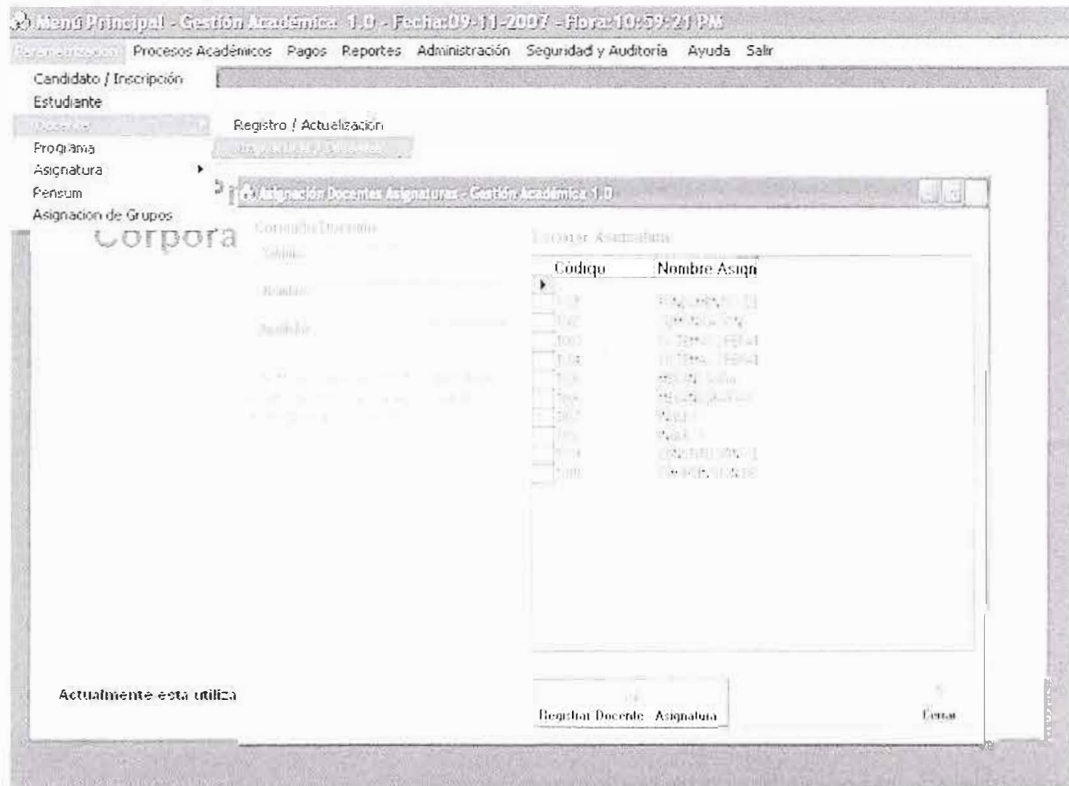
- Luego de haber diligenciado todos estos campos, damos click en el botón  para grabar la información
- En caso de que el usuario desee cancelar la creación del nuevo registro,

debe presionar el botón 

• Asignaturas / Docente

Menú de acceso – Parametrización – Docente – Asignaturas / Docente

Se realiza el proceso de asignación de asignaturas a un docente





Procedimiento:



Se empieza dándole clic sobre este botón para activar los campos del docente, se ilustra una ventana de mensaje indicando que puede diligenciar los campos y poder digitar el número de cédula.

- Cédula, en este campo se digita el número de identificación del docente y luego presionamos la tecla enter, luego se ilustrará una ventana de mensajes donde le comunica que puede escoger la asignatura.

Estos datos son llenados automáticamente por el sistema para agilizar el proceso de inscripción.

- Nombres
- Apellidos

Luego de que el sistema trae los datos del docente procedemos a seleccionar la con el Mouse la asignatura deseada, después de seleccionada la respectiva asignatura y todo el proceso sea exitoso guardamos por medio del botón



Dado el caso contrario que el proceso no se desee guardar presionamos el botón





d. Programa

Menú de acceso – Parametrización – Programa

Registre en esta opción la información de los programas académico

Procedimiento

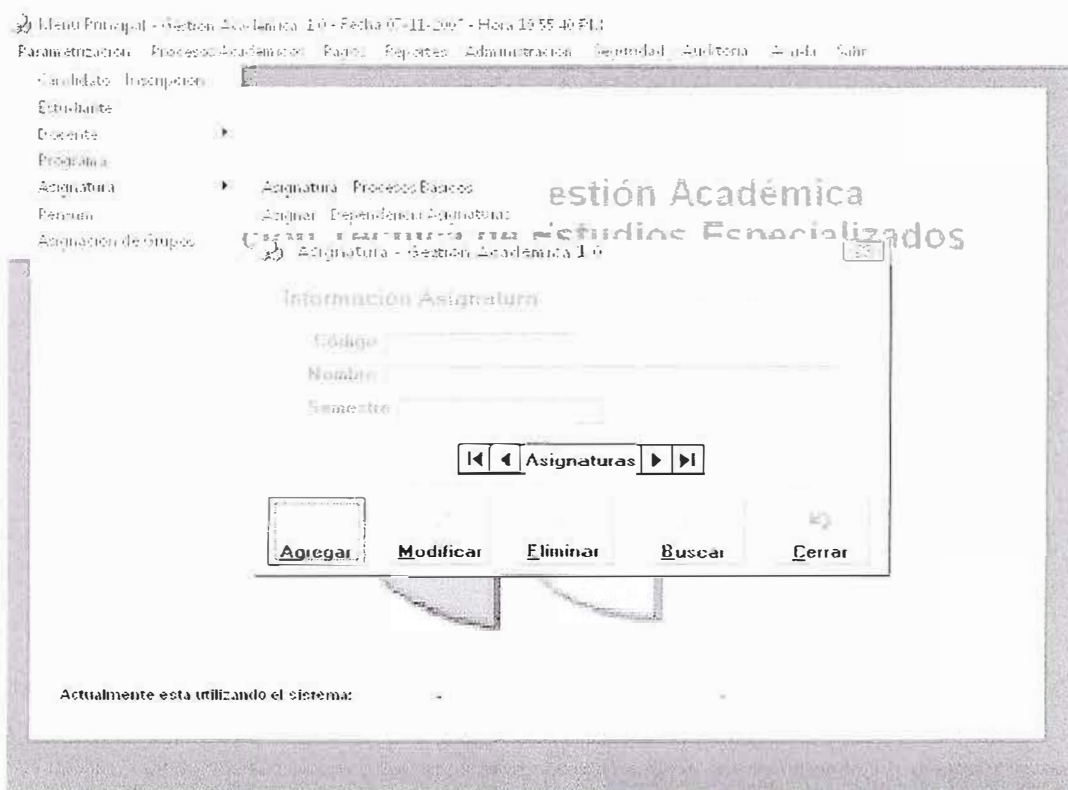
Por medio de este formulario se registran los datos de los programas que se encontraran dentro del pensum de la institución.

- El proceso de creación de los programas se inicia presionando el botón  el cual activa el resto de los campos del formulario, a excepción del campo código que lo genera el sistema como un consecutivo automáticamente.
- Luego en el campo nombre, digitamos el nombre del programa
- En el campo duración, digitamos el tiempo de duración del programa

e. Asignatura

- **Asignatura / Procesos Básicos**

Menú de acceso – Parametrización – Asignatura – Asignatura / Procesos Básicos



Permite el registro y actualización de la información de las asignaturas.

- Este proceso se inicia presionando el botón  el cual activa el resto de los campos del formulario
- El campo código no es necesario digitarlo ya que el sistema genera el código automáticamente



- En el campo nombre, digitamos el nombre de la asignatura
- En el campo semestre desplegamos la lista y seleccionamos el semestre correspondiente
- Para guardar asignatura creada , presionamos el botón 
- Para modificar los datos de una asignatura creada , primero se trae en pantalla sus datos por medio del botón de búsqueda  para luego cambiar los datos correspondientes presionando el botón 
- Luego de los cambios realizados presionamos el botón  para grabar la información
- De lo contrario presionamos el botón  para abortar la operación
- Para eliminar una asignatura , primero buscamos la signatura con el botón de búsqueda ,luego presionamos el botón 
- Por ultimo para salir del modulo presionamos el botón 
- **Asignar / Dependencia Asignaturas**

Menú de acceso – Parametrización – Asignatura – Asignatura / Procesos Básicos

[Candidato](#) - [Inscripción](#) - [Estudiante](#) - [Docente](#) - [Programa](#) - [Asignatura](#) - [Persona](#) - [Asignación de Grupos](#)

[Asignatura](#) - [Procesos Básicos](#) - [Asignar Dependencia Asignaturas](#)

Sección Académica

Asignación Técnica de Estudios Especializados

[Dependencia de Asignaturas](#) - Sección Académica 1.0

Asignación Dependencias - Asignaturas

Correo:

Preservación:

Alternar:

Nueva Asignación

Cerrar

Los usuarios de esta aplicación pueden asignar, modificar, eliminar y consultar las dependencias entre asignaturas.

Por medio de este modulo se realiza la asignación de las dependencias a las asignaturas, Correoquisito, Prerrequisito y Alterna, según sea el caso.

Procedimiento

- El proceso de asignación de dependencia de asignaturas se inicia

presionando el botón 

- Seleccionar la asignatura dependiente **Asignatura Dependiente**

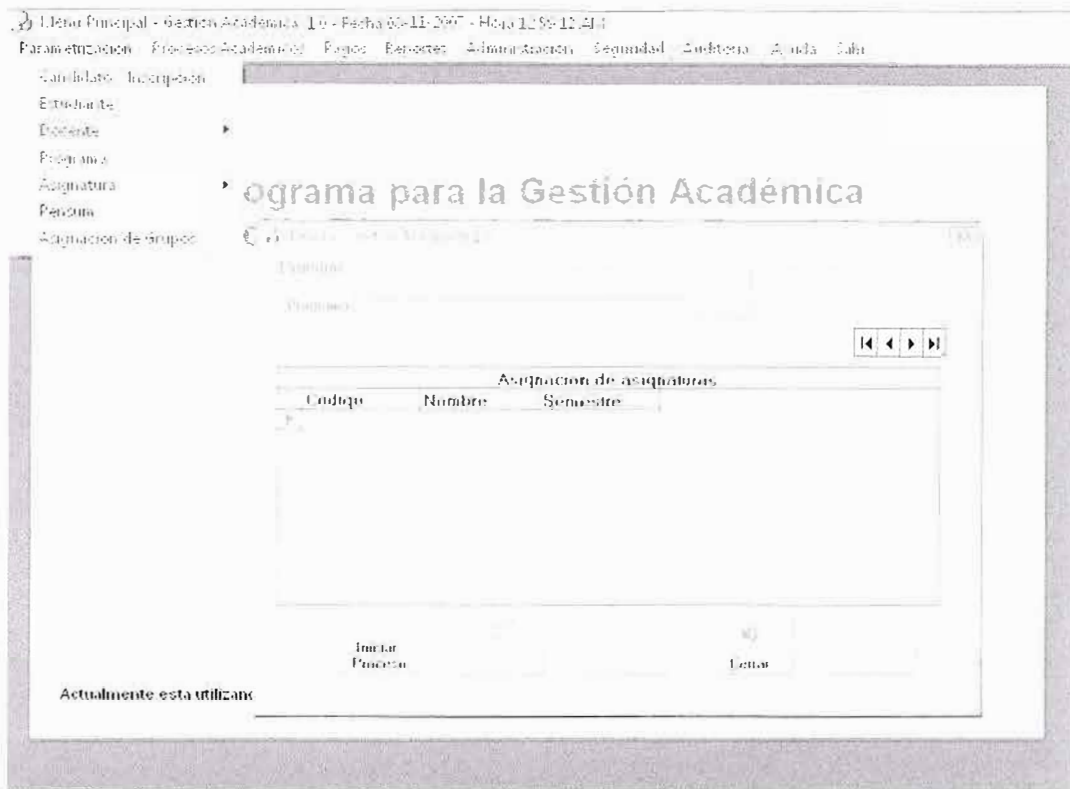
se despliega una ventana donde se permite asignar la asignatura, de igual forma se realiza con el botón **Asignatura Independiente**



- Escoger el tipo de dependencia Correoquisito, Prerrequisito y Alterna.
- Si es Alterna, escoger el programa correspondiente. 
- El proceso culmina al presionar el botón  el cual guarda la asignación realizada.
- Si se desea abortar el proceso de asignatura se selecciona el botón 

f. Pensum

Menú de acceso – Parametrización – Pensum



Permite la asignación de asignaturas a un programa con el objetivo de conformar su Pensum Académico.

