

UNIVERSIDAD NACIONAL AUTÓNOMA DE NICARAGUA
UNAN – LEÓN
FACULTAD DE CIENCIAS
DEPARTAMENTO DE COMPUTACIÓN



TRABAJO MONOGRÁFICO PARA OPTAR AL TÍTULO DE
LICENCIATURA EN COMPUTACIÓN

TEMA:

Sistema de control de becas UACH.

AUTORES:

Br. Reyna Isabel Ney Sánchez.
Br. Magaly de los Ángeles Aguilar.
Br. Mario José Lechado.

TUTOR:

Msc. Hermógenes Ruiz.

León, 10/06/10



DEDICATORIA

Dedicamos este trabajo a:

DIOS por ser nuestro padre y protector, quien en ningún momento por difícil que fuera nos abandonó, por darnos día a día toda la fortaleza y paciencia que necesitamos.

Nuestros PADRES, amigos fieles, por todo el sacrificio que han realizado durante todos estos años, con el deseo de vernos crecer profesionalmente.

Nuestros MAESTROS por ser los conductores de nuestros conocimientos profesionales.

Todas aquellas personas que nos han brindado amor, apoyo y nos han acompañado durante el transcurso de nuestras vidas.



AGRADECIMIENTO

Agradecemos a DIOS nuestro padre celestial, por derramar bendiciones sobre nosotros durante toda nuestra vida, por brindarnos la oportunidad de lograr nuestros sueños y regalarnos el don del conocimiento para culminar nuestra carrera.

A nuestros PADRES, los artífices de nuestros días, por darnos su apoyo, amor incondicional y sobre todo por guiarnos en el camino del bien formándonos como buenos profesionales.

A los DOCENTES que nos impartieron clases quienes nos brindaron las bases para nuestra formación profesional, convirtiéndonos en personas responsables y útiles para la sociedad.

A nuestro tutor Msc. Hermógenes Ruiz, un agradecimiento especial por su paciencia y esmero para la elaboración de este trabajo monográfico.

A todas aquellas personas que de manera directa e indirecta colaboraron con sus aportes en la elaboración de este trabajo.



INDICE

Introducción.....	Pág. 1
Antecedentes	Pág. 2
Justificación.....	Pág. 3
Objetivos	Pág. 4
Marco Teórico	Pág. 5
Metodología de la Investigación	Pág. 7
Especificación de Requisitos de Software (ERS)	Pág. 9
Requisitos Específicos	Pág. 12
Diagrama de Flujos de Datos (DFD)	Pág. 23
Diagrama Entidad Relación (E-R)	Pág. 26
Diccionario de datos.....	Pág. 28
Diseño Estructural	Pág. 29
Diseño Arquitectónico.....	Pág. 40
Diseño Procedimental	Pág. 41
Diseño de la Interfaz	Pág. 42
Codificación.....	Pág. 50
Conclusión.....	Pág. 76
Recomendaciones.....	Pág. 77
Bibliografía	Pág. 78



INTRODUCCION

En la ciudad de Chinandega hace quince años nace **La Universidad Autónoma de Chinandega UACH “Presbítero y Dr. Tomás Ruiz”**, en la que se ofrecen ocho carreras en las modalidades: Diurno (Presencial) y Encuentros (Sabatino y Dominical).

A través de los años la población estudiantil ha aumentado pero de igual manera el porcentaje de emigración estudiantil es grande, esto se debe a la crisis económica que actualmente se vive en nuestro país, por lo que la comisión universitaria siendo estos proveedores internos, junto con otras instituciones quienes son proveedores externos, además existen convenios entre la UACH y otras Instituciones, han brindado ayuda a estudiantes de escasos recursos económicos a través de becas, becas que han aumentado con el paso de los años.

Este control de becas se llevaba manualmente, lo que no permitía eficiencia en su manejo, por lo tanto la comisión universitaria solicitó nuestra ayuda para la realización de un sistema que permita la automatización de datos de estudiantes becados. En perspectiva propusimos como tema monográfico “**SISTEMA DE CONTROL DE BECAS UACH**”.

Este trabajo se elaboró de manera minuciosa y sencilla ya que nuestro propósito es que sea de mucha ayuda para el usuario encargado de manipular el sistema y de esta manera llevar un mejor control de los estudiantes becados; información que se lleva en el área de Caja.

Tomamos como fuente de información libros proporcionado por la biblioteca UACH, Internet, apunte de clases recibidas a lo largo de nuestros estudios universitarios e información proporcionada por el tutor.



ANTECEDENTES

A lo largo de la vida universitaria UACH, en el área de caja se lleva el control de cada uno de los estudiantes matriculados, donde se ha visto como problema principal el control de los estudiantes becados ya que al inicio se llevaba manualmente a través de formato de solicitud de becas, luego se llevaba el registro en un libro de Excel lo que no permitía mantener información eficaz.

Debido a este problema se vió la necesidad de solicitar la realización de un sistema de control automatizado que garantizará el ingreso de becas de forma ágil, veraz, y sobre todo el cumplimiento efectivo de los requerimientos de dicho sistema.



JUSTIFICACION

Desde su inicio la UACH ha contado con una diversidad de datos y registros de todos y cada uno de sus estudiantes. Como es de suponer a través del tiempo esta información se ha venido incrementando hasta llegar al punto de no poder controlar completamente los mismos por falta de un sistema de control de becas.

Esto trae como consecuencia el manejo de información poco confiable y desactualizada, lo que conlleva a la necesidad de automatizar estos datos con el objetivo de llevar un mejor control de todas las becas. Desarrollamos un software que se adapta a la realidad permitiendo de esta manera resolver el problema que se presentaba en esta Universidad, que les permite ingresar datos de los estudiantes becados de forma rápida y eficiente, que garantice disponer información ordenada y de forma inmediata para el análisis y evaluación de estos.



OBJETIVOS

General:

Desarrollar un sistema de control de becas de los estudiantes de la UACH de forma automatizada y eficiente.

Específico:

1. Organizar la información de registro de becas que permita generar informes por modalidad, tipo y carrera.
2. Disminuir el tiempo en las operaciones de control de becas.
3. Mejorar la eficiencia en el manejo de la información.
4. Facilitar la generación de reportes estadísticos.
5. Mejorar la disponibilidad de los datos para el usuario responsable de su manejo.
6. Agilizar los servicios prestado a los estudiantes.



MARCO TEORICO

Para el desarrollo del sistema se utilizó algunas técnicas tomadas de varias estrategias o métodos, entre ellas:

- ❖ Método del Ciclo de Vida Clásico, en el que se define un conjunto de actividades que se deben realizar para el desarrollo e implementación del sistema.
- ❖ Método de Desarrollo del Análisis Estructurado, especifica lo que hará el sistema o aplicación. El problema a resolver es dividido en componentes o procesos cada vez más sencillos, repitiendo esta operación hasta obtener suficientes detalles para entender totalmente la parte del sistema que se encuentra en investigación, aquí tomaremos en cuenta algunas técnicas como: Diagrama de flujo de datos (DFD), Diccionario de datos, Diagrama Entidad Relación (DER), etc.

Tomamos en cuenta también algunas técnicas y tareas que forman partes de la metodología general para el desarrollo del sistema, como:

Sistema: es conjunto de componentes que interactúan entre si para lograr un objetivo común.

Metodología: indica el conjunto de fases, etapas, actividades y tareas a seguir para completar las distintas etapas del ciclo de vida de un sistema informático.

Técnicas: proporcionan un conjunto de regla y procedimiento que nos garantiza mayor calidad en las tareas marcadas en la metodología.

Herramientas: suministrarán un soporte automático o semiautomático donde determinadas tareas pueden ser soportadas en técnicas mecanizadas mediante el uso del ordenador como una herramienta más en el uso del sistema, mejorando la calidad y aumentando la productividad.

Procedimientos: junta los métodos y las herramientas y facilita un desarrollo racional del software definiendo la secuencia en que se deben aplicar los métodos, entrega de documento que se requieren y los controles que ayuden a asegurar la calidad.

Herramientas: Al realizar determinadas tareas que son soportadas en técnicas que pueden ser mecanizadas una herramienta fundamental es el ordenador que permite desarrollar un sistema de información mejorando la calidad y aumentando la productividad.

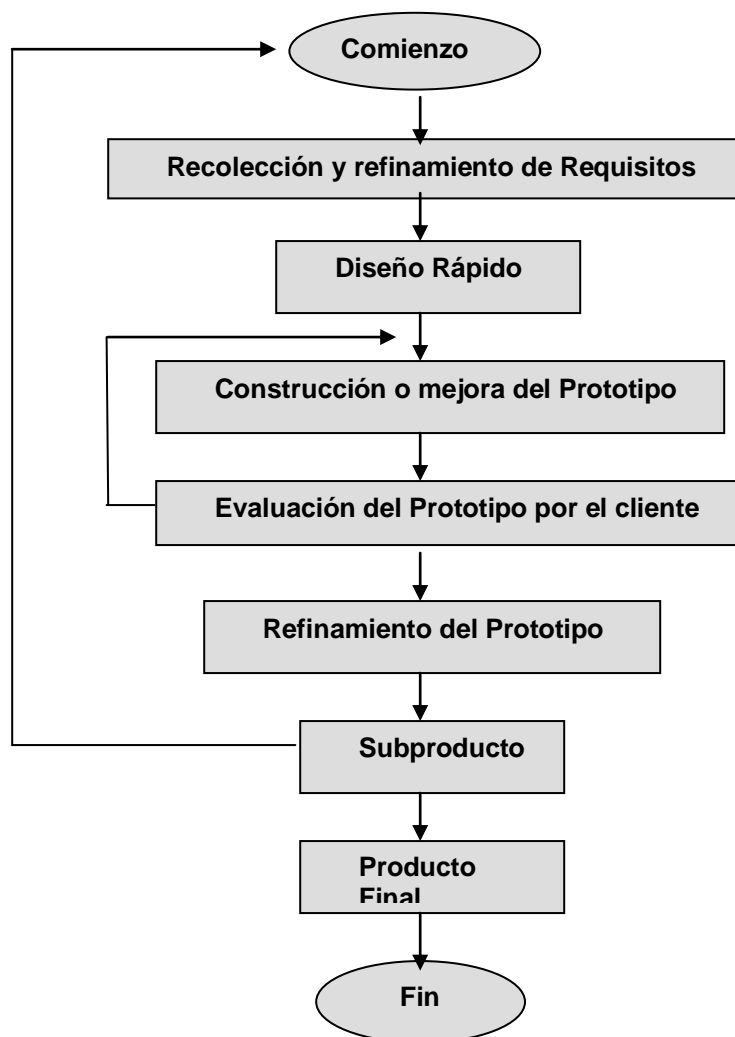
Las herramientas **UML** () es una herramienta que incluye editores gráficos interactivos que facilitan la creación de diagramas y control del software que nos servirá como herramienta para el desarrollo de nuestro sistema.



Diccionario de Requisitos: Es un listado organizado de todos los elementos de datos que son pertenecientes para el sistema, con definiciones precisas y rigurosas que permiten al usuario y al desarrollador del sistema tener la misma comprensión de las entradas, de las salidas, de los componentes del almacenamiento y de los cálculos intermedios. Es también conocido como **Diccionario de Datos**.

Método del Prototipo de Sistema: Este es un proceso que facilita al programador la creación de un modelo de software a construir en el que periódicamente el usuario final revisa el desarrollo y funciones del sistema; éste observa las fallas y brinda pautas para el mejoramiento de la aplicación por lo que el diseñador mejora y prepara los aspectos con problemas para una próxima revisión.

Los pasos a seguir en este modelo son los siguientes





METODOLOGÍA DE LA INVESTIGACIÓN

En el diseño y elaboración del “**Sistema de Control de Becas UACH**”, hicimos uso de algunos métodos de recopilación de datos como: entrevista, revisión periódica del sistema por parte del usuario, planificación, análisis y desarrollo de información, pudiendo extraer lo básicamente necesario para llegar a conclusiones y alternativas para mejorar el rendimiento y funciones del sistema.

Planificación:

En esta etapa es necesario recopilar información para la elaboración del sistema a través de entrevistas no estructuradas, realizadas al encargado del sistema, para cumplir con los requisitos del sistema.

Análisis:

Para llevar a cabo esta etapa haremos uso de algunas herramientas como:

- Diagrama de Flujo de Datos (DFD).
- Diccionario de datos.
- Especificación de Requisitos del Software (ERS).

Diseño:

Esta fase se lleva a cabo en las siguientes etapas:

Diseño de datos: en esta etapa hicimos uso del Diagrama Entidad Relación para el diseño de la base de datos, así como la elaboración de consultas necesaria para la creación de reportes.

Diseño Arquitectónico: será útil para representar el diseño de la interfaz que se presentará al usuario.

Diseño de Interfaz de Usuario: es de gran utilidad para la interacción entre el usuario y la aplicación.

Desarrollo:

Para el desarrollo de este sistema haremos uso del Lenguaje Gestor de Base de Datos: Microsoft SQL Server 2000, Lenguaje de Programación: Microsoft Visual Basic.NET 2005 y como diseñador de Reportes: Cristal Report XI

Prueba y Mantenimiento:

Esta etapa se llevará cabo a través de visitas periódicas que se harán al usuario por dos meses, una vez por semana.



Tomamos en cuenta también algunos objetivos y tareas que forman parte de la metodología general para el desarrollo de sistema como:

Análisis del Sistema: El sistema ha desarrollar tiene como objetivo facilitarle al usuario llevar un mejor control de los estudiantes becados, para la cual este hará uso de una base de datos que proporcionara la información necesaria al encargado de manejar dicho sistema.

Planificación: Este permite obtener una visión general de la Universidad, de sus funciones y necesidades de información, entre algunas de sus técnicas aplicables: Modelo de la Empresa, Modelo Entidad Relación, etc.

Después de hacer un análisis detallado del problema hemos realizado estimaciones necesarias del tiempo que nos tomará desarrollar esta aplicación, el costo, los recursos necesarios como: Hardware sobre el que será montado el sistema, Software que se utilizará para el diseño y la codificación.

Análisis de Requisitos: analiza los proceso que son necesarios para poner en marcha el área estudiada, como se interrelacionan estos procesos y qué datos se relacionan entre si. En esta etapa se utilizó la Especificación de Requisitos del software para establecer el conjunto de requisitos que debe cumplir la aplicación.