Procedimiento

- Este modulo se inicializa presionando el botón  para activar el campo programa y el botón buscar programa  para luego comenzar el proceso de asignación de pensum.



- Se presiona el botón buscar, la cual permite realizar una búsqueda donde se despliega una ventana visualizando los datos del programa para luego realizar una asignación
- Seleccionar el botón  **Seleccionar Asignatura** permite la escogencia de una asignatura de la lista.
- Seleccionar el botón  **Guardar Registro** para permitir guardar los cambios realizados.
- Para cancelar el proceso de creación del pensum presionamos el botón  **Cancelar**
- Para salir del modulo se debe presionar el boton  **Cerrar**

g. Asignación de Grupos

Menú de acceso – Parametrización – Asignación de grupos

Inicio Principal - Gestión Académica 19 - Fecha: 00-11-2007 - Hora: 11:59:12 AM

Parametrización - Procesos Académicos - Pagos - Reportes - Administración - Seguridad - Archivos - Ayuda - Salir

[Candidato](#)
[Inscripción](#)
[Estudiante](#)
[Docente](#)
[Programa](#)
[Asignatura](#)
[Persona](#)
[Asignación de Grupos](#)

Programa para la Gestión Académica

Código	Nombre	Documento	Nombre	Apellidos

Actualmente está utilizando el sistema:

Permite realizar la asignación de grupos. Organizado a los estudiantes teniendo en cuenta: Programa - Semestre - Docente – Asignatura.

- Este modulo inicia al presionar el botón

Nuevo

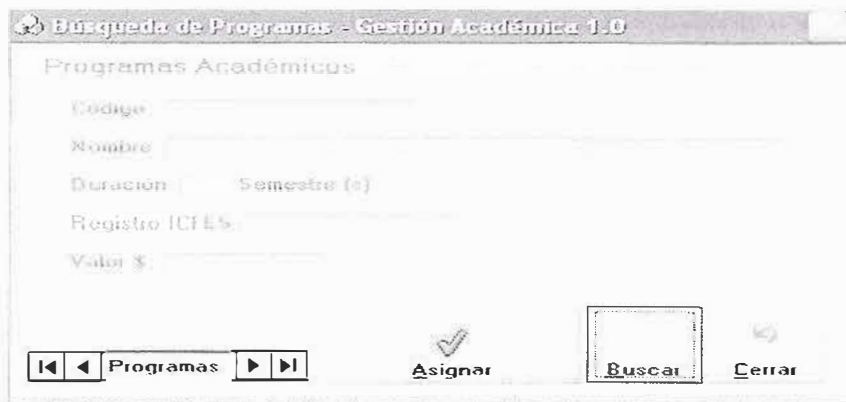
- Luego seleccione el semestre desplegando el campo semestre

Semestre / Tipo

- Siguiente des semestre se debe seleccionar el grupo desplegando el campo grupo

Grupos

- Después de seleccionar los dos criterios anteriores se debe escoger el programa por medio del botón de búsqueda programas  este botón mostrara una pantalla emergente para la búsqueda del programa ,esta pantalla esta conformada por tres botones
- Asignar Apertura  : Selecciona el programa deseado
- Buscar  : Busca el programa
- Cerrar  : Se utiliza para salir de modulo



La imagen muestra una ventana de software titulada "Búsqueda de Programas - Gestión Académica 1.0". Dentro de la ventana, hay un formulario con los siguientes campos de texto: "Codigo:", "Nombre:", "Duración:", "Semestre (s):", "Registro ICIES:", y "Valor \$:". En la parte inferior de la ventana, hay una barra de herramientas con los siguientes elementos: un botón "Programas" con flechas de navegación, un botón "Asignar" con una marca de verificación, un botón "Buscar" con una lupa, y un botón "Cerrar" con una flecha hacia afuera.

- Después de seleccionar el programa, debemos escoger la asignatura y el respectivo docente al cual se le asignara
- Siguiendo a el proceso de selección de la asignaturas y el docente ,se debe presionar el boton  para guardar la asignación
- Para generar una nueva asignación , ya sea para otra semestre o programa debemos presionar el botón  e iniciamos de nuevo el proceso



- Para cancelar la asignación presionamos el boton

Cancelar

- Salir del modulo presionamos

Cerrar

3. PROCESOS ACADÉMICOS

Menu Principal - Gestion Academica 1.0 - Fecha 05-11-2007 - Hora 13:59:12.411

Parametrización - Procesos Académicos - Pagos - Pagos - Administración - Seguridad - Auditoria - Ayuda - Salir

Matricula ▶ Generar Volante Matricula

Notas ▶ Cargar Matricula

Pro Corporac

Generar Volante Matricula - Gestion Academica 1.0

Volante de Matricula

Código 72286513

Listado de Programas	
Programa	Valor

Cerrar

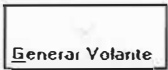
Actualmente esta utilizando el sistema:



a. Matrícula

- **Generar Volante Matrícula**

Por medio de este modulo se generan los volantes para matricula de los candidatos que ya han cancelado el valor de la inscripción

- En el campo cedula ,se digita la cedula del candidato para realizar la búsqueda por medio del boton  , el candidato debe haber cancelado el volante de inscripción previamente para poder generarle el volante de matricula
- Se selecciona el programa y se presiona el boton  para la impresión del volante
- En caso de que el candidato desee cambiar de programa, debe presionar el boton  , este boton mostrara una ventana emergente para seleccionar el nuevo programa



- Esta ventana se conforma por los datos del programa y tres botones:

- El botón  : Utilizado para adicionar el nuevo programa en la ventana de volante de matrícula

- El botón  : Utilizado para buscar los programas por el código

- El botón  : Para salir de la ventana de búsqueda de programas , o en su defecto del modulo generar volante

- Luego del proceso de selección del nuevo programas, se escoge de nuevo el programa y se presiona el boton Generar Volante 

• Oficializar Matrícula

Menú Principal - Gestión Académica 1.0 - Fecha: 09-11-2007 - Hora: 10:59:21 PM

Parametrización | **Programas Académicos** | Pagos | Reportes | Administración | Seguridad y Auditoría | Ayuda | Salir

Notas | **Generar Volante Matrícula** | **Oficializar Matrícula**

Oficializar Matrícula - Gestión Académica 1.0 09-11-2007 11:04:37 PM

Información básica del Estudiante

P Corpora

Matrícula: Código Matrícula:

Extrínseca: Tipo de documento:

Identificación: Apellido:

Nombre: Fecha de nacimiento:

Sexo: Fecha de ingreso:

Matrícula: Fecha de ingreso:

Extrínseca: Fecha de ingreso:

Fecha Matrícula: 05/05/2007 Estado Civil: Tipo Estudiante:

Fecha Matrícula: Fecha de Nacimiento: 05/05/2007

Programa: Fecha de ingreso:

Semestre: Fecha de ingreso:

Grupos:

Código: Asignatura: Semestre:

Actualmente esta utiliz



Procesos

En este modulo se realizara el proceso de legalización académica de la matricula previamente cancelada, es decir, que se encuentre el volante de matricula cancelado y el pago legalizado en el modulo de Pagos/ Aplicar pagos Inscripción

- Para iniciar este proceso el campo volante, digitamos el código del volante ya cancelado y aplicado, el sistema automáticamente mostrara el resto de los datos ya diligenciados por el candidato o estudiante
- En el campo el sistema generara un código automático
- En este modulo se encuentran unos campos nuevos a diligenciar y otros ya diligenciados como son:
 - **05/05/2007** aquí el se escoge la fecha de nacimiento del estudiante
 - se despliega la barra para seleccionar el estado civil del estudiante
 - este campo ya viene con la información correspondiente al estado del estudiante, estos estados son dos: Regular, que son los estudiantes que



aprobaron todas sus asignaturas y cursaran el semestre con todas las asignaturas correspondientes a ese semestre y los No regulares, que son estudiantes que se encuentran en nivelación de asignaturas de semestres menores o mayores al que se esta matriculando, ya sea por reprobado, reintegro o traslado de otra institución

- Fecha Matricula 05/05/2007 esta fecha la dará el sistema automáticamente

- Hora Matricula la hora será tomada por el sistema automáticamente

- En el campo Semestre / Tipo Primero se diligenciará automáticamente en el caso de que el estudiante sea de tipo Regular, de lo contrario se activará la selección de otra semestre

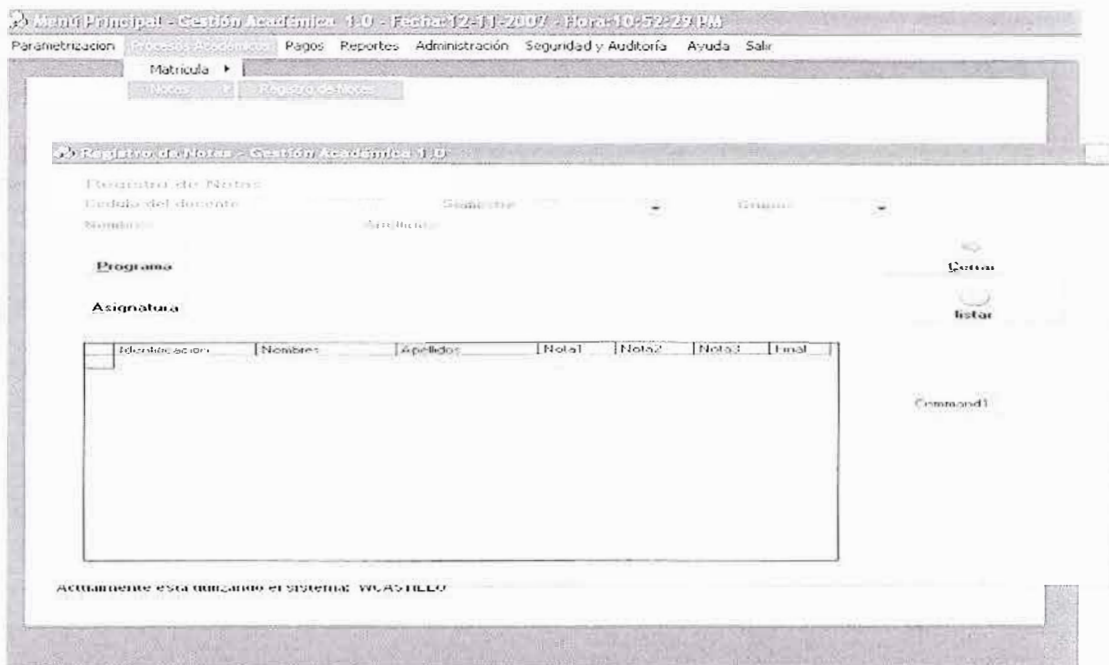
- En el campo Programa se mostrara el programa que fue escogido por el candidato en el momento de generarle el volante de matricula, dado el caso que se desee cambiar de programa presionamos el botón que se encuentra al lado izquierdo del campo este desplegara una pantalla emergente que permitirá seleccionar el nuevo programa



- Después de escoger el programa, sesionamos las asignaturas a cursar por el candidato dependiendo de su estado (Reprobado ó Aprobado)
- Para guardar la matricula con todos los datos ya diligenciados presionamos el botón  Guardar Matricula
- Para cancelar el proceso presionamos  Cancelar
- Por ultimo para salir del modulo presionamos el botón  Cerrar

b. Notas

- Registro de Notas



Manifi Principal - Gestión Académica 1.0 - Fecha: 12-11-2007 - Hora: 10:52:29 AM

Parametrización | **Proceso Académico** | Pagos | Reportes | Administración | Seguridad y Auditoría | Ayuda | Salir

Matricula | **Notas** | Registro de Notas

Registro de Notas - Gestión Académica 1.0

Excentro de Notas: Código del docente: Nombre: Apellido: Semestre: Grupo:

Programa

Asignatura

Identificación	Nombre	Apellidos	Nota1	Nota2	Nota3	Total

Cerrar

Listar

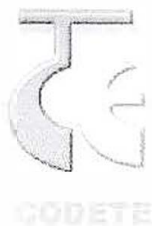
Comando1

ACUERDARSE ESTE ORDENAMIENTO DE SISTEMAS: WUAS/ILLO



Este modulo permitirá digitar las notas por cada estudiante con su respectivas asignaturas, como también su actualización

- Para inicializar el ingreso de las notas por estudiante demos primero filtrar los datos para la búsquedas según lo requerido por el docente
- Para realizar el ingreso buscamos inicialmente al docente por su cedula en el campo **Cedula del Docente**
- Luego de que el sistema muestre en pantalla los datos del docente que desea realizar el ingreso de las notas, desplegamos la barra **Semestre** y seleccionamos el semestre deseado
- Después de seleccionar el semestre , debemos seleccionar el **Grupos** al que pertenecen los estudiantes que se les realizara el ingreso o modificación de las notas
- Luego de la escogencia de los criterios anteriores procedemos realizar la búsqueda del programa deseado por medio del botón Programa
- Luego escogemos la asignatura a la cual se le ingresara las notas ,con el botón Asignatura
- Para completar el proceso debemos presionar el botón  **listar**



para que se listen todos los estudiantes del **Semestre, Grupo, Programa y Asignatura** escogidos

- Para ingresar o modificar las notas de cada estudiante, primero lo seleccionamos en la tabla

	Identificación	Nombres	Apellidos	Nota1	Nota2	Nota3	Final
▶	72276818	FREDY	RENYZ	5	3	1	3,0
	72170914	NARCISO	CANTILLO	5	5	4	4,7
	72429396	JOSÉ	DE LA CRUZ	5	2	3	3,3

Esta selección mostrara una pantalla emergente con tres campos de ingreso de las notas a modificar

- Para modificar la notas que se desee, primero nos ubicamos en el campo deseado y digitamos la nota, de igual forma se realiza en el resto de los campos, **Nota1, Notas2, Nota3**


**Actualizar
Notas**

- Para guardar los cambios presionamos el botón



- Para salir del modulo presionamos el botón **Cerrar**

También permite consultar los registros por **Nombre de Usuario** y por **Operaciones Realizadas**



- Para realizar la consulta por fecha se debe seleccionar **Búsqueda Por Fecha**

rectio inter!

- Luego de la selección digitamos en los campos y

las fechas entre las que se desea ver los registros, la fecha

final siempre debe ser mayor que la fecha inicial

- Después de digitar el rango de fechas presionamos el botón **Buscar** para mostrar los registros en la cuadrícula

[illegible]

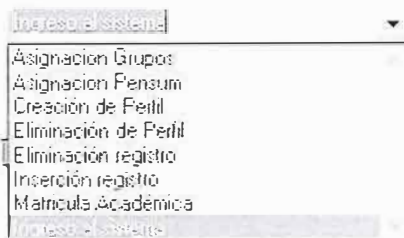


- Si la consulta que se quiere realizar es por **Nombre de Usuario** , debemos seleccionar el recuadro **Búsqueda Por Nombre Usuario** para luego desplegar



la lista de los usuarios

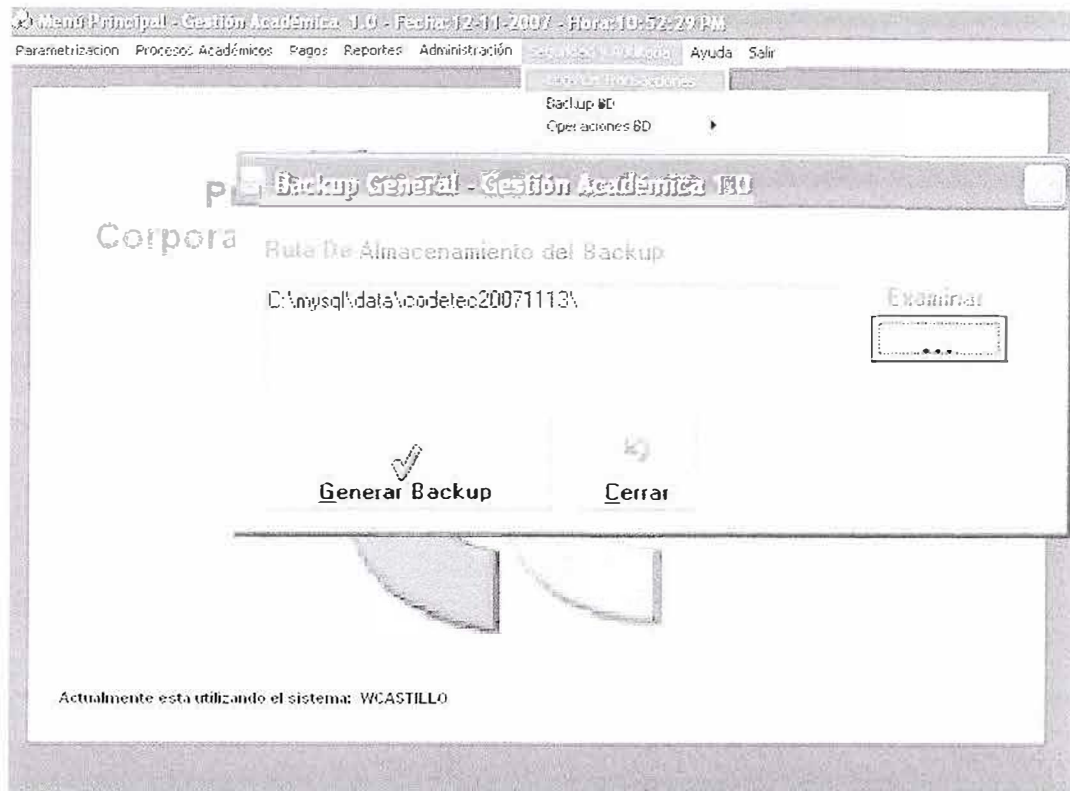
- Después de la selección presionamos el botón **Buscar** para listar los registros a consultar
- Para realizar la búsqueda por **Operaciones Realizadas**, seleccionamos el recuadro **Búsqueda Por Operación Realizada** y luego desplegamos la lista de todas las posibles operaciones que puede realizar un usuario



- para salir del sistema presionamos el botón **Cerrar**



- **Backup BD**

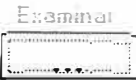


Este modulo de seguridad del sistema permite generar una copia de seguridad de la base de datos del aplicativo, con el fin de mantener siempre una copia para recuperar información en caso de daños o fallas del sistema

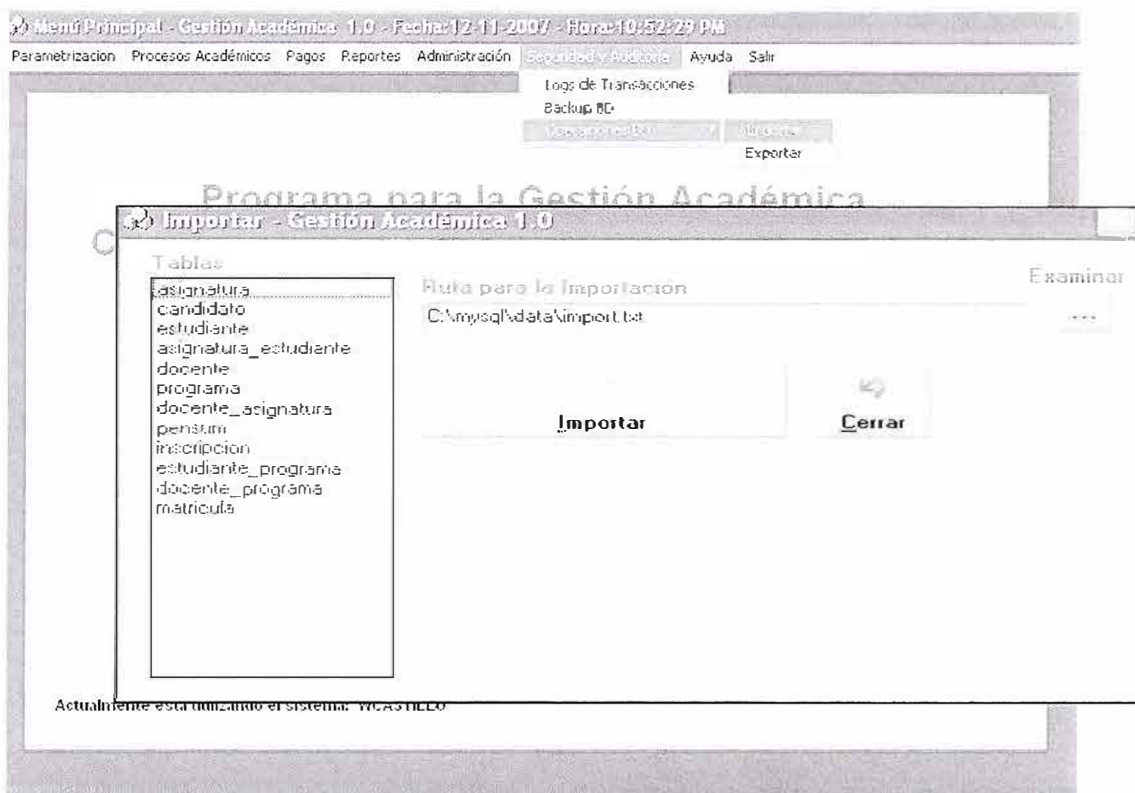
Procesos

- Para realizar el Backup de la base de datos , inicialmente debemos seleccionamos la ruta donde queremos guardar la copia. **Nota:** La primera vez que se genera le Backup debemos seleccionar la ruta , ya que para el próximo Backup el sistema nos trae la ruta anteriormente seccionada.