A parte de lo antes mencionado haremos uso del Modelo Conceptual de Datos que se encarga de la descripción de los datos y sus relaciones y se hacen con la ayuda de tres conceptos:

- **Diseño del Software:** analiza cómo las actividades del área estudiada son implementadas en procesos, y cómo trabajan éstos. Para lograr el desarrollo de esta etapa haremos uso de la implementación del Modelo Relacional.
- **Codificación:** Aquí se implementan todos los procedimientos a usarse donde puedan utilizarse los lenguajes de programación, generadores de código, etc. Este sistema será implantado en Visual 2005 que es un lenguaje de programación de alto nivel para el desarrollo de aplicación diseñadas para ejecutarse bajo ambiente WINDOWS también haremos uso de SQL Server donde se diseñarán las bases de datos.
- **Prueba y Mantenimiento:** Son los cambios en el software después, de haberse entregado al cliente. Este sistema será probado con datos reales proporcionados por el usuario para detectar fallas en el software.



ESPECIFICACION DE REQUISITOS DE SOFTWARE (ERS)

INTRODUCCION

La especificación de requisitos de software es el establecimiento conciso de un conjunto de requisitos que deben ser procesados para obtener un producto, esperando siempre que sea adecuado el procedimiento mediante el cual se puede determinar si se ha logrado cumplir los requisitos. Tiene como objetivo el establecimiento de un acuerdo entre el usuario y los analistas sobre qué debe hacer el software.

Propósito

Aquí definimos el conjunto de Especificaciones de Requisitos Software que debe cumplir la **aplicación “Sistema de Control de Becas UACH”** de la Universidad Autónoma de Chiandega, agilizando de esta manera el proceso de becas y sobretodo garantizando seguridad y rapidez en su proceso.

Alcance:

El nombre con el que se conocerá esta aplicación será: **SISTEMA DE CONTROL DE BECAS UACH**. Este proyecto realiza las siguientes funciones:

1. Inicio de sesión
2. Datos generales de Solicitudes de becas
3. Registrar y controlar los datos generales del estudiante
4. Controlar solicitudes pendientes, autorizadas y rechazadas
5. Registrar y controlar los datos generales de proveedores de beca.
6. Listas de becas activas
7. Listas de becas canceladas
8. Reportes: Consultas por parámetro:
 - Carrera del estudiante
 - Modalidad del estudiante
 - Proveedor (a quien se le solicita la beca)
 - Institución del proveedor (empresa donde labora)
 - Tipo (puede ser interno, externo o por convenio)
9. Reportes: Becas x Carrera y Tipo
10. Reportes: Becas x Modalidad y Tipo
11. Reportes: Becas pendientes de renovar
12. Reporte global de becas
13. Reporte: Becas x Tipo de proveedor
14. Respaldo de base de datos.
15. Cuenta con ayuda para el usuario.

Listado de funciones que proporciona el sistema:

- ✚ Modificación de datos
- ✚ Eliminar datos
- ✚ Guardar datos
- ✚ Buscar estudiante por parámetros: carnet, nombre, carrera y modalidad
- ✚ Buscar proveedor por parámetros: identificación y nombre
- ✚ Actualizar datos



Definiciones, Acrónimos y Abreviaturas:

Sistema: Programa o conjunto de programas que efectúan la gestión de los procesos básicos de un sistema informático, y permite la ejecución de varias operaciones.

Seguridad: Tiene como función proteger la entrada al sistema.

Contraseña: Es un código de acceso al sistema.

Estudiante: Persona que solicita beca.

Proveedor: Es la persona encargada de suministrar beca puede ser de la universidad (Interno) u otra institución (Externo) o por convenio (Universidad-Otra institución).

Tipo de Beca: Se otorgan becas completas (100%), media beca (50%) y un cuarto de beca (25%) y 10% de beca.

Estado de Beca: Una beca puede estar activa o cancelada.

Tipo de Solicitud: Una solicitud puede ser nueva beca o renovada.

Estado de Solicitud: Puede ser: *Autorizada* es la que ya se aprobó, *Rechazada* es a la que no se le concedió la beca y *Pendiente* a la que no se le ha dado respuesta.

Referencias: Informes obtenidos como resultados de entrevistas realizadas.

Visión General:

Realizamos primero una descripción general del proyecto que se desarrolló para pasar posteriormente a estudiar cada uno de los requisitos específicos individualmente.

Descripción general:

La base de datos se ha instalado en una máquina que funciona como servidor y tiene las siguientes características:

- Procesador Intel Pentium IV
- 448 MB de memoria RAM
- 1.80 Ghz de velocidad de procesamiento

Funciones del sistema:

Nuestra aplicación contiene las siguientes tareas que se llevan a cabo manualmente en el área de Caja de la UACH de forma diaria:

1. Para acceder al sistema aparece una ventana que pide el usuario y la contraseña. (Debe ser llenado obligatoriamente)



2. En el caso que se desee eliminar o ingresar una nueva carrera, un nuevo tipo de proveedor, un nuevo tipo de beca o un nuevo requisito; cuenta con la opción de catálogos.
3. Cuando se solicita una beca permite introducir los datos del estudiante en el ordenador.
4. Existe un formulario que permite registrar los datos de los proveedores de becas.
5. Cuando se desee realizar modificaciones, eliminar o actualizar los datos del estudiante o proveedor es necesario realizar búsqueda ya sea por nombre o identificación de ambos.
6. Lista las becas que se encuentran activas y las que se encuentran canceladas.
7. Lista todos los estudiantes que han solicitado beca y están pendientes.
8. Lista todos los estudiantes que solicitaron becas y fueron rechazadas.
9. El usuario dispone de consultas que le permite saber la cantidad de estudiantes que solicitaron becas por carrera, modalidad, proveedor, institución del proveedor y por tipo.
10. El usuario dispone de los siguientes reportes.
 - Becas x Carrera y Tipo, permite saber quienes son los estudiantes que solicitaron becas por carrera y tipo de beca.
 - Becas x Modalidad y Tipo, permite saber quienes son los estudiantes que solicitaron beca por modalidad y tipo de beca.
 - Becas pendientes de renovar, permite saber quienes son los estudiantes que aún no han renovado su beca.
 - Reporte global de becas, lista todos los estudiantes que solicitaron beca.
 - Becas x Tipo de proveedor, muestra los estudiantes que solicitaron beca por tipo de proveedor ya sea interno, externo o por convenio.
11. Tiene la opción de imprimir el historial de un estudiante en específico.
12. Existe una herramienta que permite realizar respaldo de la base de datos, indicando la fecha, hora y dispositivo de almacenamiento en el que se realizó.

Característica del usuario:

El usuario final de la aplicación serán personas cuya experiencia informática es escasa motivo por el que se ha incluido ayuda en el producto final.

Restricciones generales:

- ✚ No se instala el sistema en máquinas que no cumplan con los requisitos técnicos.
- ✚ No pueden acceder personas que no pertenezcan al personal administrativo de la universidad.



REQUISITOS ESPECÍFICOS

Requisitos funcionales:

+ Inicio de Sesión

Introducción: Esta función es requisito obligatorio, limita el acceso a usuarios que no pertenezcan a la institución.

Entradas:

Por pantalla:

- Usuario:
- Contraseña:

Proceso: El sistema realiza la comprobación de los datos ingresados con los establecidos, permitiendo o no el acceso inmediato al formulario principal.

Salidas: Si el dato ingresado es incorrecto el sistema envía un mensaje indicando que *Presione el botón aceptar e intente de nuevo* y si la contraseña es correcta envía un mensaje indicando que la contraseña es correcta.

+ Datos generales de Solicitudes de Becas

Introducción: Este proceso realiza la captura de datos del estudiante que está solicitando beca por primera vez o bien para el que esta renovándola.

Entradas

Por pantalla datos para presentar:

Recepción de Solicitudes de Becas

- Número
- Fecha de Solicitud
- Año Lectivo
- Año Académico



Datos generales del estudiante:

- Carnet
- Nombre completo
- Modalidad
- Carrera
- Solicitado a:
- Proveedor:
- Observación
- Solicitud de:
- Beca Solicitada

Requisitos presentados:

- Certificado Académico
- Carta de Solicitud
- Deportes
- Chica UACH
- Crear solicitud
- Cancelar

Proceso: Este formulario inicia presentando las opciones que permiten capturar los datos de la solicitud de beca, primero se debe verificar si el estudiante está registrado en el sistema, dando clic en el botón que mostrará un formulario **Buscador de Registro**, si el estudiante se encuentra en el sistema sus datos son transferidos al formulario *Recepción de Solicitudes de Becas*, de lo contrario se procede a ingresar los datos de la nueva solicitud de beca; el sistema automáticamente genera el número de la solicitud que se está ingresando en ese momento y son almacenados en la base de datos.

Datos necesarios a introducir:

- Número : Se genera automáticamente
- Fecha : Dato obligatorio
- Año Lectivo : Dato obligatorio
- Año Académico : Dato obligatorio



Sistema de control de Becas UACH



- Carnet del Estudiante : Dato obligatorio
- Nombre Completo : Dato obligatorio
- Modalidad : Dato obligatorio
- Carrera : Dato obligatorio
- Solicitado a : Dato obligatorio
- Observación : Dato no obligatorio
- Requisitos Presentados : Dato obligatorio
- Solicitud de : Dato obligatorio
- Beca Solicitada : Dato obligatorio
- Crear Solicitud : Dato obligatorio
- Cancelar

Salidas: Cuando los datos mencionados anteriormente se han ingresado se envía un mensaje indicando que los datos han sido guardados, en caso de que se omita un dato envía un mensaje indicado el dato que hace falta para ser almacenado.

Registrar y controlar los datos generales del estudiante

Introducción: Este proceso permite buscar a un estudiante presentando los datos generales del mismo y también permite el ingreso de un nuevo registro, logrando desde este formulario modificar datos o eliminar registro.

Entradas

Por pantalla se muestran los siguientes datos:

- Carnet
- Primer nombre
- Segundo nombre
- Primer apellido
- Segundo apellido
- Fecha de Nacimiento
- Edad
- Sexo



Sistema de control de Becas UACH



- Teléfono
- Dirección domiciliar
- Modalidad
- Carrera
- Proveedor de la beca
- Tipo de beca
- Año académico
- Fecha otorgada

Proceso: Este formulario inicia presentando los datos generales del primer estudiante que se ingresó al sistema, permitiendo la búsqueda de un determinado registro a través de la barra de desplazamiento o desde el menú Editar; en el caso que el estudiante no se encuentre en la base de datos, desde este formulario se pueden modificar o eliminar registros.

Los datos a introducir son:

- Carnet : Dato obligatorio
- Primer nombre : Dato Obligatorio
- Segundo nombre : Dato opcional
- Primer apellido : Dato obligatorio
- Segundo apellido : Dato opcional
- Fecha de Nacimiento : Dato obligatorio
- Edad : Generado por el sistema
- Sexo : Dato obligatorio
- Teléfono : Dato opcional
- Dirección domiciliar : Dato obligatorio
- Modalidad : Dato obligatorio
- Carrera : Dato obligatorio
- Proveedor de la beca : Dato obligatorio
- Tipo de beca : Dato obligatorio
- Año académico : Dato obligatorio
- Fecha otorgada : Generada por el sistema



Salida: Con todos los datos capturados se tendrá un registro completo de las solicitudes para llevar un mejor control de éstos en la base de datos del sistema.

+ Solicitudes Pendientes, Rechazadas y Autorizadas

Introducción

Solicitudes Pendientes muestra listado de solicitudes que aún se encuentran pendientes por Aprobar o Rechazar. *Solicitudes Rechazadas* muestra listado de solicitudes que han sido denegadas por la Universidad. *Solicitudes Autorizadas* muestra listado de solicitudes que fueron aprobadas por la universidad.

Entrada: Por pantalla se muestran los siguientes datos:

Lista de Solicitudes Pendientes de procesar/Rechazadas/Autorizadas

Parámetros de búsqueda

- Carnet
- Año lectivo
- Año académico
- Carrera
- Modalidad
- Solicitado a:
- Tipo de beca solicitada

Proceso: Este formulario presenta listado de los diferentes estados en que puede estar una beca solicitada con sus parámetros de búsqueda que nos ayudan a filtrar información. Debemos seleccionar los parámetros de búsqueda que deseamos, y luego dar click derecho en la parte superior del formulario y seleccionar Buscar. Podemos imprimir, para ello debemos dar click derecho en la parte superior del formulario y seleccionar la opción **imprimir listado**.

Salidas: Todos los datos filtrados mediante la búsqueda para un estudio bien detallado.

+ Datos generales del Proveedor de beca

Introducción: Este proceso realiza la captura de datos generales de los diferentes proveedores de becas, dado que puede ser interno, externo o por convenio.

Entrada: Por pantalla se muestran los siguientes datos:

Proveedores de la beca

- Identificación
- Primer nombre
- Segundo nombre
- Primer apellido



- Segundo apellido
- Institución
- Cargo
- Teléfono
- Tipo
- Dirección domiciliar

Proceso: En este formulario podemos agregar un nuevo proveedor y hacerle modificaciones desde la barra de desplazamiento o desde el menú editar, esto permitirá la administración de los datos del proveedor dándolo de alta o de baja.

Se presentará al usuario en la pantalla de introducción de datos a ingresar:

Proveedores de becas:

- Identificación : Dato obligatorio
- Primer nombre : Dato obligatorio
- Segundo nombre : Dato opcional
- Primer apellido : Dato obligatorio
- Segundo apellido : Dato opcional
- Institución : Dato obligatorio
- Cargo : Dato obligatorio
- Teléfono : Dato opcional
- Tipo : Dato obligatorio
- Dirección domiciliar : Dato obligatorio

Salidas: Todos los datos referentes a los proveedores de becas se almacenarán en el sistema para uso posterior.

Listas de becas Activas

Introducción: En este proceso podemos ver que estudiantes han obtenido becas en todo el año escolar

Entrada: Por pantalla se muestran los siguientes datos:

Listado de becas Activas

Parámetros de búsqueda

- Carnet
- Año lectivo
- Año académico
- Carrera
- Modalidad
- Proveedor de beca
- Tipo de beca



Proceso: Este proceso muestra a los estudiantes que se le ha otorgado beca; debemos seleccionar los parámetros de búsqueda que deseamos, y luego dar click derecho en la parte superior del formulario y seleccionar Buscar. Una vez obtenido nuestra lista podemos imprimir, para ello dar click derecho en la parte superior del formulario y seleccionamos la opción **imprimir listado**. Podemos ver el historial del estudiante dando click sobre la barra estado. Desde ahí se puede modificar, renovar o cancelar la beca. Se puede imprimir de manera global o individual a los estudiantes con su historial.

Salida: Se visualizará el listado de becas que se encuentran activas durante el año lectivo seleccionado por parámetros de búsqueda para su uso posterior.

Listas de becas Canceladas

Introducción: Este formulario muestra una lista de los estudiantes que se le ha cancelado la beca.