- La ruta la buscamos presionando el botón 
- Luego de escogida la ruta, presionamos el botón  para generar la copia
- Por ultimo para salir del modulo de Backup presionamos el boton 

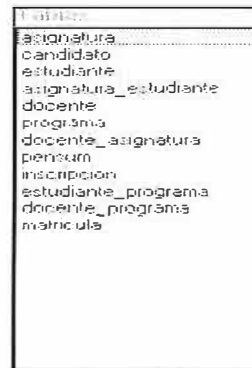
• IMPORTAR





Este modulo nos permite importar las tablas de la base de datos, con el fin de poder recuperación de datos exportados

- Para la importación debemos tener archivos txt ya exportados previamente
- Primero seleccionamos la ruta de donde vamos a traer la tabla a importar
- Luego seleccionamos la tabla en la lista de las tablas, debe coincidir con el nombre de la tabla que se importara



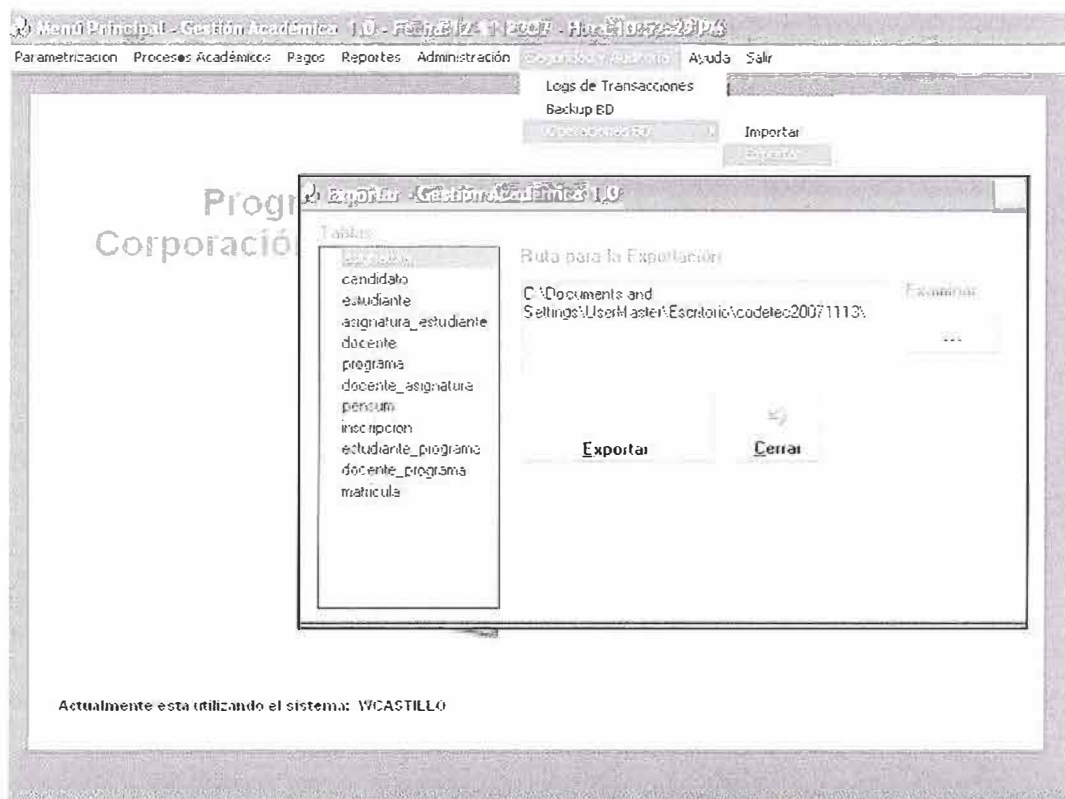
- Para realizar el proceso de importación después de ya seleccionada la

ruta presionamos la tecla Importar

- Para salir del modulo de importación presionamos el botón Cerrar



- **EXPORTAR**

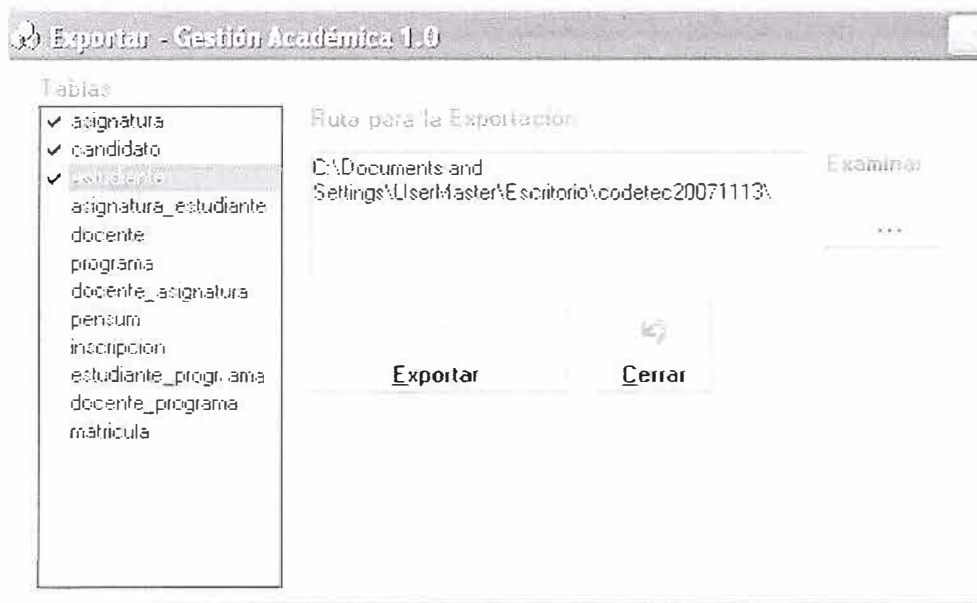


En este modulo se realiza la exportación de las tablas de la base de datos con su respectiva información, con el fin de generar copias de seguridad individuales en caso de fallas del sistema



Procesos

- Para realizar la exportación de las tabla primero debemos seleccionar las tablas que se desea exportar



- Después de seleccionar las tablas, procedemos a escoger la ruta en la que se desea guardar la exportacion
- Luego de seleccionar la ruta para exportar, presionamos el botón

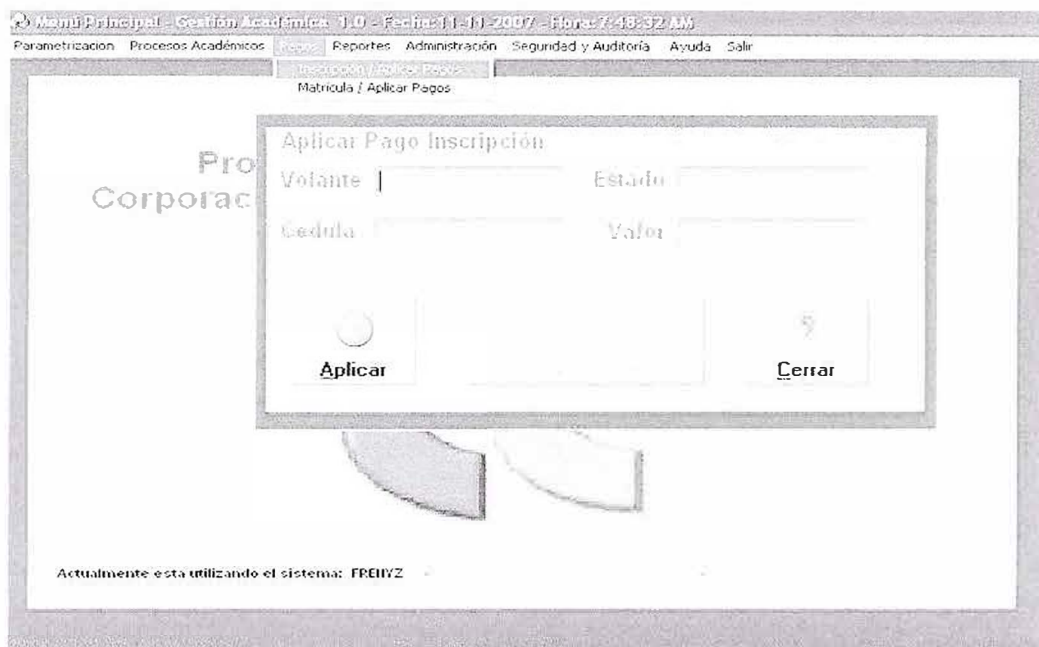
Exportar

Exportar

- por ultimo para salir del modulo presionamos el botón Cerrar

4. PAGOS

- **Inscripción / Aplicar Pagos**



The screenshot shows a web application interface. At the top, a status bar reads 'Modo Principal - Gestión Académica 1.0 - Fecha: 1-11-2007 - Hora: 7:48:32 AM'. Below this is a navigation menu with links: 'Parametrización', 'Procesos Académicos', 'Inscripción / Aplicar Pagos' (highlighted), 'Reportes', 'Administración', 'Seguridad y Auditoría', 'Ayuda', and 'Salir'. The main content area features a large, semi-transparent watermark of the CODETE logo. Overlaid on this is a form titled 'Aplicar Pago Inscripción'. The form contains four input fields: 'Volante' (with a vertical line icon), 'Estado', 'Cedula', and 'Valor'. At the bottom of the form are two buttons: 'Aplicar' (with a circular arrow icon) and 'Cerrar' (with a square icon). At the very bottom of the page, a small text line states 'Actualmente está utilizando el sistema: FREIYZ'.

Procesos

Por medio de este modulo se aplicara y legaliza el pago de la inscripción del candidato en el programa escogido, previo a este modulo el candidato de tener generado un volante de inscripción por medio del modulo de Inscripción/candidato.

- Para aplicar el pago se debe digitar en el campo volante, el código del volante que se genera en el modulo de Inscripción



- Luego de la digitación, presionamos el botón aplicar  para aplicar el pago

- Para la legalización del pago previamente aplicado presionamos el botón



- Por ultimo para salir del modulo presionamos el botón



- **Matrícula / Aplicar Pagos**

Menú Principal - Gestión Académica 1.0 - Fecha: 11-11-2007 - Hora: 7:48:32 AM

Parametrización Procesos Académicos Pagos Reportes Administración Seguridad y Auditoría Ayuda Salir

Inscripción / Aplicar Pagos

Programa Corporación

Aplicar Pago Matrícula

Volante Estado

Cédula Valor

Actualmente está utilizando el sistema: FREIYZ

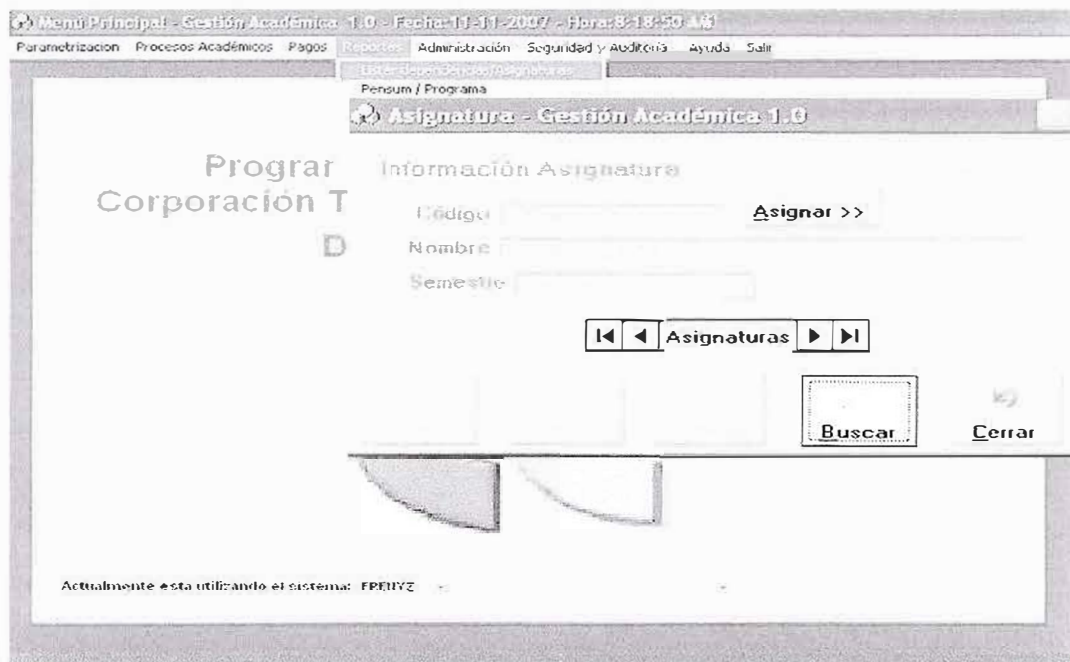
Por medio de este modulo se aplica el pago de la matricula del candidato, previo a este proceso al candidato se le debe haber generado el volante de matricula en el modulo Inscripción/Generar Volante de Matricula



- Para aplicar el pago de la matricula se debe digitar el campo  el numero del volante de matricula previamente generado
- Luego de digitar el código presionamos el botón  para aplicar el pago
- Por ultimo para salir del modulo presionamos el botón 

5. REPORTES

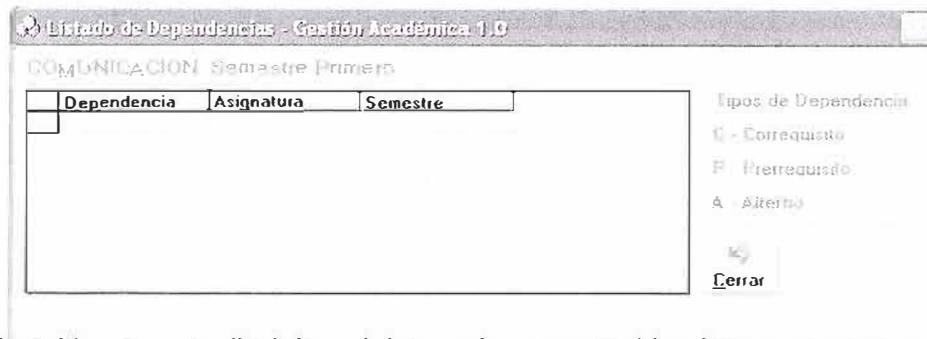
- **Listar dependencias / Asignaturas**



Este modulo nos permitirá ver el listado de las dependencias de las asignaturas, ya sea: Correquisito, Prerrequisito y Alterna

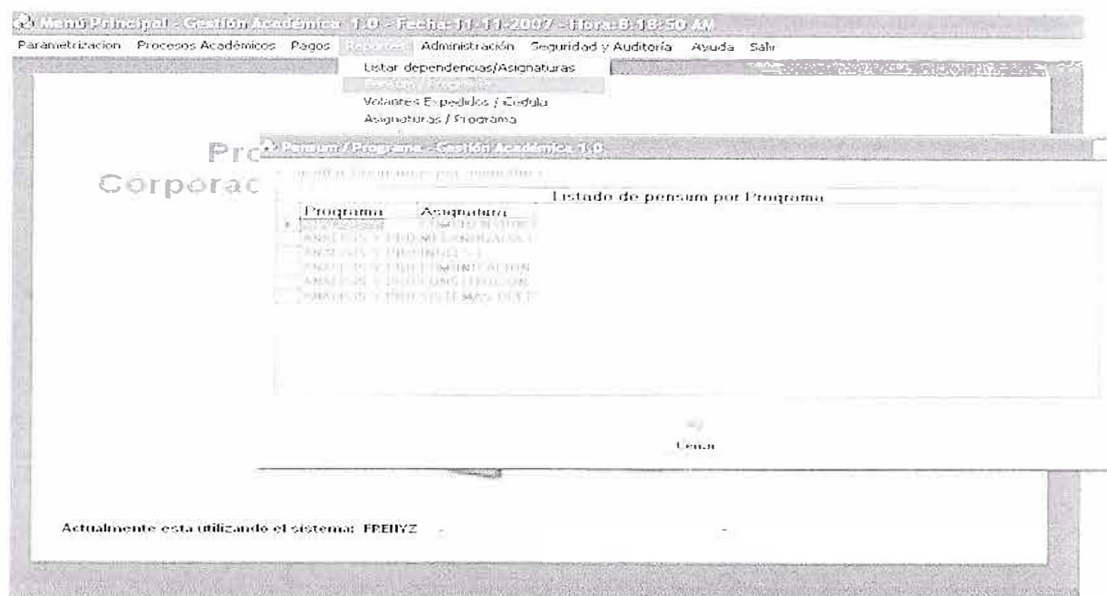
Procesos

- En este modulo podremos consultar las dependencias a las que pertenece cada asignatura
- El modulo esta formado por tres botones y una barra para desplazarse de una en una asignatura
- El botón  es utilizado para buscar la asignatura que se desea consultar
- La barra  es utilizada para ver una asignatura tras otra, como también ir al primer y ultimo registro
- El botón Asignar >> mostrar en pantalla una pantalla emergente permitiendo la visualización de la dependencia la que pertenece la asignatura y el semestre



- Por ultimo para salir del modulo presionamos el botón

- **Pensum / Programa**

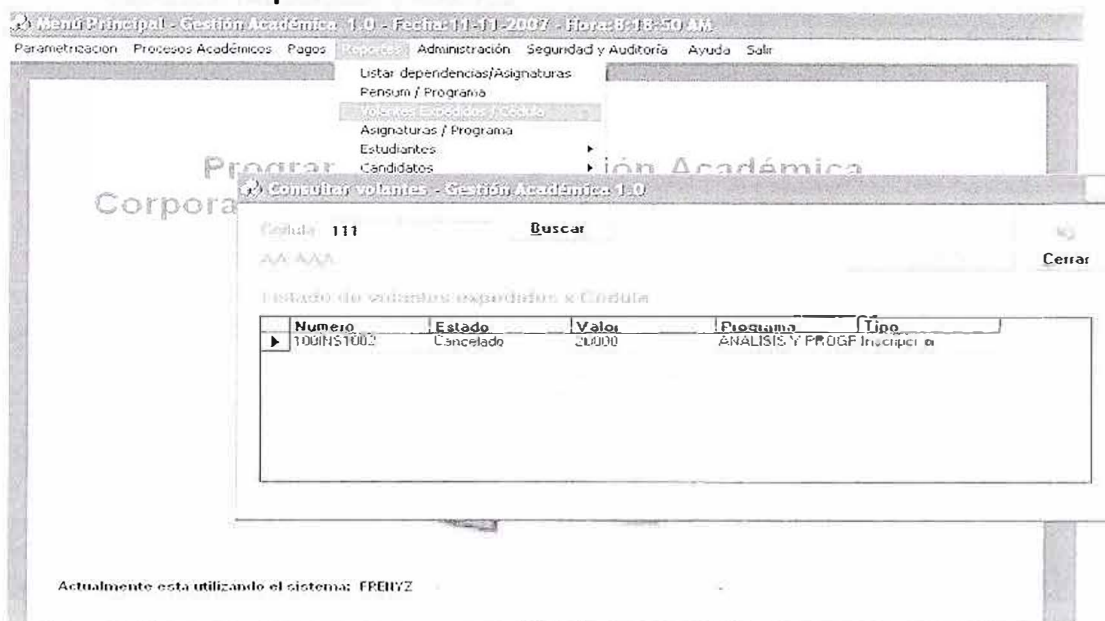


Este modulo nos permite visualizar el pensum por cada programa con sus asignaturas

- para salir presionamos el botón



- **Volantes expedidos / Cédula**





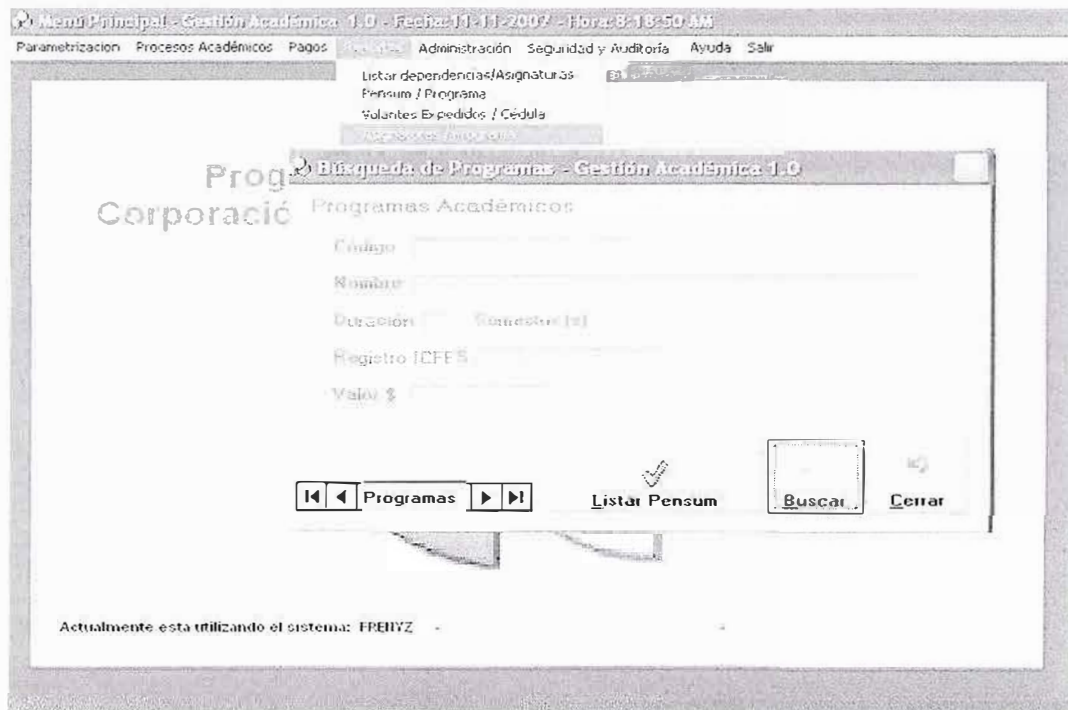
Este modulo nos permite consultar todos los volantes expedidos por cada usuario, con sus especificaciones:

- Numero del Volante
- Estado del volante
- Valor por el cual se genero el volante
- Programa
- El tipo de volante , ya sea: Inscripción o Matricula

Procesos

- Para iniciar la consulta debemos digitar el numero de cedula del candidato o estudiante en el campo
- Como siguiente paso para la consulta presionamos el botón
- En caso de requerirse el duplicado de un volante, podemos generar por medio del botón , pero antes debemos seleccionar el volante al cual se le generara el duplicado
- Para salir del modulo de consulta presionamos el botón

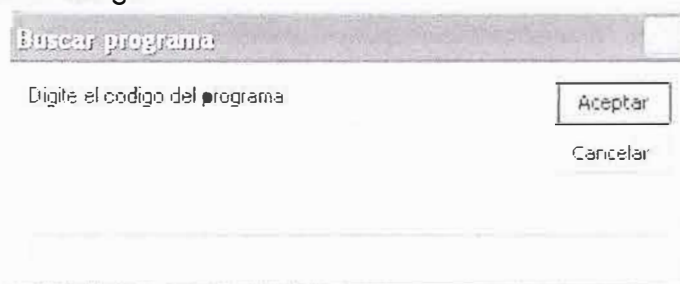
- **Asignaturas/programas**



En este modulo podremos consultar las asignaturas que pertenecen a cada programa y su semestre

Procesos

- Para realizar la consulta debemos iniciar buscando el programa deseado presionando el botón , la búsqueda del programa se realiza por el código