Entrada: Por pantalla se muestran los siguientes datos:

Listado de becas canceladas

Parámetros de búsqueda

- Carnet
- Año lectivo
- Año académico
- Carrera
- Modalidad
- Proveedor de beca
- Tipo de beca

Proceso: Este formulario muestra la lista de estudiantes que se les canceló la beca con su historial, motivo por el que fue cancelada. Debemos seleccionar los parámetros de búsqueda que deseamos, y luego dar click derecho en la parte superior del formulario y seleccionar Buscar. Una vez que tenemos nuestro informe podemos imprimir el Listado, para ello debemos dar click derecho en la parte superior del formulario y seleccionamos la opción **imprimir listado**.

Salida: Se visualizará el listado de becas que se encuentran canceladas, seleccionado por parámetros de búsqueda.

Respaldo de Base de Datos

Introducción: Esta función le permite al usuario realizar copia de seguridad de la base de datos del sistema.

Proceso: Existe un menú de herramientas donde el usuario deberá escoger la opción Respaldo BD, la que realizará una copia de la base de datos, solo tiene que especificar en que unidad o dispositivo de almacenamiento desea que se realice la copia, la que indica la fecha y la hora que se ha realizado.



Reportes: Consultas

Introducción: Este proceso permite al usuario realizar consultas por parámetros.

Entradas: Se presentará al usuario por pantalla las consultas los siguientes datos:

Cantidad de becas por parámetros

- Cantidad por:
- Año lectivo

Proceso: En este formulario se pueden realizar consultas por los siguientes parámetros: Carrera, Modalidad, Proveedor, Institución del proveedor y Tipo; donde el usuario podrá seleccionar el parámetro a consultar y el año lectivo; que mostrará listado del parámetro seleccionado, luego damos clic en el registro que deseamos que se detalle, damos clic al inicio de la celda y seleccionamos imprimir.

Salida: Con estos datos el sistema va a generar la información almacenada en la base de datos.

Reportes: Becas x Carrera y Tipo

Introducción: Este formulario permite buscar becas mediante filtros por carrera y Tipo

Entrada: Se presentará al usuario por pantalla los siguientes datos:

Filtros del reporte

- Año lectivo
- Tipo
- Carrera
- Botón Aceptar
- Botón Cancelar

Proceso: Este proceso genera reportes de becas por Año lectivo, Tipo y Carrera; una vez seleccionados los parámetros damos clic en aceptar y se nos mostrará el reporte listo para ser impreso.

Salida: Con estos datos el sistema va a generar la información almacenada en la base de datos.

Reportes: Becas x Modalidad y Tipo

Introducción: Este formulario permite filtrar becas por Modalidad y Tipo

Entrada: Se presentará al usuario por pantalla los siguientes datos:



Filtros del reporte

- Año lectivo
- Tipo
- Modalidad
- Botón Aceptar
- Botón Cancelar

Proceso: Este proceso genera reportes de becas por Año lectivo, Tipo y Modalidad; una vez seleccionados los parámetros damos clic en aceptar y se nos mostrará el reporte listo para ser impreso.

Salida: Con estos datos el sistema va a generar la información almacenada en la base de datos.

+ Reportes: Becas pendientes de renovar

Introducción: Este formulario permite filtrar becas que aún no se ha hecho solicitud de renovación.

Entrada:

Se presentará al usuario por pantalla los siguientes datos:

Filtros del reporte

- Año Lectivo
- Tipo
- Carrera
- Botón Aceptar
- Botón Cancelar

Proceso: Este proceso genera reportes de becas por parámetros Año Lectivo, Tipo y Carrera; una vez seleccionados los parámetros damos clic en aceptar y se nos mostrará el reporte listo para ser impreso.

Salida: Con estos datos el sistema va a generar la información almacenada en la base de datos.

+ Reporte global de becas

Introducción: Este formulario genera reporte global de becas por Año lectivo.

Entrada: Se presentará al usuario por pantalla los siguientes datos:

Filtros del reporte

- Año lectivo
- Botón Aceptar
- Botón Cancelar



Sistema de control de Becas UACH



Proceso: Este proceso genera reportes global de becas por Año lectivo, una vez seleccionado el Año lectivo damos clic en aceptar y se nos mostrará el reporte listo para ser impreso.

Salida: Con estos datos el sistema va a generar la información almacenada en la base de datos.

Reporte: Becas x Tipo de proveedor

Introducción: Este formulario genera reporte de becas por tipo de proveedor que pueden ser internos de la institución, externos de la institución o por convenio (Institución y otra Institución).

Entrada: Se presentará al usuario por pantalla los siguientes datos:

Filtros del reporte

- Año lectivo
- Tipo
- Botón Aceptar
- Botón Cancelar

Proceso: Este proceso genera reportes de becas por Año lectivo y Tipo de proveedor, una vez seleccionado los parámetros damos clic en aceptar y se nos mostrará el reporte listo para ser impreso.

Salida: Con estos datos el sistema va a generar la información almacenada en la base de datos.



Interfaces externa:

Interfaces de usuario: La captura de la información en general de los datos del estudiante y proveedores de becas se realiza de forma interactiva por pantalla.

Interfaces de hardware: El software está diseñado en arquitectura cliente-servidor y se instaló en una máquina que actúa como servidor.

Interfaces de software: El proceso interactúa con la base de datos de ficheros maestros ya mencionados.

Interfaces de comunicaciones: No existe ninguna interfaz de comunicación en la aplicación.

Requisitos de funcionamiento:

Requisitos estáticos: No existen restricciones sobre la cantidad de usuario que trabajen el sistema.

Restricciones de diseño: La pantalla principal de la aplicación lleva el logotipo y el nombre de la institución.

Atributos:

Seguridad: La aplicación deberá estar protegida mediante una contraseña. Esto significa que habrá una persona encargada de operar en el sistema.

Mantenimiento: Cualquier modificación que afecte los requisitos mencionados en este documento deberá ser reflejado en el mismo, así como la documentación obtenida en la fase de análisis, diseño y programación.

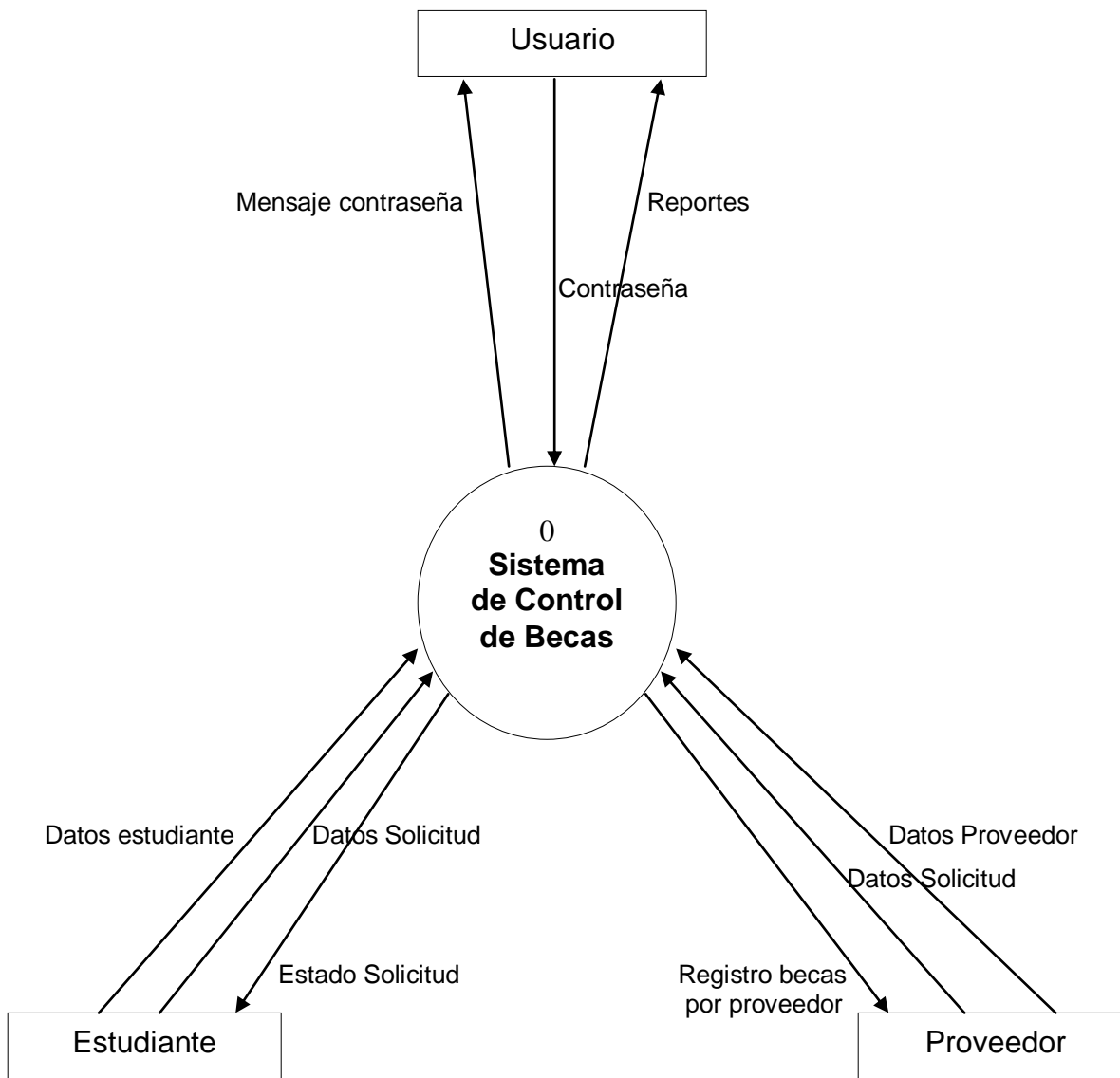
Otros requisitos:

Base de datos: La información será almacenada por medio de una base de datos correspondiente a las actividades realizadas en el área de caja UACH. Esta base de datos fue realizada en SQL Server 2000.

Operaciones: Todas las operaciones se llevarán a cabo de acuerdo a las restricciones de seguridad, donde la primera operación fue crear la clave de acceso para entrar a la base de datos. Realizamos menús contextuales para la introducción de la información y la visualización de los reportes.

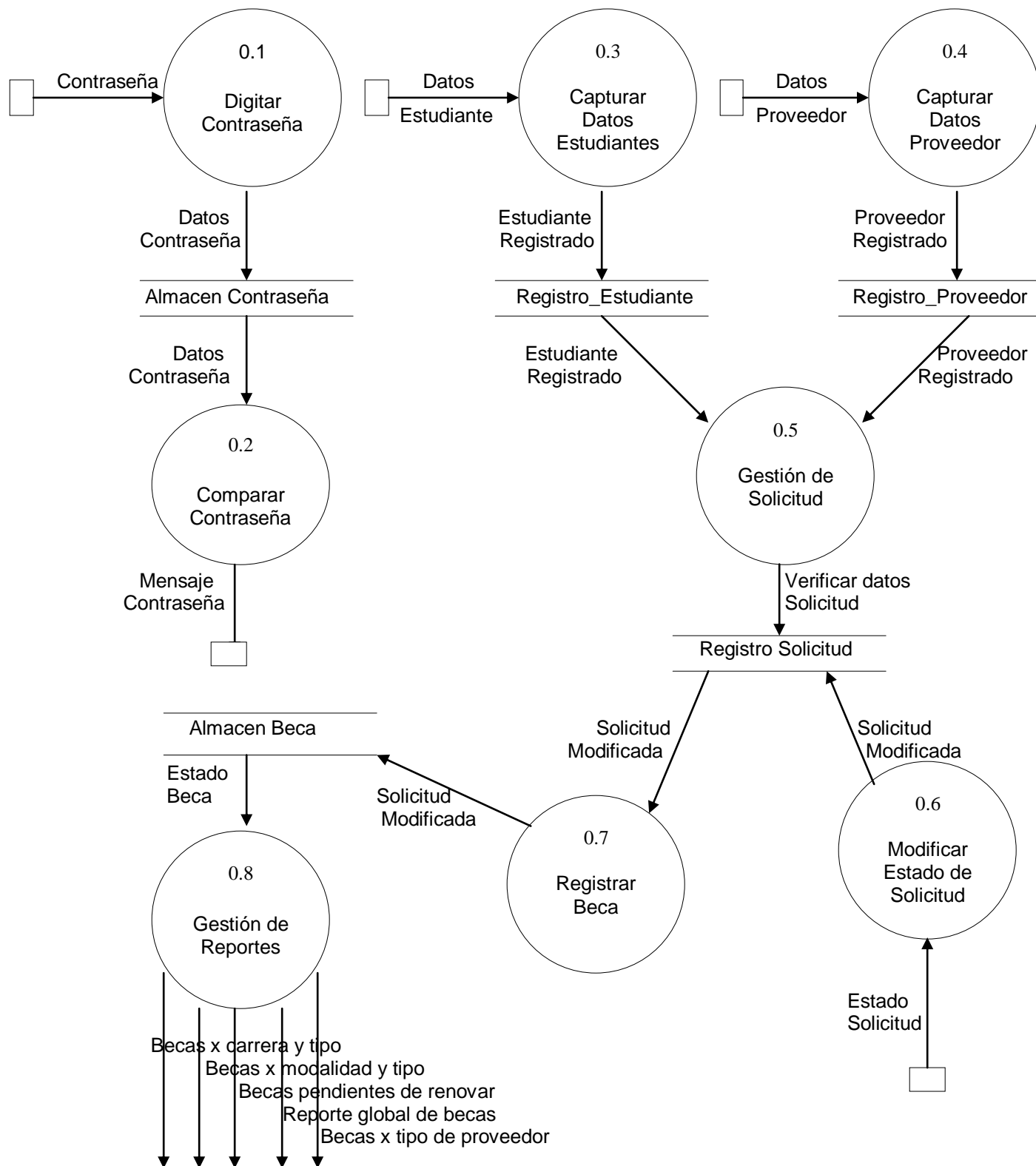


DIAGRAMA DE FLUJO DE DATOS (DFD)





Nivel 1





Nivel 2

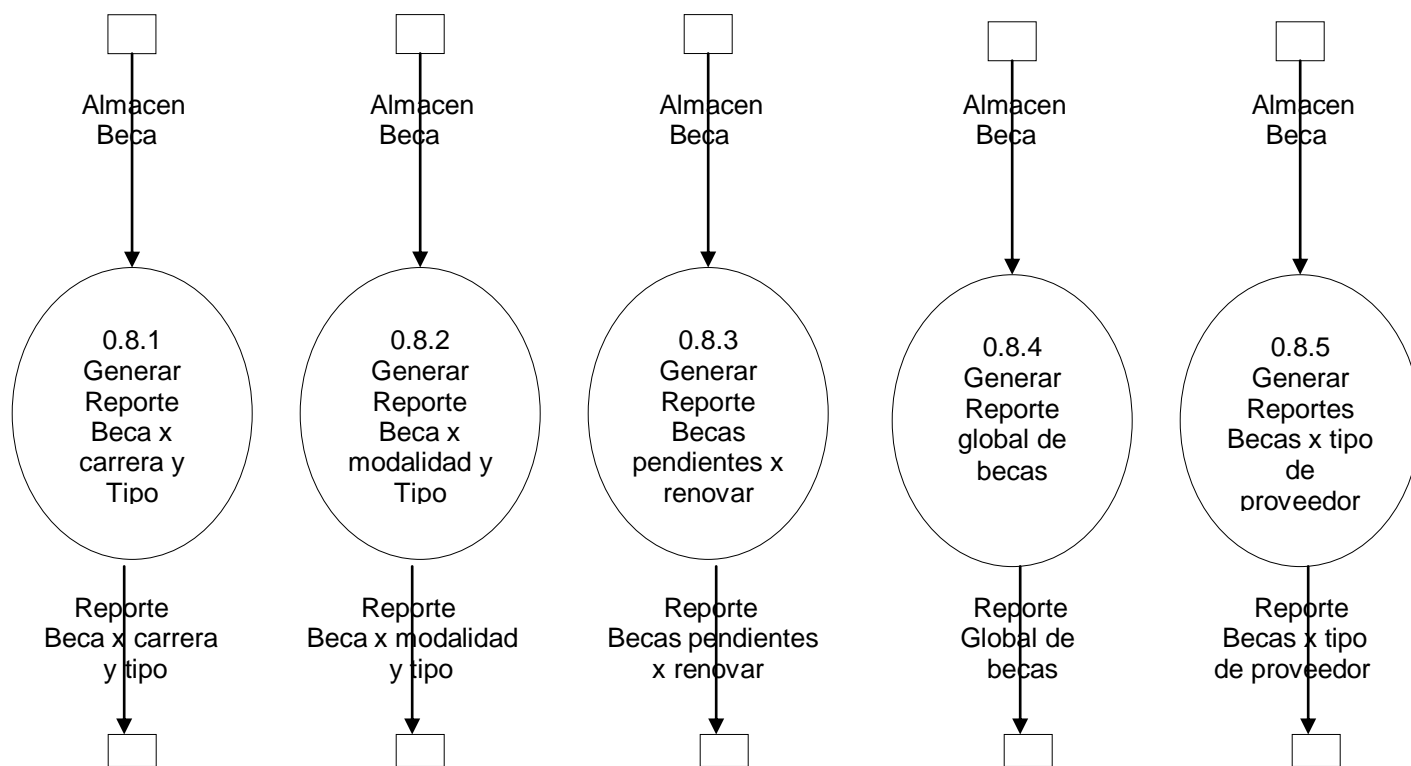
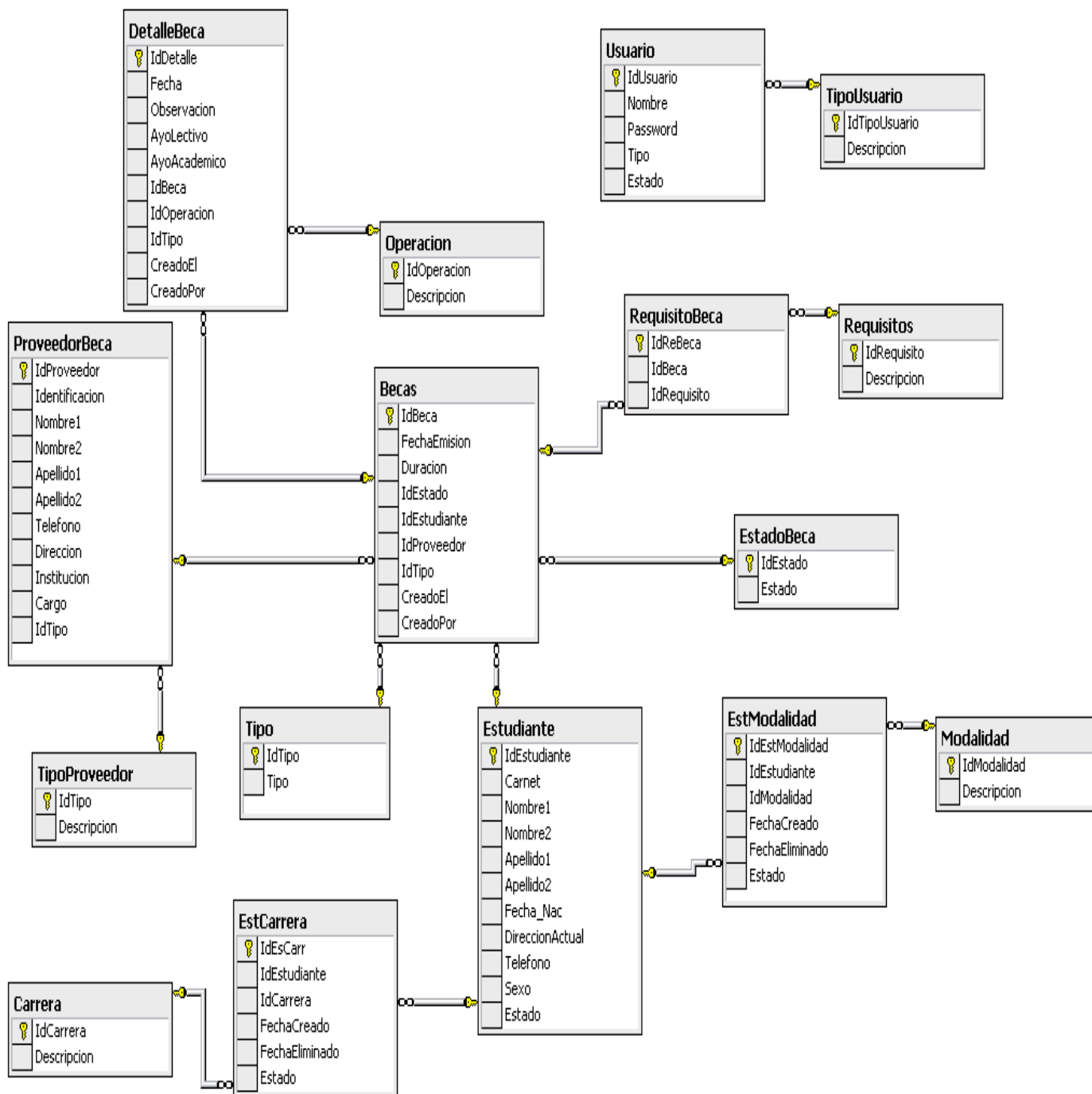


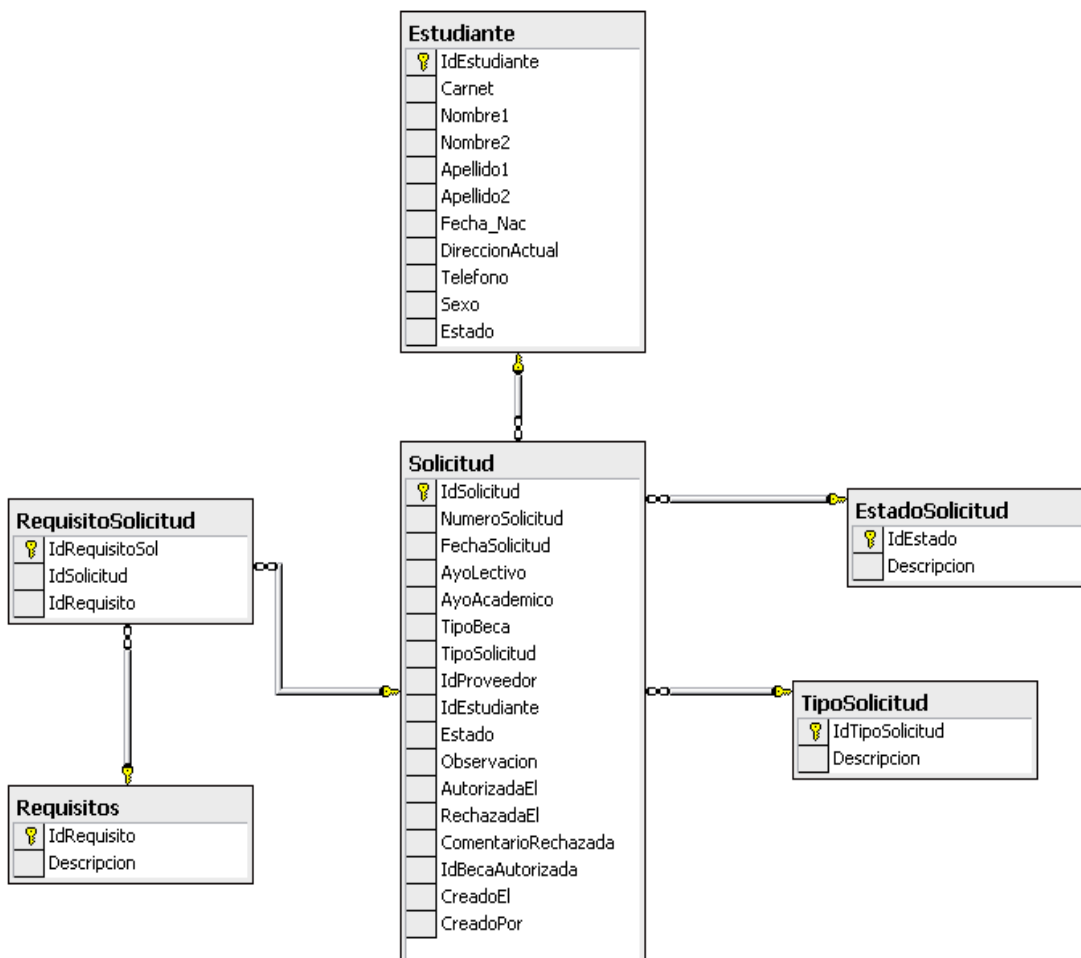


DIAGRAMA ENTIDAD RELACION (E-R)





Sistema de control de Becas UACH





DICCIONARIO DE DATOS

- **Tabla Beca:** IdBeca + FechaEmision + Duracion + IdEstado + Estudiante + IdProveedor + IdTipo + CreadoEl + CreadoPor
- **Tabla Carrera:** Idcarrera + Descripcion

- **Tabla DetalleBeca:** IdDetalle + Fecha + Observacion + AyoLectivo + AyoAcademico + IdBeca + IdOperacion + IdTipo + CreadoEl + CreadoPor

- **Tabla EstadoBeca:** IdEstado + Estado
- **Tabla EstadoSolicitud:** IdEstado + Descripcion
- **Tabla EstCarrera:** IdEsCarr + IdEstudiante + IdCarrera + FechaCreado + FechaEliminado + Estado

- **Tabla EstModalidad:** IdEstModalidad + IdEstudiante + IdModalidad + FechaCreado + FechaEliminado + Estado

- **Tabla Estudiante:** IdEstudiante + Carnet + Nombre1 + Nombre2 + Apellido1 + Apellido2 + Fecha_Nac + DireccionActual + Telefono + Sexo + Estado

- **Tabla Modalidad:** IdModalidad + Descripción
- **Tabla Operacion:** IdOperacion + Descripcion
- **Tabla ProveedorBeca:** IdProveedor + Identificacion + Nombre1 + Nombre2 + Apellido1 + Apellido2 + Telefono + Direccion + Institucion + Cargo + IdTipo

- **Tabla RequisitoBeca:** IdReBeca + IdBeca + IdRequisito
- **Tabla Requisito:** IdRequisito + Descripcion

- **Tabla RequisitoSolicitud:** IdRequisitoSol + IdSolicitud + IdRequisito
- **Tabla Solicitud:** IdSolicitud + NumeroSolicitud + FechaSolicitud + AyoLectivo + AyoAcademico + TipoBeca + TipoSolicitud + IdProveedor + IdEstudiante + Estado + Observacion + AutorizadaEl + RechazadaEl + ComentarioRech + IdBecaAutorizada + CreadoEl + CreadoPor

- **Tabla Tipo:** IdTipo + Tipo
- **Tabla Proveedor:** IdProveedor + Descripcion

- **Tabla TipoSolicitud:** IdTipoSolicitud + Descripcion
- **Tabla TipoUsuario:** IdTipoUsuario + Descripcion

- **Tabla Usuario:** IdUsuario + Nombre + Password + Tipo + Estado



DISEÑO ESTRUCTURAL

Tabla Becas

Propiedades de la tabla: Becas

General

Nombre: Becas

Propietario: dbo

Fecha de creación: 25/11/2008 13:53:12

Grupo de archivos: PRIMARY

Filas: 11

Columnas:

Cla...	Id.	Nombre	Tipo de d...	Tama...	Valor...	Predete
	<input checked="" type="checkbox"/>	IdBeca	int	4		<input type="checkbox"/>
		FechaEmision	datetime	8	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
		Duracion	int	4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
		IdEstado	int	4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
		IdEstudiante	int	4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
		IdProveedor	int	4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
		IdTipo	int	4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Aceptar Cancelar Aplicar Ayuda

Tabla DetalleBeca

Propiedades de la tabla: DetalleBeca

General

Nombre: DetalleBeca

Propietario: dbo

Fecha de creación: 25/11/2008 13:53:12

Grupo de archivos: PRIMARY

Filas: 16

Columnas:

Cla...	Id.	Nombre	Tipo de d...	Tama...	Valor...	Predete
	<input checked="" type="checkbox"/>	IdDetalle	int	4		<input type="checkbox"/>
		Fecha	datetime	8	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
		Observacion	nvarchar	250	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
		AyoLectivo	int	4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
		AyoAcademico	int	4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
		IdBeca	int	4		<input type="checkbox"/>
		IdOperacion	int	4		<input type="checkbox"/>

Aceptar Cancelar Aplicar Ayuda



Tabla EstadoBeca

Propiedades de la tabla: EstadoBeca

General

Nombre: EstadoBeca Permisos...

Propietario: dbo
Fecha de creación: 25/11/2008 13:53:11
Grupo de archivos: PRIMARY
Filas: 2

Columnas:

Cla...	Id.	Nombre	Tipo de d...	Tama...	Valor...	Predetermi
	<input checked="" type="checkbox"/>	IdEstado	int	4	<input type="checkbox"/>	
		Estado	nvarchar	15	<input checked="" type="checkbox"/>	

Aceptar Cancelar Aplicar Ayuda

Tabla Estudiante

Propiedades de la tabla: Estudiante

General

Nombre: Estudiante Permisos...