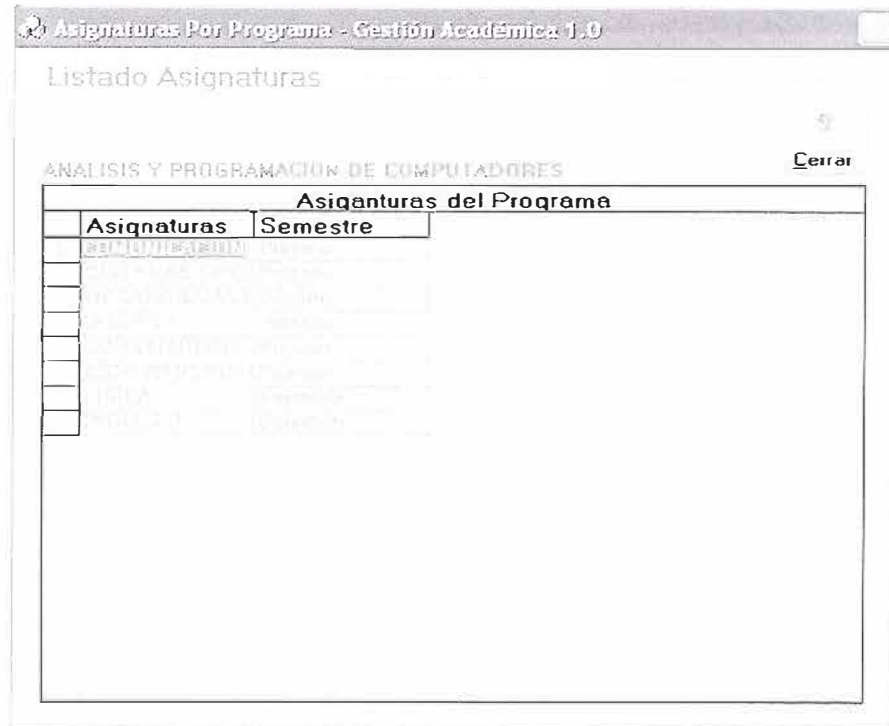
- Seguidamente después de realizada la búsqueda presionamos el botón



Listar Pensum

para de visualizar una pantalla con todas las

asignaturas que se encuentran relacionas con el programa consultado

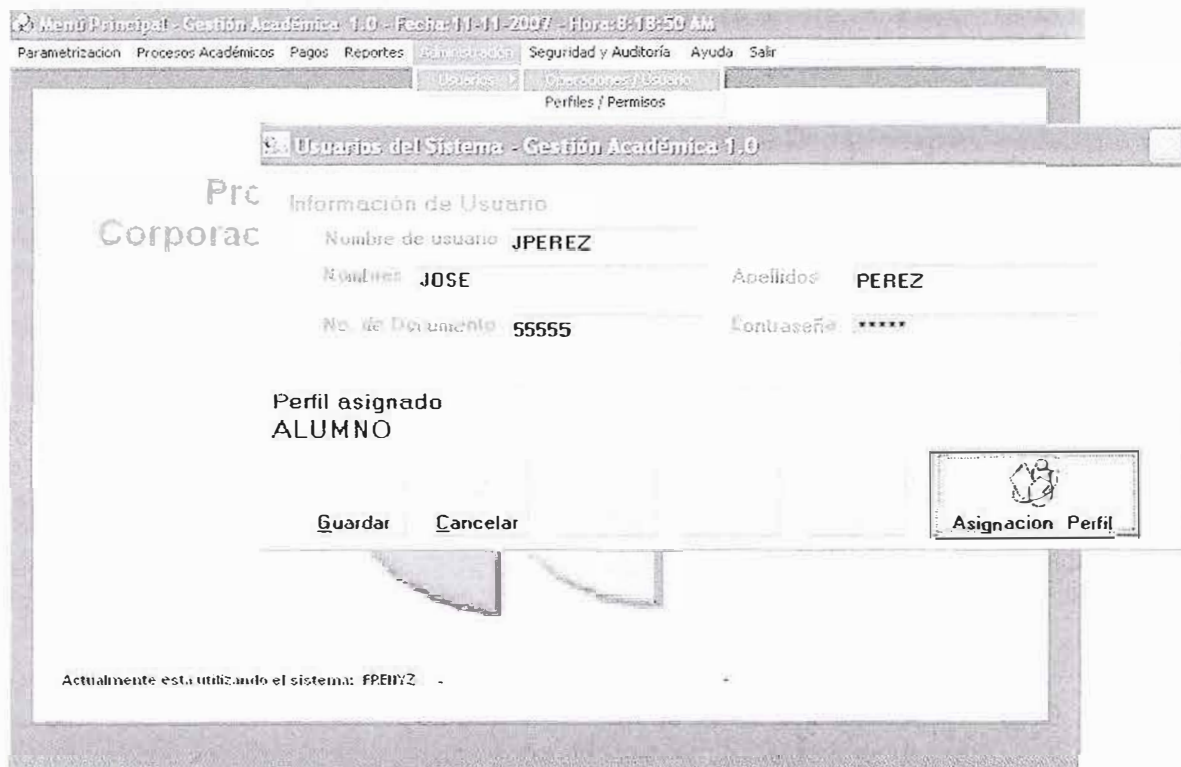


- Para salir de la pantalla de consulta presionamos el botón



6. ADMINISTRACIÓN

- **Usuarios/operaciones/usuarios**



Menú Principal - Gestión Académica 1.0 - Fecha: 11-11-2007 - Hora: 8:18:50 AM

Parametrización Procesos Académicos Pagos Reportes Administración Seguridad y Auditoría Ayuda Salir

Usuarios Operaciones/Usuarios Perfiles / Permisos

Usuarios del Sistema - Gestión Académica 1.0

Pro Corporac

Información de Usuario

Nombre de usuario **JPEREZ**

Nombre **JOSE** Apellidos **PEREZ**

No. de Documento **55555** Contraseña: *****

Perfil asignado
ALUMNO

Guardar Cancelar

 **Asignación Perfil**

Actualmente está utilizando el sistema: **JPEREZ**

En este modulo se realizara la creación de los usuarios del sistema, como también la asignación de los perfiles, con el fin de restringir el acceso a ciertos módulos del sistema

Procesos

- para la creación de usuario y asignación de su perfil, iniciamos presionando el botón **Agregar**
- Luego en el campo **Nombre de usuario JPEREZ** digitamos el nombre con el que el usuario iniciara la sección

- En los campos **Nombres** **JOSE** y **Apellidos** **PEREZ**

Digitamos el nombre y apellidos del usuario

- En el campo **No. de Documento** **55555** digitaremos el numero de cedula del usuario

- En el campo **Contraseña** ********* no es necesario digitar contraseña ya que el sistema por defecto asigna la cedula del usuario como contraseña. El usuario en el caso que desee cambiar la contraseña lo puede hacer por el formulario de inicio de sección

- Para la asignación del perfil del usuario presionamos el botón



Asignacion Perfil

este botón desplegara una pantalla con lo diferentes perfiles para la asignación



Código	Nombre Perfil
500	SECRETARIA
100	ADMINISTRADOR
200	PROFESOR
300	ALUMNO
400	CELADOR

Asignar perfil

- Para asignar el perfil al usuario en la pantalla de **PERFILES** primero debemos seleccionar el perfil y luego presionar el botón para



de esta forma cargar el perfil al usuario en la pantalla de creación de usuarios

- Para guardar el usuario creado presionamos el botón **Nota:** No se puede guardar un usuario sin antes asignarle un perfil

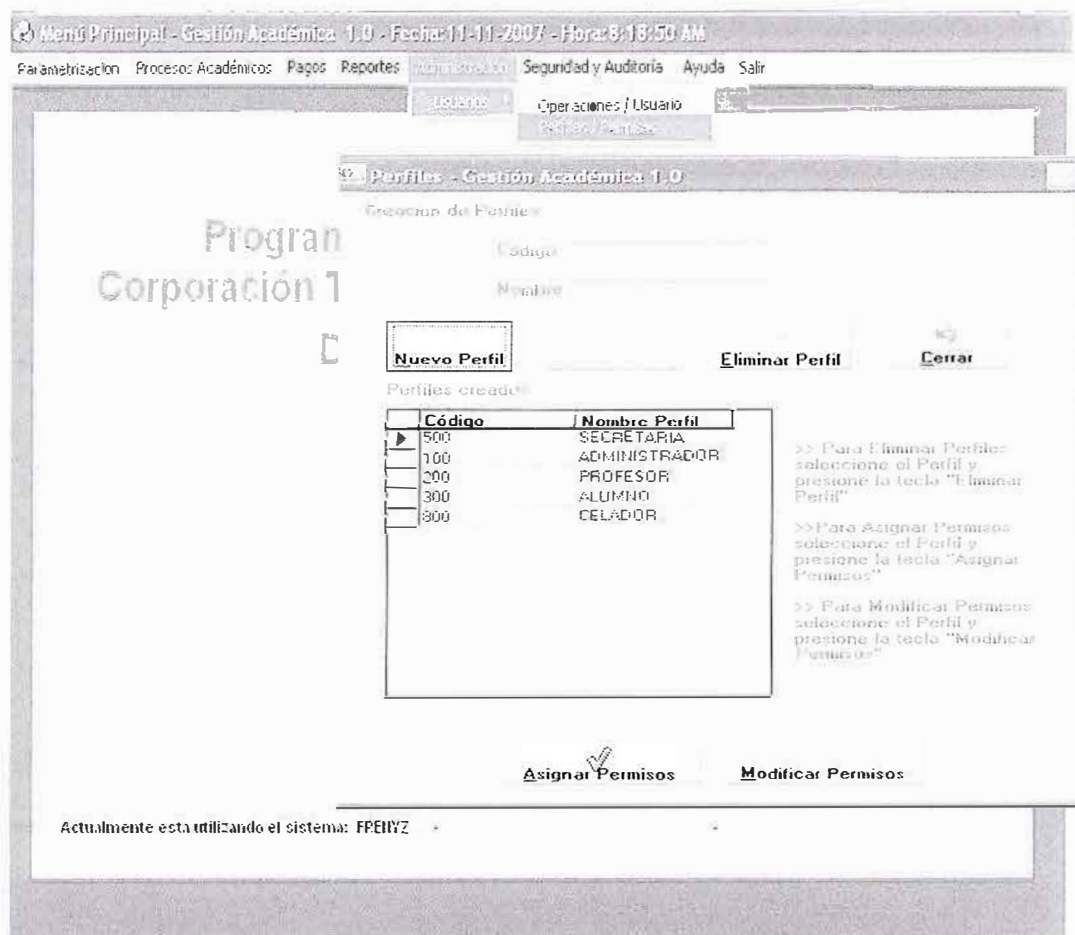
- Para modificar un usuario presionamos el botón  , al presionar este botón se activa  **Asignación Perfil** que también nos permite la reasignación de otro perfil dado el caso que se desee, en caso de que los datos del usuario a modificar no se encuentren visibles en la pantalla , los podemos buscar presionando el botón 

Nota: La búsqueda se realiza por el numero de identificación del usuario

- Para guardar los cambios presionamos 
- Para eliminar un usuario presionamos el botón  , en el caso que no se encuentre en pantalla el usuario que se desea eliminar lo podemos buscar por medio del botón  **Nota :** La búsqueda se realiza por el numero de identificación del usuario
- Para cerrar el modulo de creación de usuario presionamos el botón



- **Perfiles/ Permisos**



En este modulo se realiza la creación de los perfiles con sus permisos de ingreso al sistema

Procesos

- para crear un perfil presionamos el botón



- Luego en el campo **Codigo** digitamos el código del perfil
- en el campo **Nombre** digitamos en nombre del perfil



- para guardar el perfil creado presionamos el botón **Guardar**

Nota: En el momento que presione el botón guardar únicamente se esta guardando los datos digitados del perfil como son: **Nombre y Código**, no se guardara ningún permiso hasta que no se asignen con el botón **Asignar permisos**

- Para asignar los permisos presionamos el botón **Asignar Permisos** este botón muestra una pantalla para la asignación de los permisos que serán asignados al usuario



- En este modulo se encuentran todos los posibles permisos que puede tener un perfil para acceder al sistema, en la parte superior derecha de la pantalla se muestra el nombre y el código seleccionado para la asignación de permisos

- Para la selección damos click en los permisos deseados y presionamos el botón 

- Para salir del modulo de selección presionamos el botón 

- Para modificar un los permisos de un perfil presionamos el

Modificar Permisos

Este nos traerá de nuevo la pantalla de selección de permisos pero con un nuevo botón que debemos presionar  que permite ver que permisos tiene asignado el perfil a modificar, luego de presionarlo , realizamos las modificación de igual forma que en la asignación inicial de los permisos, por ultimo para guardar los cambios presionamos el botón



- Para la eliminación de un perfil primero debemos seleccionar el perfil a eliminar , luego presionamos el botón Eliminar

- Por ultimo para salir del modulo completo presionamos el botón 

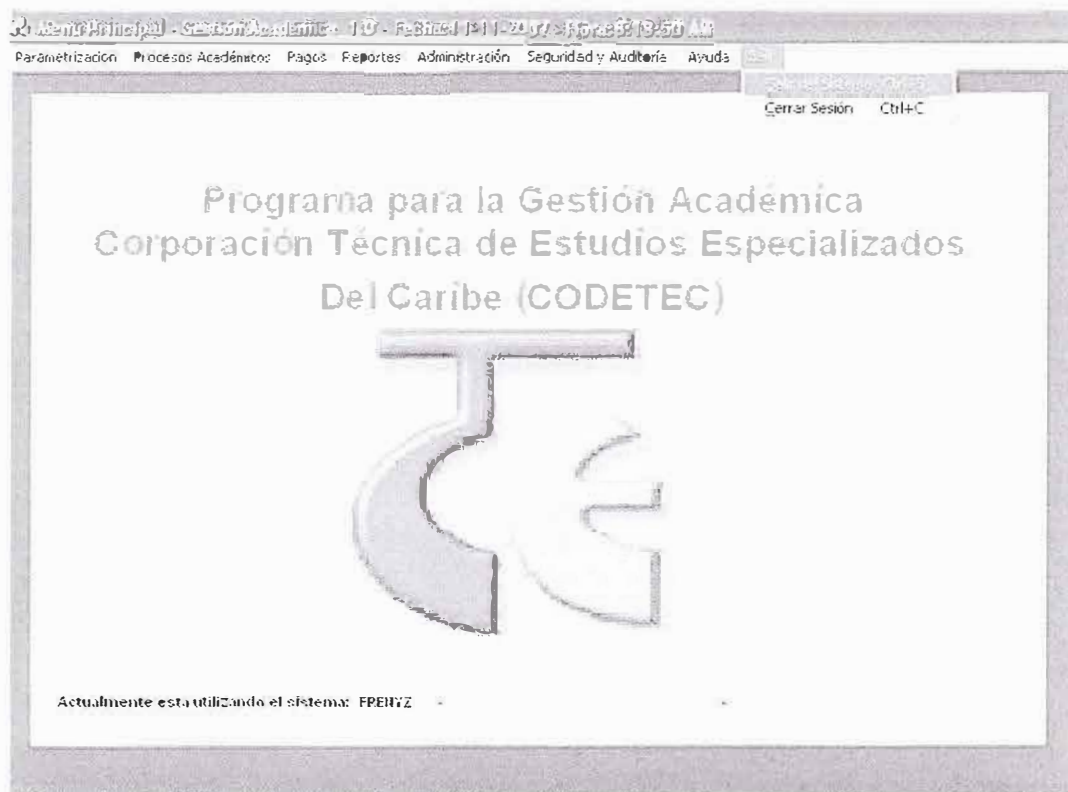


7. AYUDA

- **Acerca de Gestión Académica 1.0**

8. Salir

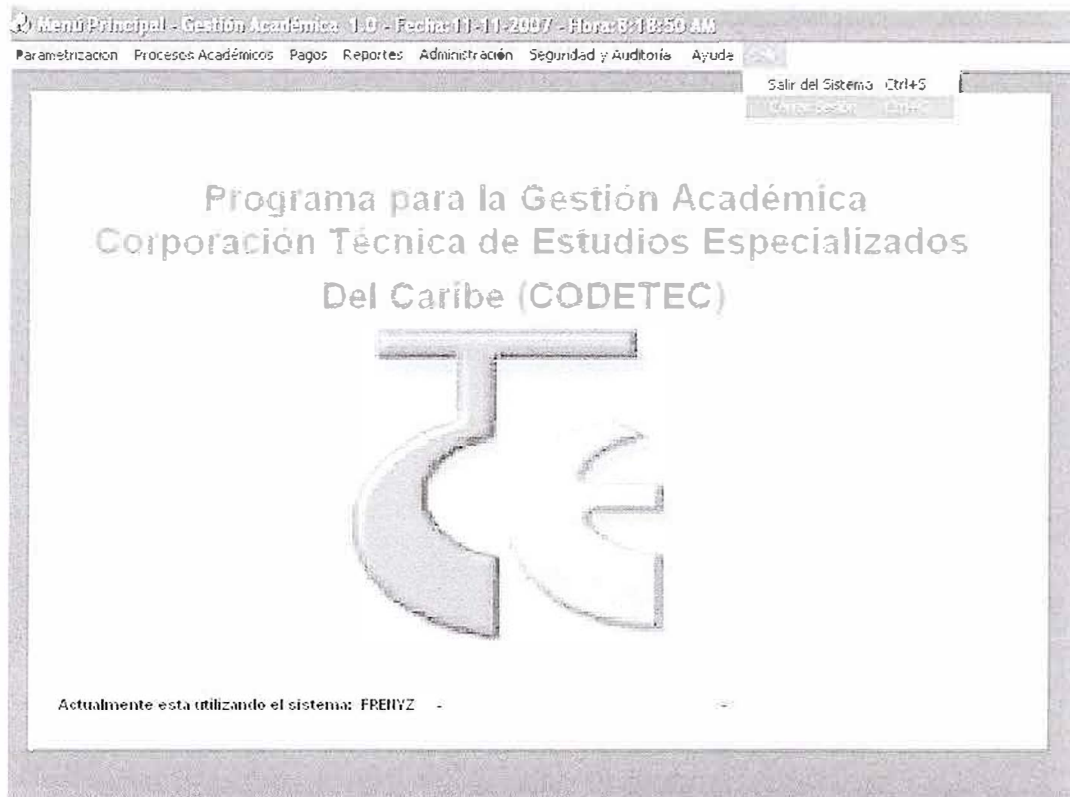
- **Salir del Sistema**



para salir del sistema, debemos ir a la opción Salir del menú principal y presionar Salir del Sistema o también con la combinación de teclas Ctrl+s



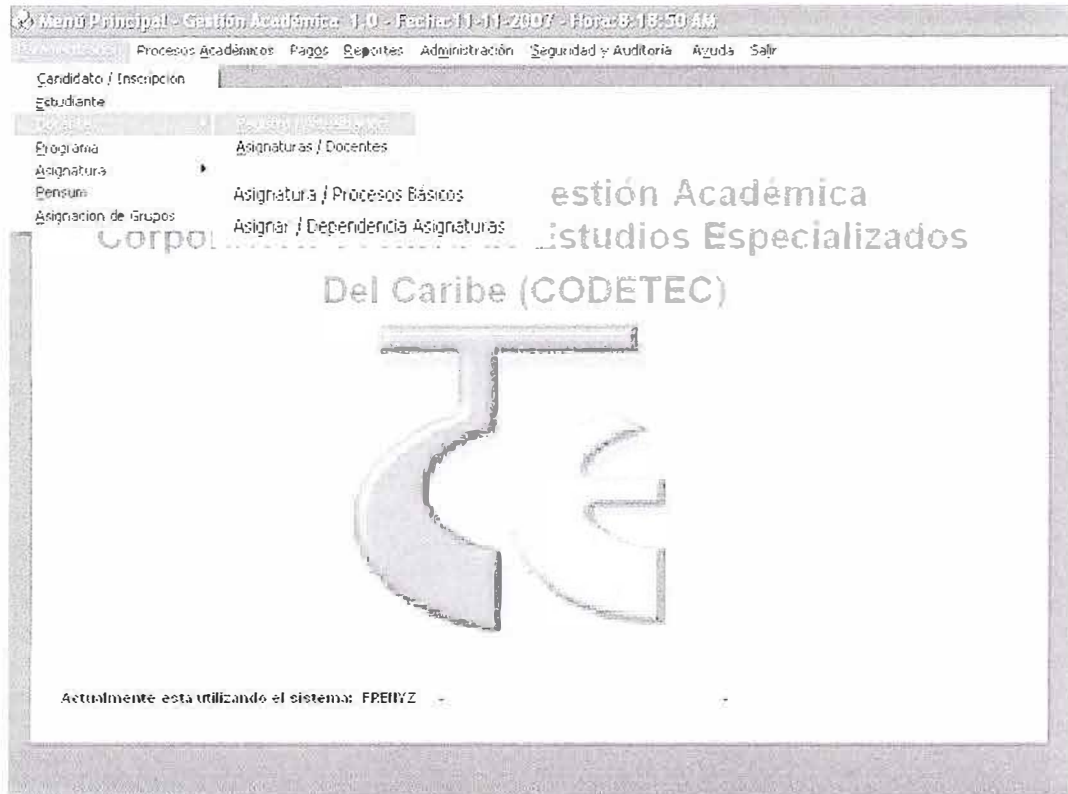
- **Cerrar Sesión**



Para cerrar la sesión, nos vamos a la opción **Cerrar Sesión** o con la combinación de teclas **Ctrl+C**



AYUDAS PARA RECORRER EL MENÚ CON EL TECLADO



Esta parte del manual se realizó con el fin de brindar otra alternativa para acceder al aplicativo y a todas sus opciones por medio de las distintas combinaciones de teclas que nos ofrecen el Sistema Operativo Windows

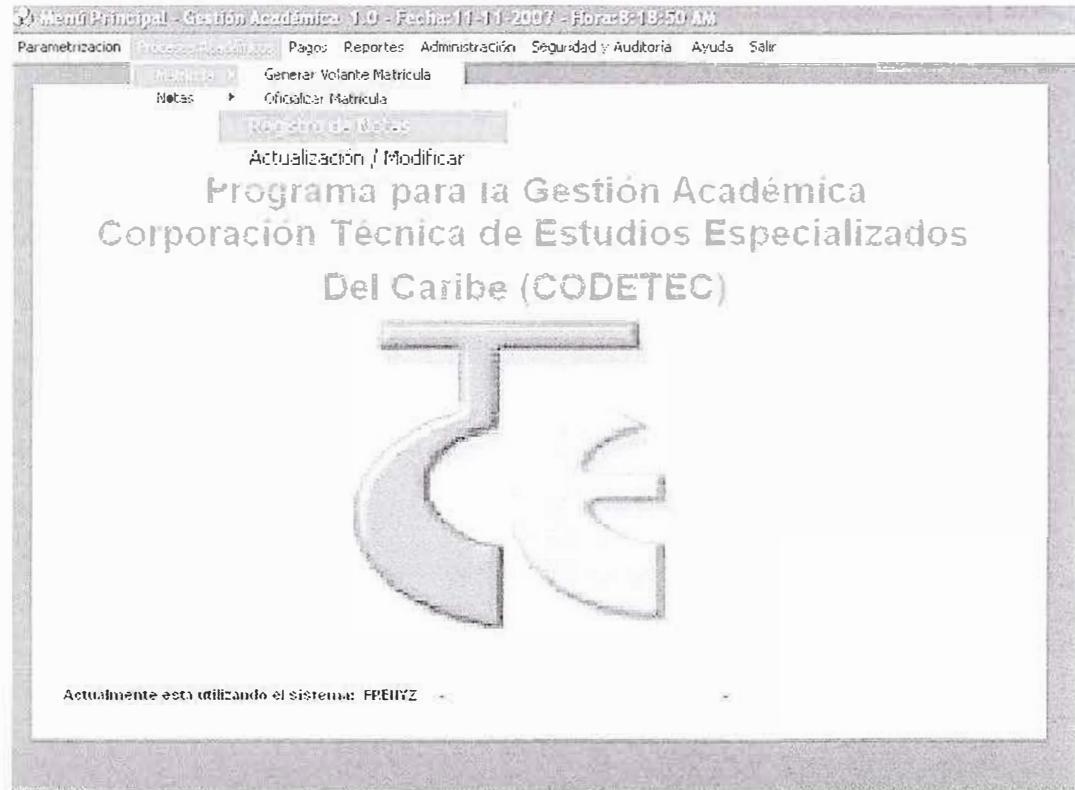
- **PRAMETRIZACION** Ctrl + P
 - Candidatos / Inscripción Ctrl + C
 - Estudiante Ctrl + E
 - Docente Ctrl + D