Propietario: dbo
Fecha de creación: 25/11/2008 13:53:12
Grupo de archivos: PRIMARY
Filas: 28

Columnas:

Cla...	Id.	Nombre	Tipo de d...	Tama...	Valor...	Predetermi
	<input checked="" type="checkbox"/>	IdEstudiante	int	4	<input type="checkbox"/>	
		Carnet	nvarchar	15	<input checked="" type="checkbox"/>	
		Nombre1	nvarchar	20	<input checked="" type="checkbox"/>	
		Nombre2	nvarchar	20	<input checked="" type="checkbox"/>	
		Apellido1	nvarchar	20	<input checked="" type="checkbox"/>	
		Apellido2	nvarchar	20	<input checked="" type="checkbox"/>	
		Fecha_Nac	datetime	8	<input checked="" type="checkbox"/>	

Aceptar Cancelar Aplicar Ayuda



Tabla Carrera

Propiedades de la tabla: Carrera

General

Nombre: Carrera Permisos...

Propietario: dbo
Fecha de creación: 25/11/2008 13:53:11
Grupo de archivos: PRIMARY
Filas: 9

Columnas:

Cla...	Id.	Nombre	Tipo de d...	Tama...	Valor...	Predetermi
	<input checked="" type="checkbox"/>	IdCarrera	int	4	<input type="checkbox"/>	
		Descripcion	nvarchar	50	<input checked="" type="checkbox"/>	

Aceptar Cancelar Aplicar Ayuda

Tabla EstCarrera

Propiedades de la tabla: EstCarrera

General

Nombre: EstCarrera Permisos...

Propietario: dbo
Fecha de creación: 25/11/2008 13:53:12
Grupo de archivos: PRIMARY
Filas: 29

Columnas:

Cla...	Id.	Nombre	Tipo de d...	Tama...	Valor...	Predetermi
	<input checked="" type="checkbox"/>	IdEsCarr	int	4	<input type="checkbox"/>	
		IdEstudiante	int	4	<input checked="" type="checkbox"/>	
		IdCarrera	int	4	<input checked="" type="checkbox"/>	
		FechaCreado	datetime	8	<input checked="" type="checkbox"/>	
		FechaEliminado	datetime	8	<input checked="" type="checkbox"/>	
		Estado	int	4	<input checked="" type="checkbox"/>	

Aceptar Cancelar Aplicar Ayuda



Tabla Solicitud

Propiedades de la tabla: Solicitud

General

Nombre: Solicitud

Propietario: dbo

Fecha de creación: 25/11/2008 13:53:12

Grupo de archivos: PRIMARY

Filas: 26

Columnas:

Cla...	Id.	Nombre	Tipo de d...	Tama...	Valor...	Predete
	<input checked="" type="checkbox"/>	IdSolicitud	int	4	<input type="checkbox"/>	
		NumeroSolicitud	int	4	<input type="checkbox"/>	
		FechaSolicitud	datetime	8	<input type="checkbox"/>	
		AyoLectivo	int	4	<input type="checkbox"/>	
		AyoAcademico	int	4	<input type="checkbox"/>	
		TipoBeca	int	4	<input type="checkbox"/>	
		TipoSolicitud	int	4	<input type="checkbox"/>	

Aceptar Cancelar Aplicar Ayuda

Propiedades de la tabla: Solicitud

General

Nombre: Solicitud

Propietario: dbo

Fecha de creación: 25/11/2008 01:53:12 p.m.

Grupo de archivos: PRIMARY

Filas: 45

Columnas:

Cla...	Id.	Nombre	Tipo de d...	Tama...	Valor...	Predete
		IdProveedor	int	4	<input type="checkbox"/>	
		IdEstudiante	int	4	<input type="checkbox"/>	
		Estado	int	4	<input type="checkbox"/>	(1)
		Observacion	nvarchar	50	<input checked="" type="checkbox"/>	
		AutorizadaEI	datetime	8	<input checked="" type="checkbox"/>	
		RechazadaEI	datetime	8	<input checked="" type="checkbox"/>	
		ComentarioRech...	nvarchar	50	<input checked="" type="checkbox"/>	

Aceptar Cancelar Aplicar Ayuda



Tabla EstadoSolicitud

Propiedades de la tabla: EstadoSolicitud

General

Nombre: EstadoSolicitud Permisos...

Propietario: dbo
Fecha de creación: 25/11/2008 13:53:12
Grupo de archivos: PRIMARY
Filas: 3

Columnas:

Cla...	Id	Nombre	Tipo de d...	Tama...	Valor...	Predetermi
		IdEstado	int	4	<input type="checkbox"/>	
		Descripcion	nvarchar	50	<input checked="" type="checkbox"/>	

Aceptar Cancelar Aplicar Ayuda

Tabla Modalidad

Propiedades de la tabla: Modalidad

General

Nombre: Modalidad Permisos...

Propietario: dbo
Fecha de creación: 25/11/2008 13:53:11
Grupo de archivos: PRIMARY
Filas: 4

Columnas:

Cla...	Id	Nombre	Tipo de d...	Tama...	Valor...	Predetermi
	<input checked="" type="checkbox"/>	IdModalidad	int	4	<input type="checkbox"/>	
		Descripcion	char	18	<input checked="" type="checkbox"/>	

Aceptar Cancelar Aplicar Ayuda



Tabla EstModalidad

Propiedades de la tabla: EstadoSolicitud

General

Nombre: EstadoSolicitud

Propietario: dbo
Fecha de creación: 25/11/2008 13:53:12
Grupo de archivos: PRIMARY
Filas: 3

Columnas:

Cla...	Id.	Nombre	Tipo de d...	Tama...	Valor...	Predetermi
		IdEstado	int	4	<input type="checkbox"/>	
		Descripcion	nvarchar	50	<input checked="" type="checkbox"/>	

Tabla Operación

Propiedades de la tabla: Operacion

General

Nombre: Operacion

Propietario: dbo
Fecha de creación: 25/11/2008 13:53:12
Grupo de archivos: PRIMARY
Filas: 3

Columnas:

Cla...	Id.	Nombre	Tipo de d...	Tama...	Valor...	Predetermi
		IdOperacion	int	4	<input type="checkbox"/>	
		Descripcion	nvarchar	25	<input checked="" type="checkbox"/>	



Tabla ProveedorBeca

Propiedades de la tabla: ProveedorBeca

General

Nombre: ProveedorBeca Permisos...

Propietario: dbo
Fecha de creación: 25/11/2008 13:53:12
Grupo de archivos: PRIMARY
Filas: 4

Columnas:

Cla...	Id.	Nombre	Tipo de d..	Tama...	Valor...	Predeter
	<input checked="" type="checkbox"/>	IdProveedor	int	4	<input type="checkbox"/>	
		Identificacion	nvarchar	18	<input type="checkbox"/>	
		Nombre1	nvarchar	25	<input checked="" type="checkbox"/>	
		Nombre2	nvarchar	25	<input checked="" type="checkbox"/>	
		Apellido1	nvarchar	25	<input checked="" type="checkbox"/>	
		Apellido2	nvarchar	25	<input checked="" type="checkbox"/>	
		Telefono	nvarchar	9	<input checked="" type="checkbox"/>	

Aceptar Cancelar Aplicar Ayuda

Tabla Requisito

Propiedades de la tabla: Requisitos

General

Nombre: Requisitos Permisos...

Propietario: dbo
Fecha de creación: 25/11/2008 13:53:11
Grupo de archivos: PRIMARY
Filas: 5

Columnas:

Cla...	Id.	Nombre	Tipo de d..	Tama...	Valor...	Predetermi
	<input checked="" type="checkbox"/>	IdRequisito	int	4	<input type="checkbox"/>	
		Descripcion	nvarchar	25	<input checked="" type="checkbox"/>	

Aceptar Cancelar Aplicar Ayuda



Tabla RequisitoBeca

Propiedades de la tabla: RequisitoBeca

General

Nombre: RequisitoBeca Permisos...

Propietario: dbo
Fecha de creación: 25/11/2008 13:53:12
Grupo de archivos: PRIMARY
Filas: 24

Columnas:

Cla...	Id.	Nombre	Tipo de d...	Tama...	Valor...	Predetermi
	<input checked="" type="checkbox"/>	IdReBeca	int	4		<input type="checkbox"/>
		IdBeca	int	4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
		IdRequisito	int	4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Aceptar Cancelar Aplicar Ayuda

Tabla RequisitoSolicitud

Propiedades de la tabla: RequisitoSolicitud

General

Nombre: RequisitoSolicitud Permisos...

Propietario: dbo
Fecha de creación: 25/11/2008 13:53:12
Grupo de archivos: PRIMARY
Filas: 16

Columnas:

Cla...	Id.	Nombre	Tipo de d...	Tama...	Valor...	Predetermi
	<input checked="" type="checkbox"/>	IdRequisitoSol	int	4		<input type="checkbox"/>
		IdSolicitud	int	4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
		IdRequisito	int	4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Aceptar Cancelar Aplicar Ayuda



Tabla Tipo

Propiedades de la tabla: Tipo

General

Nombre: Tipo Permisos...

Propietario: dbo
Fecha de creación: 25/11/2008 13:53:12
Grupo de archivos: PRIMARY
Filas: 4

Columnas:

Cla...	Id.	Nombre	Tipo de d...	Tama...	Valor...	Predetermi
	<input checked="" type="checkbox"/>	IdTipo	int	4	<input type="checkbox"/>	
		Tipo	nvarchar	25	<input checked="" type="checkbox"/>	

Aceptar Cancelar Aplicar Ayuda

Tabla TipoProveedor

Propiedades de la tabla: TipoProveedor

General

Nombre: TipoProveedor Permisos...

Propietario: dbo
Fecha de creación: 25/11/2008 13:53:12
Grupo de archivos: PRIMARY
Filas: 3

Columnas:

Cla...	Id.	Nombre	Tipo de d...	Tama...	Valor...	Predetermi
	<input checked="" type="checkbox"/>	IdTipo	int	4	<input type="checkbox"/>	
		Descripcion	nvarchar	30	<input checked="" type="checkbox"/>	

Aceptar Cancelar Aplicar Ayuda



Tabla TipoSolicitud

Propiedades de la tabla: TipoSolicitud

General

Nombre: TipoSolicitud Permisos...

Propietario: dbo
Fecha de creación: 25/11/2008 13:53:11
Grupo de archivos: PRIMARY
Filas: 2

Columnas:

Cla...	Id.	Nombre	Tipo de d..	Tama...	Valor...	Predetermi
	<input checked="" type="checkbox"/>	IdTipoSolicitud	int	4	<input type="checkbox"/>	
		Descripcion	nvarchar	50	<input checked="" type="checkbox"/>	

Aceptar Cancelar Aplicar Ayuda

Tabla TipoUsuario

Propiedades de la tabla: TipoUsuario

General

Nombre: TipoUsuario Permisos...

Propietario: dbo
Fecha de creación: 25/11/2008 13:53:12
Grupo de archivos: PRIMARY
Filas: 2

Columnas:

Cla...	Id.	Nombre	Tipo de d..	Tama...	Valor...	Predetermi
	<input checked="" type="checkbox"/>	IdTipoUsuario	int	4	<input type="checkbox"/>	
		Descripcion	nvarchar	50	<input checked="" type="checkbox"/>	

Aceptar Cancelar Aplicar Ayuda



Tabla Usuario

Propiedades de la tabla: Usuario

General

Nombre: Usuario Permisos...

Propietario: dbo

Fecha de creación: 25/11/2008 13:53:12

Grupo de archivos: PRIMARY

Filas: 4

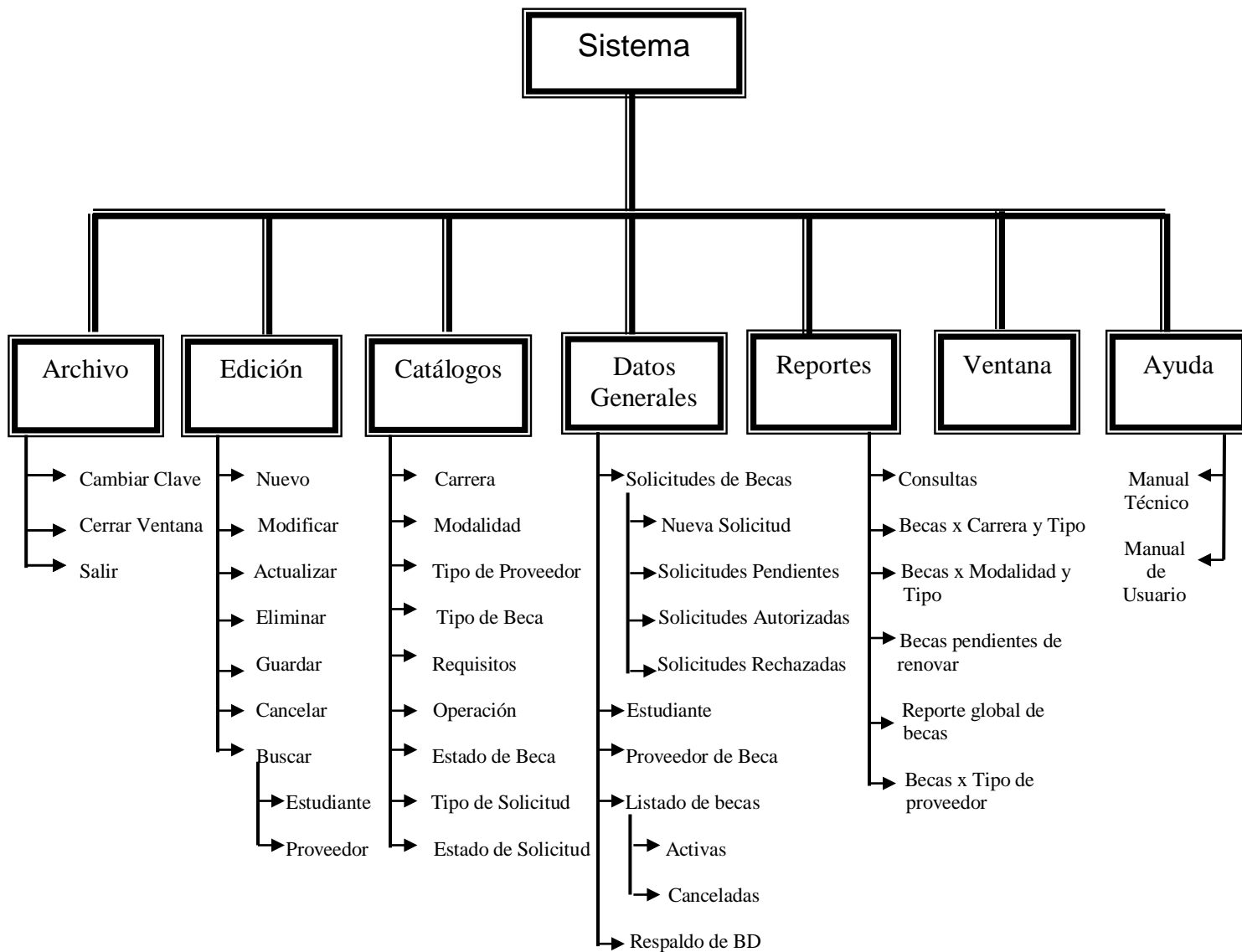
Columnas:

Cla...	Id.	Nombre	Tipo de d...	Tama...	Valor...	Predetermi
		IdUsuario	int	4	<input type="checkbox"/>	
		Nombre	nvarchar	50	<input checked="" type="checkbox"/>	
		Password	nvarchar	50	<input checked="" type="checkbox"/>	
		Tipo	int	4	<input checked="" type="checkbox"/>	
		Estado	int	4	<input checked="" type="checkbox"/>	

Aceptar Cancelar Aplicar Ayuda

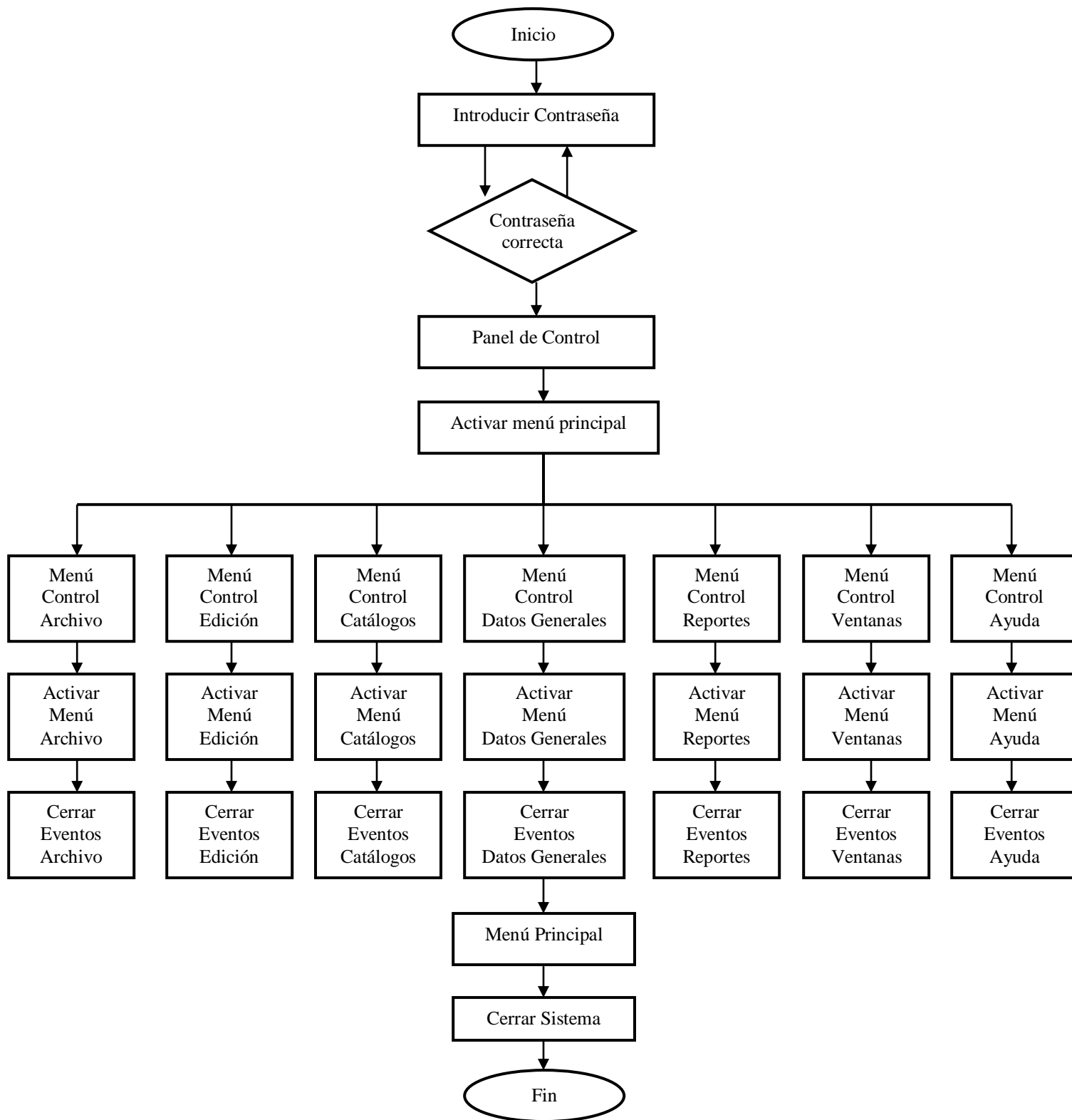


DISEÑO ARQUITECTÓNICO





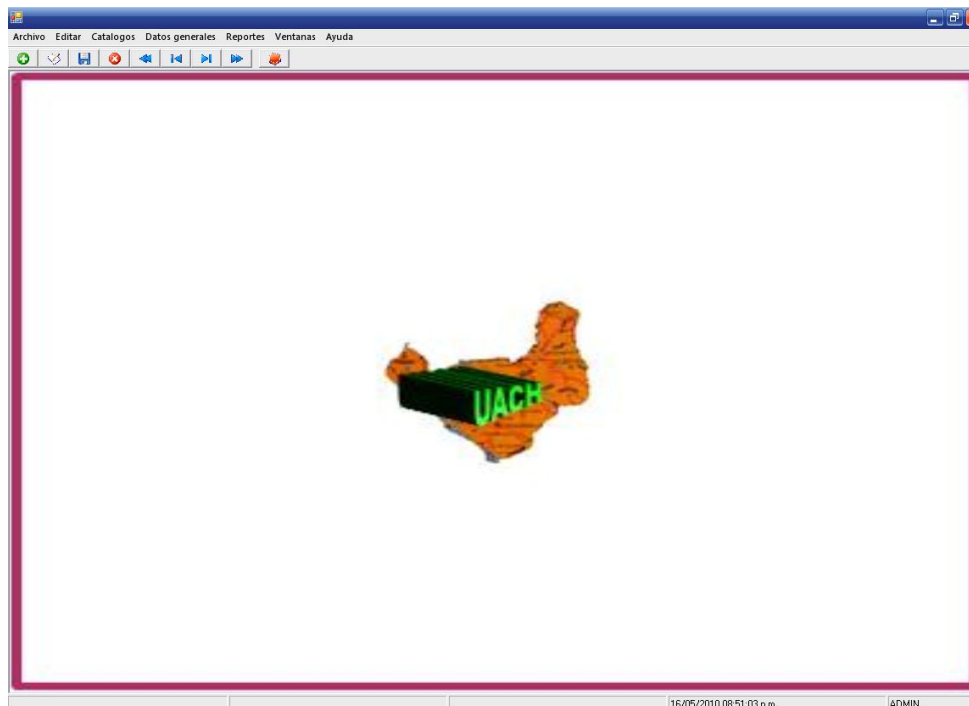
DISEÑO PROCEDIMENTAL





DISEÑO DE LA INTERFAZ

A continuación se muestra el formulario principal, que presenta las diferentes opciones necesarias para que el usuario interactúe con el sistema donde podrá llevar un mejor control de los alumnos que solicitan beca en la UACH.



Según lo antes descrito pasamos a detallar cada una de las opciones del formulario principal:

Archivo:

Encontramos las opciones de: **Cambiar Clave, Cerrar ventana y salir del sistema.**

Cambiar Clave: Podemos cambiar la contraseña existente por otra. Solo hay que ingresar la clave anterior y escribir la nueva clave y confirmar.

Editar:

Nos permite movernos entre los registros del formulario que está activo. Tenemos las opciones Nuevo, Modificar, Eliminar, Actualizar, Guardar, Cancelar y Buscar.

También podemos utilizar la barra de desplazamiento para movernos entre los registro del formulario que tenemos activo, se encuentra ubicada en la parte superior.





Catálogos

Encontramos los datos básicos que nos permite configurar el sistema para iniciar a trabajar en éste y suelen ser modificadas muy pocas veces. Incluye: Carrera, Modalidad, Tipo de proveedor, Tipo de beca, Requisitos, Operación, Estado de la beca, Tipo de solicitud, Estado de la solicitud.

Datos Generales:

Permite ingresar al sistema los datos generales de los estudiantes que solicitan becas y datos generales de proveedores de becas; encontramos las siguientes opciones: Solicitudes de becas, Estudiantes, ProveedorBeca y Becas.

Solicitudes de becas: incluye **Nueva solicitud**, solicitudes [pendientes, rechazadas y aprobadas].

The screenshot shows a software application window titled "Solicitud de becas." with a menu bar containing "Archivo", "Editar", "Catalogos", "Datos generales", "Reportes", "Ventanas", and "Ayuda". A dropdown menu is open under "Solicitudes de becas", showing options: "Nueva Solicitud", "Solicitudes Pendientes", "Solicitudes Rechazadas", and "Solicitudes Autorizadas".

The main form, titled "RECEPCIÓN DE SOLICITUDES DE BECA", contains the following fields and controls:

- Número:
- Fecha de solicitud:
- Año Lectivo:
- Año academico: año
- Datos personales de los estudiantes:
 - Carnet: ... Año academico actual:
 - Nombre completo:
 - Modalidad: Carrera:
- Solicitado a proveedor interno:
- Observación:
- Requisitos presentados:
 - 1 - Certificado Academico
 - 2 - Carta de solicitud
 - 3 - Deportes
 - 4 - Chica UACH
- Solicitud de:
- Beca solicitada:
-

The status bar at the bottom right shows the date and time "16/05/2010 07:56:13 m" and the user name "ADMIN".





Sistema de control de Becas UACH

Solicitudes Pendientes: Este formulario nos muestra el listado de solicitudes que aún no se les ha dado respuesta, permitiendo desde aquí Autorizar o Rechazar una solicitud. Sólo debemos dar click al inicio de la celda del estudiante en la barra de estado y seleccionar la operación a realizar.

Parameters of search:

Carnet: Año lectivo: 0 Año académico: Carrera: Modalidad: Solicitada a: Tipo de beca solicitada:

Numero	Carnet	Nombre del Estudiante	Carrera	Modalidad	Identificaci	Nombre del Proveedor	
		Nida Lisbeth Galeano Bauti	Contabilidad	Diario	6568-565	Edwin Antonio Melendez A	1
Autorizar solicitud		Nida Lisbeth Galeano Bauti	Ingenieria de Sistem	Diario	6568-565	Edwin Antonio Melendez A	1
Rechazar Solicitud		Maura Maria Perez Urey	Contabilidad	Diario	6568-565	Edwin Antonio Melendez A	3
Cerrar		Maura Maria Perez Urey	Ingenieria de Sistem	Diario	6568-565	Edwin Antonio Melendez A	3
		Caralia Mercedes Hernandez	Contabilidad	Diario	6568-565	Edwin Antonio Melendez A	4
15	CH-10-0378	Caralia Mercedes Hernandez	Ingenieria de Sistem	Diario	6568-565	Edwin Antonio Melendez A	4
16	CH-10-0199	Jessica Patricia Seidel Para	Contabilidad	Diario	4465-5144	Alvaro Alberto Fajardo Salga	2
16	CH-10-0199	Jessica Patricia Seidel Para	Ingenieria de Sistem	Diario	4465-5144	Alvaro Alberto Fajardo Salga	2
18	CH-09-0546	Afonso Jose Calin Fajardo	Administración de E	Diario	4465-5144	Alvaro Alberto Fajardo Salga	2
21	CH-09-0022	Mariela José Pacheco Hemá	Turismo	Diario	6568-565	Edwin Antonio Melendez A	3
21	CH-09-0022	Mariela José Pacheco Hemá	Turismo	Sabalino	6568-565	Edwin Antonio Melendez A	3
22	CH-09-0036	Nancy Adelaida Osconce Vilá	Turismo	Diario	4465-5144	Alvaro Alberto Fajardo Salga	2
23	CH-10-6172	Carmen Cecilia Martínez Me	Ingenieria Agronomi	Sabalino	6568-565	Edwin Antonio Melendez A	1
23	CH-10-6172	Carmen Cecilia Martínez Me	Derecho	Sabalino	6568-565	Edwin Antonio Melendez A	1
24	CH-09-0222	Ricardo Antonio Gomez Cas	Administración de E	Sabalino	6568-565	Edwin Antonio Melendez A	2
24	CH-09-0222	Ricardo Antonio Gomez Cas	Derecho	Sabalino	6568-565	Edwin Antonio Melendez A	2

Estudiante: En este formulario se puede: Agregar un nuevo estudiante , Modificar o eliminar el registro de un estudiante existente . Esto se puede realizar a través de la Barra de Desplazamiento ó por el menú Editar que permitirá editar datos del formulario Activo.

frmEstudiante

Carnet:

Primer nombre: Segundo nombre:

Primer apellido: Segundo apellido:

Fecha de Nacimiento: Edad:

Sexo: Teléfono:

Dirección domiciliar:

Modalidad: Carrera:

Proveedor de la beca: Año Académico:

Tipo de Beca: Fecha otorgada:



Sistema de control de Becas UACH

Proveedor de Beca: En este formulario podemos agregar un nuevo proveedor y hacerle modificaciones desde la barra de desplazamiento o desde el menú editar, esto permitirá la administración de los datos del proveedor dándole de alta o de baja.

The screenshot shows a Windows application window titled 'Proveedores de la beca'. The main menu includes 'Archivo', 'Editar', 'Catalogos', 'Datos generales', 'Reportes', 'Ventanas', and 'Ayuda'. A dropdown menu is open under 'Datos generales', showing options: 'Solicitudes de becas', 'Estudiantes', 'Proveedor Beca', 'Listado de becas', and 'Respaldo de BD'. The form fields are as follows:

- Identificación: 6568-565
- Primer nombre: Edwin
- Segundo nombre: A
- Primer apellido: Melendez
- Segundo apellido: A
- Institución: UACH
- Cargo: Vice Rector
- Teléfono: 2466
- Tipo: INTERNO
- Dirección domiciliar: Chinandega

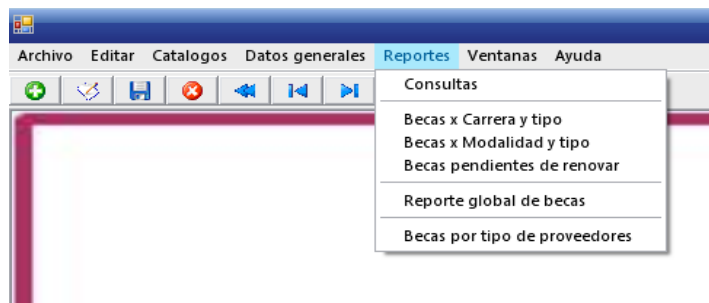
The status bar at the bottom indicates the date and time as 16/05/2010 08:00:22 p.m. and the user as ADMIN.