- Docente / Registro / Actualización Ctrl + R
- Docente / Asignaturas /Docentes Ctrl + A
- Programa Ctrl +P
- Asignatura Ctrl + A
- Asignatura / Procesos / Básicos Ctrl + P
- Asignatura / Asignar / Dependencia Asignaturas Ctrl + A
- Pensum Ctrl + P
- Asignación de Grupos Ctrl + A



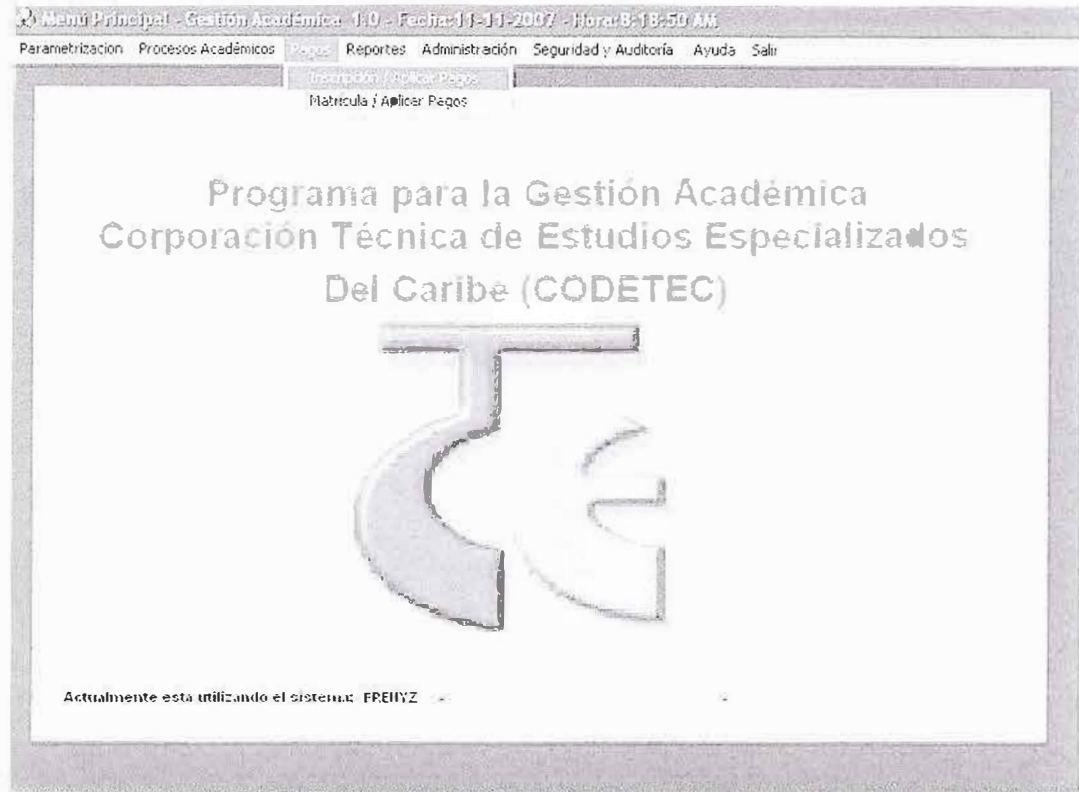
- **PROCESOS ACADEMICOS Ctrl + A**



- **Procesos Académicos / Matrícula Ctrl +M**
- **Procesos Académicos / Matrícula / Generar Volantes Matricula Ctrl +G**
- **Procesos Académicos / Matrícula / Oficializar Matricula Ctrl + O**
- **Notas Ctrl + N**
- **Notas / Registro de Notas Ctrl +R**
- **Notas / Actualización / Modificación Ctrl + A**

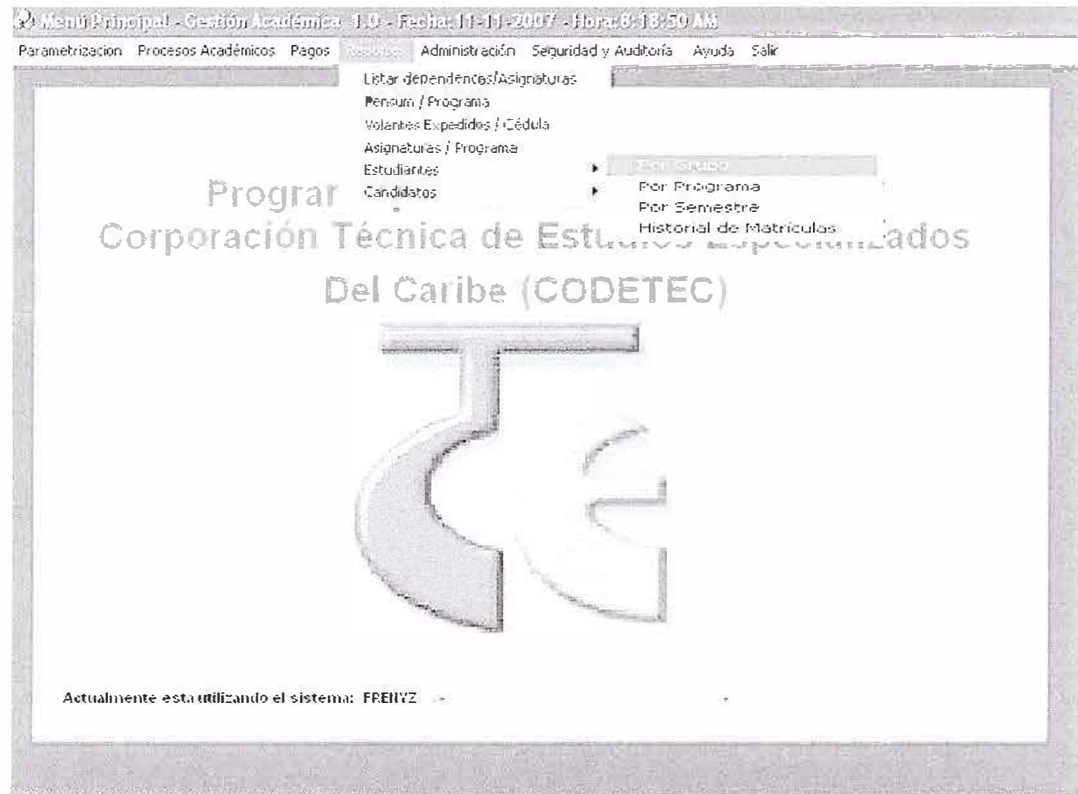


- **PAGOS** Ctrl + O



- Pagos / Inscripción / Aplicar Pagos Ctrl + A
- Pagos / Matricula / Aplicar Pagos Ctrl + A

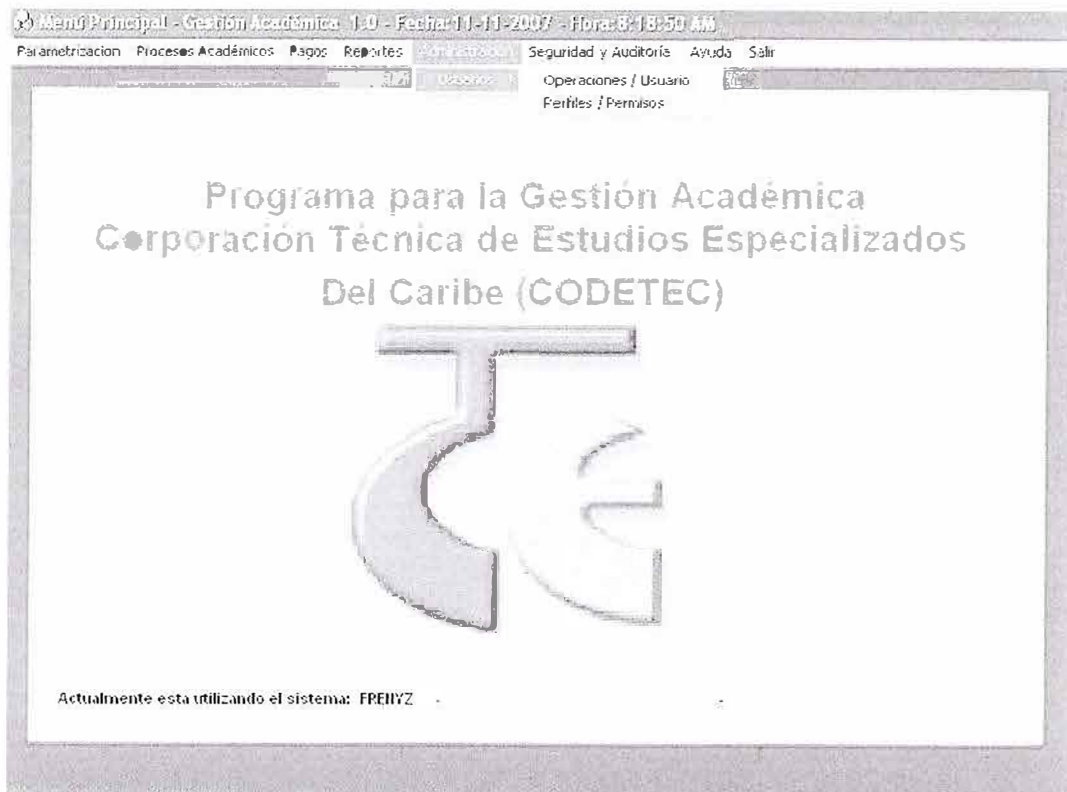
- **REPORTES** Ctrl + R



- Reportes - > Listar dependencias / Asignaturas Ctrl + L
- Reportes - > Pensum / Programa Ctrl + P
- Reportes - > Volantes Expedidos / Cedula Ctrl + V
- Reportes - > Asignaturas / Programas Ctrl + A
- Reportes - > Estudiantes Ctrl + E
- Reportes - > Estudiantes -> Por Grupo Ctrl + G
- Reportes - > Estudiantes -> Por Programa Ctrl + P



- Reportes -> Estudiantes -> Por Semestre Ctrl + S
- Reportes -> Estudiantes -> Historial de Matriculas Ctrl + H
- **ADMINISTRACIÓN** Ctrl + M



- Administración -> Operaciones / Usuario Ctrl + O
- Administración -> Perfiles / Permisos Ctrl + P
- **SEGURIDAD Y AUDITORIA** Ctrl + S
- Seguridad y auditoria -> Log de Transacciones Ctrl + L