Respaldo de Bases de Datos: Desde este formulario podemos realizar respaldo de la Bases de Datos; el sistema automáticamente manda la opción de guardar en la unidad C:\, pero tenemos la opción de guardar en otro dispositivo dando click en **Examinar** y seleccionar el dispositivo requerido.

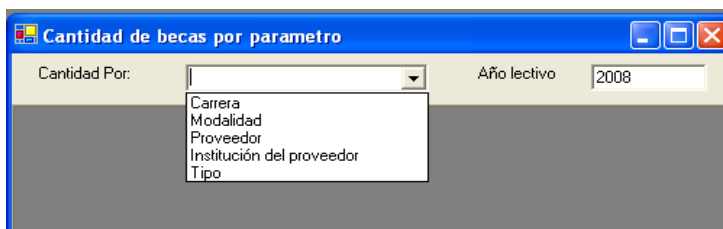
The screenshot shows the 'Ruta del respaldo' dialog box with the 'Ruta' field set to 'C:\'. Below the field are buttons for 'Examinar', 'Aceptar', and 'Cancelar'. An open file explorer window titled 'Buscar carpeta' shows the contents of the C:\ drive, including folders like 'Escritorio', 'Mis documentos', 'Mi PC', 'SISTEMA (C:)', 'Unidad DVD-RAM (D:)', 'RESPALDOS (E:)', 'Disco extraíble (F:)', 'Unidad de CD (G:)', and 'REYNA (I:)'. The status bar at the bottom indicates the date and time as 16/05/2010 08:22:27 p.m. and the user as ADMIN.



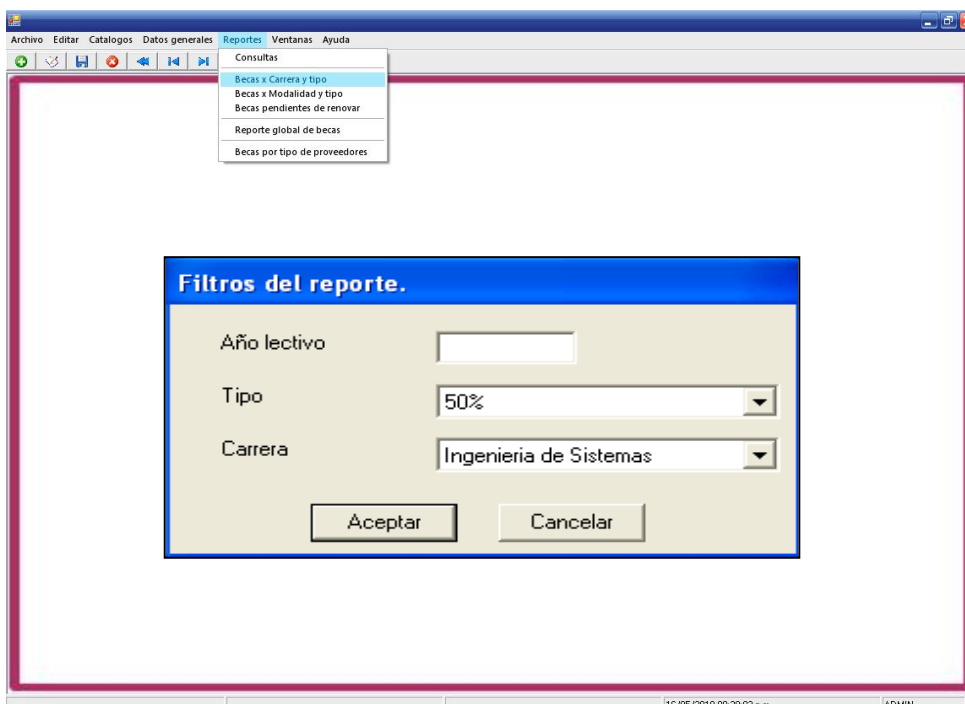
Reportes: encontramos las siguientes opciones:



Consulta: Para realizar una consulta debemos seleccionar el año que queremos mostrar. En este formulario se pueden realizar consultas por: modalidad, carrera, proveedor, institución del proveedor y tipo.



Becas x carrera y tipo Este formulario permite buscar becas mediante parámetros como: Año lectivo, Tipo y carrera.





Sistema de control de Becas UACH

Becas x modalidad y tipo: Este formulario permite buscar becas mediante parámetros como: Año lectivo, Tipo [% de becas] y Modalidad, una vez seleccionados los parámetros damos click en aceptar.

The screenshot shows a software window titled 'Reportes' with a menu open. The menu item 'Becas x Modalidad y tipo' is selected. A dialog box titled 'Filtros del reporte.' is displayed in the center. It contains three input fields: 'Año lectivo' (empty text box), 'Tipo' (dropdown menu with '50%' selected), and 'Modalidad' (dropdown menu with 'Presencial' selected). At the bottom of the dialog are two buttons: 'Aceptar' and 'Cancelar'. The status bar at the bottom of the window shows the date and time '16/05/2010 08:40:27 p.m.' and the user 'ADMIN'.

Becas pendientes por renovar: Este formulario permite buscar becas pendientes por renovar mediante parámetros como: Año Lectivo, Tipo [% de becas] y Carrera una vez seleccionados los parámetros damos click en aceptar.

The screenshot shows the same software window with the menu item 'Becas pendientes de renovar' selected. The 'Filtros del reporte.' dialog box is shown with three input fields: 'Año lectivo' (empty text box), 'Tipo' (dropdown menu with '50%' selected), and 'Carrera' (dropdown menu with 'Ingeniería de Sistemas' selected). At the bottom of the dialog are two buttons: 'Aceptar' and 'Cancelar'. The status bar at the bottom of the window shows the date and time '16/05/2010 08:42:39 p.m.' and the user 'ADMIN'.



Sistema de control de Becas UACH

Reporte global de becas: Este formulario muestra un reporte global de becas mediante el parámetro del Año Lectivo, una vez seleccionado el año lectivo damos click en aceptar.

Archivo Editar Catalogos Datos generales **Reportes** Ventanas Ayuda

Consultas

- Becas x Carrera y tipo
- Becas x Modalidad y tipo
- Becas pendientes de renovar
- Reporte global de becas**
- Becas por tipo de proveedores

Filtros del reporte.

Año lectivo

Aceptar Cancelar

16/05/2010 08:44:32 p.m. ADMIN

Reportes de Becas x tipo de proveedor: Este formulario permite buscar becas mediante los parámetros de Año lectivo y Tipo [Interno, Externo y Convenio], una vez seleccionados los parámetros damos click en aceptar.

Archivo Editar Catalogos Datos generales **Reportes** Ventanas Ayuda

Consultas

- Becas x Carrera y tipo
- Becas x Modalidad y tipo
- Becas pendientes de renovar
- Reporte global de becas
- Becas por tipo de proveedores**

Filtros del reporte.

Año lectivo

Tipo

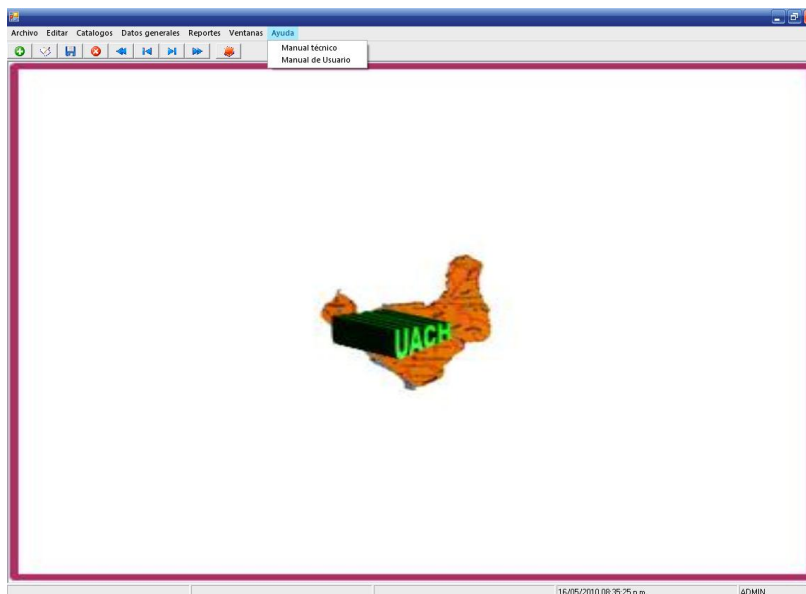
Aceptar Cancelar

16/05/2010 08:46:59 p.m. ADMIN

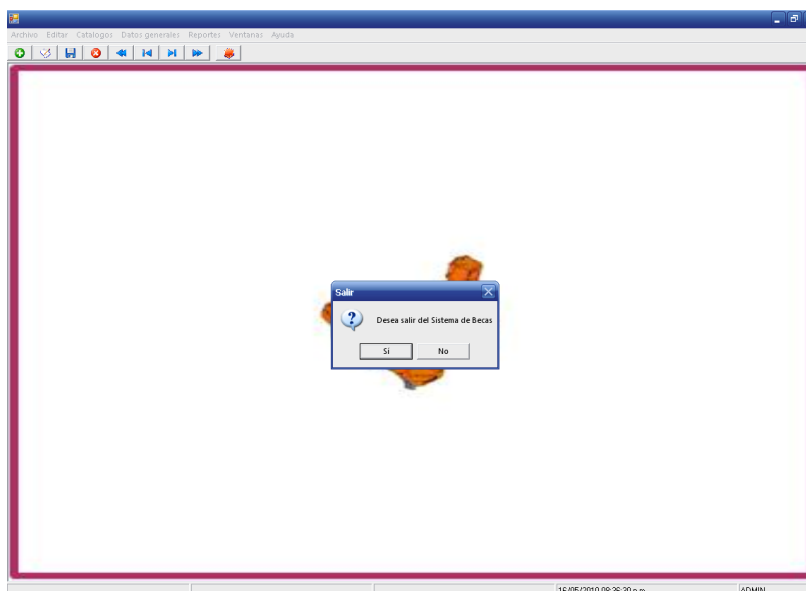


Sistema de control de Becas UACH

Ayuda: Dispone de un Manual de Usuario que indica como funciona cada uno de los formularios del sistema, pero además se le agregó el Manual Técnico que nos indica paso a paso la instalación del sistema y de la base de datos como también algunos procedimientos almacenados, que le ayude al usuario interactuar con el sistema.



Salir del Sistema de Becas: Para salir del sistema damos click en archivo salir del sistema o bien cerramos la ventana del sistema en ambos caso nos mostrará un mensaje de confirmación de salir damos click en sí.





CODIFICACIÓN

La aplicación consta de un pequeño módulo cuyo código permite centrar cada formulario que hace referencia a él. A continuación se muestra la codificación de algunos formularios principales:

Formulario Contraseña

The image shows a Windows-style dialog box titled "Control de acceso al sistema". It has a standard title bar with minimize, maximize, and close buttons. The dialog contains two text input fields: the first is labeled "Usuario" and the second is labeled "Contraseña". Below the input fields are two buttons: "Aceptar" (Accept) and "Cancelar" (Cancel).

```
Dim cndb As New clsConexionesDb
Dim da As SqlDataAdapter
Dim dt As DataTable
Dim cn As New SqlConnection(cndb.SqlCnPrincipal.ConnectionString)
Dim frmP As New frmPrincipal
Dim IdUsuarioBusc As Integer
Dim TipoUsuario As Integer

Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnAceptar.Click
    Static Intent As Integer
    Const Tintent = 3
    Dim Usuario As String
    Dim Contraseñas As String
    Try
        If txtNombre.Text = "" Or (txtNombre.Text Is Nothing) Then
            ElseIf dt.Rows.Count = 0 Then
                Intent = Intent + 1
                If Tintent = Intent Then
                    MsgBox("Lo Siento.. No hay más Oportunidades", vbCritical, "Acceso Denegado")
                End
            Else
                MsgBox("Presione el boton Aceptar e intente de nuevo", vbInformation, "Password Incorrecto")
                txtContraseña.Focus()
                txtContraseña.Text = ""
            End If
            ElseIf (ValidarUsuario(LCase(txtNombre.Text), txtContraseña.Text)) Then
                frmP.NombreUsuario = txtNombre.Text
                frmP.IdUsuario = IdUsuarioBusc
                frmP.TipoUsuario = TipoUsuario
                MsgBox("Password Correcto, Bienvenido al Sistema" & " " & txtNombre.Text, vbExclamation, "Acceso al Sistema control de Becas")
                Me.Hide()
                frmP.Show()
            Else
```



Sistema de control de Becas UACH

```
Intent = Intent + 1
If Tintent = Intent Then
MsgBox("Lo Siento.. No hay más Oportunidades", vbCritical, "Acceso Denegado")

End
Else
MsgBox("Presione el boton Aceptar e intente de nuevo", vbInformation, "Password Incorrecto")
txtContraseña.Focus()
txtContraseña.Text = ""
End If
End If
Exit Sub
Catch ex As Exception
MessageBox.Show(ex.Message)
End Try
End Sub

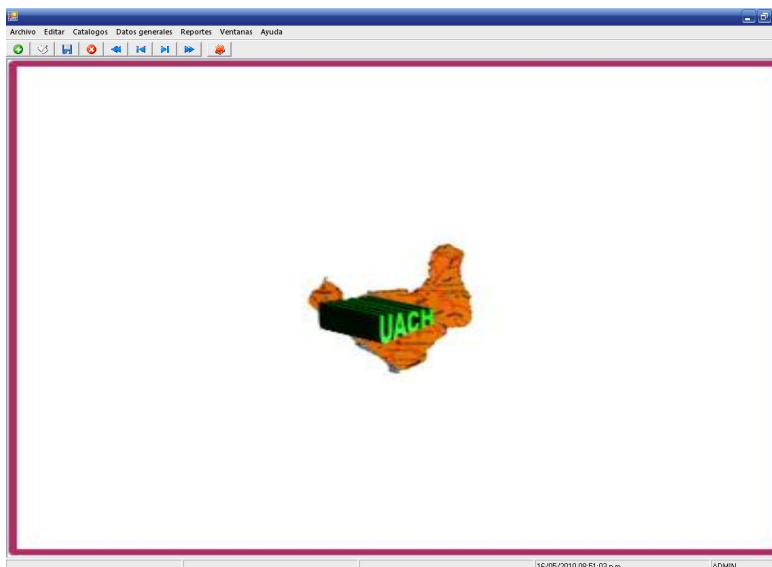
Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnCancelar.Click
End
End Sub

Private Sub frmContraseña_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
Try
da = New SqlDataAdapter("Select * From Usuario", cnbd.SqlCnPrincipal.ConnectionString)
dt = New DataTable
da.Fill(dt)
Catch ex As Exception
MessageBox.Show(ex.Message)
End Try
End Sub

Function ValidarUsuario(ByVal Nombre As String, ByVal Clave As String) As Boolean
For Each fil As DataRow In dt.Rows
If (Nombre = fil("Nombre")) And (Clave = fil("Password")) Then
ValidarUsuario = True
IdUsuarioBusc = fil("IdUsuario")
TipoUsuario = fil("Tipo")
Exit Function
End If
Next
ValidarUsuario = False
End Function
End Class
```



Formulario Principal del sistema de becas



```
Public NombreUsuario As String = ""  
Public IdUsuario As Integer  
Public TipoUsuario As String = ""
```

```
Private Sub mnuEstudiantes_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles  
mnuEstudiantes.Click  
frmEst = New frmEstudiante  
Me.Cursor = Cursors.WaitCursor  
frmEst.MdiParent = Me  
frmEst.Show()  
Me.Cursor = Cursors.Default  
End Sub
```

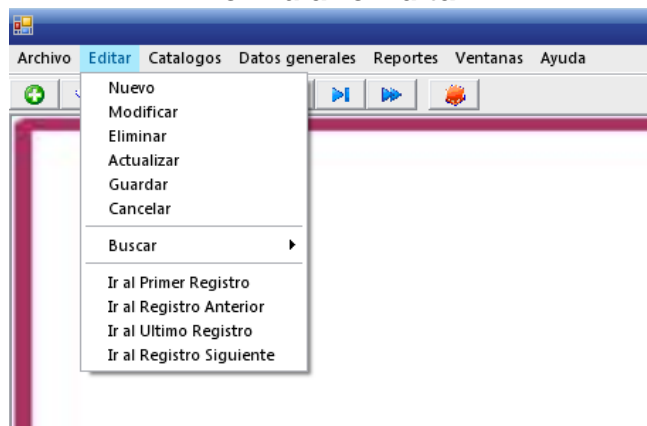
```
Private Sub mnuProveedorBeca_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles  
mnuProveedorBeca.Click  
frmPBeca = New frmProveedorBeca  
Me.Cursor = Cursors.WaitCursor  
frmPBeca.MdiParent = Me  
frmPBeca.Show()  
Me.Cursor = Cursors.Default  
End Sub
```

```
Private Sub mnuCerrar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles mnuCerrar.Click  
If Not (ActiveMdiChild Is Nothing) Then  
ActiveMdiChild.Close()  
End If  
End Sub
```

```
Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Timer1.Tick  
Me.sbFechaHora.Text = Now  
End Sub
```




Formulario Editar



```
Private Sub tbPrincipal_ButtonClick(ByVal sender As System.Object, ByVal e As System.Windows.Forms.ToolBarButtonClickEventArgs) Handles tbPrincipal.ButtonClick
If Not (Me.ActiveMdiChild Is Nothing) Then
```

```
    Select Case (e.Button.Text)
```

```
        'Movimiento de registros
```

```
        Case "Ir al primero"
```

```
            mnulrPrimero_Click(Nothing, Nothing)
```

```
        Case "Ir al anterior"
```

```
            mnulrAnterior_Click(Nothing, Nothing)
```

```
        Case "Ir al siguiente"
```

```
            mnulrSiguiente_Click(Nothing, Nothing)
```

```
        Case "Ir al ultimo"
```

```
            mnulrUltimo_Click(Nothing, Nothing)
```

```
        'Operaciones con registros
```

```
        Case "Nuevo"
```

```
            mnuNuevo_Click(Nothing, Nothing)
```

```
        Case "Modificar"
```

```
            mnuModificar_Click(Nothing, Nothing)
```

```
        Case "Actualizar"
```

```
            mnuGuardar_Click(Nothing, Nothing)
```

```
        Case "Eliminar"
```

```
            mnuEliminar_Click(Nothing, Nothing)
```

```
        Case "Cancelar"
```

```
            mnuCancelar_Click(Nothing, Nothing)
```

```
    End Select
```

```
    Else
```

```
        MessageBox.Show("No hay ningun formulario Activo", "Barra de herramienta", MessageBoxButtons.OK,
```

```
        MessageBoxIcon.Exclamation)
```

```
    End If
```

```
End Sub
```

```
Private Sub MenuItem4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MenuItem4.Click
```

```
    Dim frmC As New frmCatalogos
```

```
    frmC.NombreTabla = "Carrera"
```

```
    frmC.MdiParent = Me
```

```
    frmC.Show()
```

```
End Sub
```

```
Private Sub MenuItem5_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MenuItem5.Click
```

```
    Dim frmC As New frmCatalogos
```



Sistema de control de Becas UACH

```
frmC.NombreTabla = "Modalidad"  
frmC.MdiParent = Me  
frmC.Show()  
End Sub
```

```
Private Sub MenuItem7_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MenuItem7.Click  
Dim frmC As New frmCatalogos  
frmC.NombreTabla = "TipoProveedor"  
frmC.MdiParent = Me  
frmC.Show()  
End Sub
```

```
Private Sub MenuItem8_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MenuItem8.Click  
Dim frmC As New frmCatalogos  
frmC.NombreTabla = "Tipo"  
frmC.MdiParent = Me  
frmC.Show()  
End Sub
```

```
Private Sub MenuItem9_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MenuItem9.Click  
Dim frmC As New frmCatalogos  
frmC.NombreTabla = "Requisitos"  
frmC.MdiParent = Me  
frmC.Show()  
End Sub
```

```
Private Sub MenuItem10_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MenuItem10.Click  
Dim frmC As New frmCatalogos  
frmC.NombreTabla = "Operacion"  
frmC.MdiParent = Me  
frmC.Show()  
End Sub
```

```
Private Sub MenuItem11_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MenuItem11.Click  
Dim frmC As New frmCatalogos  
frmC.NombreTabla = "EstadoBeca"  
frmC.MdiParent = Me  
frmC.Show()  
End Sub
```

```
Private Sub mnuSalir_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles mnuSalir.Click  
End  
End Sub
```

```
Private Sub frmPrincipal_Closing(ByVal sender As Object, ByVal e As System.ComponentModel.CancelEventArgs) Handles MyBase.Closing  
If (MessageBox.Show("Desea salir del Sistema de Becas", "Salir", MessageBoxButtons.YesNo, MessageBoxIcon.Question)  
= DialogResult.No) Then  
e.Cancel = True  
End If  
End Sub
```

```
Private Sub mnuCancelar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles mnuCancelar.Click  
If Not (Me.ActiveMdiChild Is Nothing) Then  
If (Me.ActiveMdiChild.Name = "frmEstudiante") Then  
clsControlRegistro.OperacionReg(frmEst,"dsRegistroBecas", frmEst.objDSEstudiantes, frmEst.NombreTabla, "Cancelar",  
frmEst.sqlCnEst, frmEst.sqlDA)  
If frmEst.PosAnterior >= 0 Then  
clsControlRegistro.DesplazarReg(frmEst.PosAnterior,frmEst, frmEst.objDSEstudiantes, frmEst.NombreTabla)  
End If  
clsControlRegistro.BloquearControles(frmEst)  
ElseIf (Me.ActiveMdiChild.Name = "frmProveedorBeca") Then
```



Sistema de control de Becas UACH

```
clsControlRegistro.OperacionReg(frmPBeca,"dsRegistroBecas",
frmPBeca.objDSProveedorBeca, "ProveedorBeca","Cancelar", frmPBeca.sqlCnPBeca, frmPBeca.sqlDA)
If frmPBeca.PosAnterior >= 0 Then
clsControlRegistro.DesplazarReg(frmPBeca.PosAnterior, frmPBeca, frmPBeca.objDSProveedorBeca, "ProveedorBeca")
End If
clsControlRegistro.BloquearControles(frmPBeca)
End If
Else
MessageBox.Show("No hay ningun formulario Activo", "Nuevo Registro", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
End If
End Sub
Private Sub mnuNuevaBeca_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
mnuNuevaBeca.Click
Dim frmBeca As New frmBecas
frmBeca.TipoOperacion = "Nuevo"
frmBeca.MdiParent = Me
frmBeca.Show()
End Sub

Private Sub mnuListaBeca_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
mnuListaBeca.Click
End Sub
Private Sub mnuCarretaTipo_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
mnuCarretaTipo.Click
Dim frm As New frmParametrosRpt
frm.MdiParent = Me
frm.NombreRpt = "rptBecasXCarrera"
frm.Show()
End Sub
Private Sub mnuModalidadTipo_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
mnuModalidadTipo.Click
Dim frm As New frmParametrosRpt
frm.MdiParent = Me
frm.NombreRpt = "rptBecasXModalidad"
frm.Show()
End Sub
Private Sub mnuCantCarrera_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Dim frmC As New frmCatalogos
frmC.NombreTabla = "vwCantidadBecasXCarrera"
frmC.MdiParent = Me
frmC.Show()
End Sub

Private Sub mnuCantModalidad_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Dim frmC As New frmCatalogos
frmC.NombreTabla = "vwCantidadBecasXModalidad"
frmC.MdiParent = Me
frmC.Text = "Cantidad de becas x modalidad"
frmC.Show()
End Sub
Private Sub mnuActivas_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles mnuActivas.Click
Dim frmLBeca As New frmListadoBecas
frmLBeca.Estado = "Activas"
frmLBeca.MdiParent = Me
frmLBeca.Show()
End Sub

Private Sub mnuCanceladas_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
mnuCanceladas.Click
Dim frmLBeca As New frmListadoBecas
```



Sistema de control de Becas UACH

```
frmLBeca.Estado = "Canceladas"
frmLBeca.MdiParent = Me
frmLBeca.Show()
End Sub
Private Sub mnuConsulta_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
mnuConsulta.Click
Dim frmC As New frmListaBecasXCantidad
frmC.MdiParent = Me
frmC.Show()
End Sub

Private Sub mnuRenovarPendiente_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
mnuRenovarPendiente.Click
Dim frm As New frmParametrosRpt
frm.MdiParent = Me
frm.NombreRpt = "rptBecasXRenovar"
frm.Show()
End Sub

Private Sub frmPrincipal_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
sUser.Text = UCase(NombreUsuario)
End Sub
Private Sub frmPrincipal_Closed(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.Closed
End
End Sub
Private Sub mnuNuevaSolicitud_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
mnuNuevaSolicitud.Click
Dim frmSol As New frmSolicitud
frmSol.ShowDialog()
End Sub

Private Sub mnuSolicitudesPendientes_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
mnuSolicitudesPendientes.Click
Dim frmSolPendientes As New frmListaSolicitudesPendientes
frmSolPendientes.MdiParent = Me
frmSolPendientes.Estado = 1
frmSolPendientes.Show()
End Sub
Private Sub mnuSolicitudesRechazadas_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
mnuSolicitudesRechazadas.Click
Dim frmSolRechazadas As New frmListaSolicitudesPendientes
frmSolRechazadas.MdiParent = Me
frmSolRechazadas.Estado = 21
frmSolRechazadas.Show()
End Sub
Private Sub mnuSolicitudesAutorizadas_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
mnuSolicitudesAutorizadas.Click
Dim frmSolAutorizadas As New frmListaSolicitudesPendientes
frmSolAutorizadas.MdiParent = Me
frmSolAutorizadas.Estado = 11
frmSolAutorizadas.Show()
End Sub

Private Sub mnuBuscarEstudiante_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
mnuBuscarEstudiante.Click
Dim frmBuscar As New frmBuscador
frmBuscar.NombreTabla = "Estudiante"
frmBuscar.ShowDialog()
End Sub
```



Sistema de control de Becas UACH

```
Private Sub mnuBuscarProveedor_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles  
mnuBuscarProveedor.Click  
Dim frmBuscar As New frmBuscador  
frmBuscar.NombreTabla = "ProveedorBeca"  
frmBuscar.ShowDialog()  
End Sub
```

```
Private Sub mnuTipoSolicitud_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles  
mnuTipoSolicitud.Click  
Dim frmC As New frmCatalogos  
frmC.NombreTabla = "TipoSolicitud"  
frmC.MdiParent = Me  
frmC.Show()  
End Sub
```

```
Private Sub mnuEstadoSolicitud_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles  
mnuEstadoSolicitud.Click  
Dim frmC As New frmCatalogos  
frmC.NombreTabla = "EstadoSolicitud"  
frmC.MdiParent = Me  
frmC.Show()  
End Sub
```

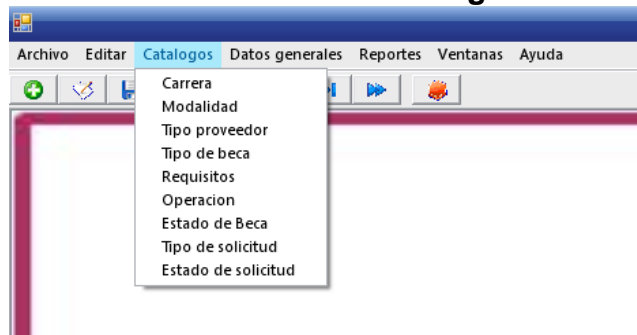
```
Private Sub mnurptGlobal_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles  
mnurptGlobal.Click  
Dim frm As New frmParametrosRpt  
frm.MdiParent = Me  
frm.NombreRpt = "rptGlobal"  
frm.Show()  
End Sub
```

```
Private Sub mnuBecasInternas_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles  
mnuBecasInternas.Click  
Dim frm As New frmParametrosRpt  
frm.MdiParent = Me  
frm.NombreRpt = "rptEstudiantesXTipoBeca"  
frm.Show()  
End Sub
```

```
Private Sub MenuItem20_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MenuItem20.Click  
Dim frm As New frmCambiarClave  
frm.Usuario = NombreUsuario  
frm.ShowDialog()  
End Sub  
End Class
```



Formulario de catálogo



```
Private Sub frmCatalogos_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
    dgDatos.ReadOnly = True
    Me.Text = "Datos de la tabla: " & NombreTabla
    ConfigurarDatos(NombreTabla)
    dgDatos.Focus()
End Sub
```

```
Sub ConfigurarDatos(ByVal Tabla As String)
    Dim cn As New SqlConnection(ConDb.SqlCnPrincipal.ConnectionString)
    Dim sqlda As New SqlDataAdapter("Select * From " & Tabla, cn)
    Try
        If (ds.Tables.Count > 0) Then
            If ds.Tables(Tabla).Rows.Count > 0 Then
                ds.Tables(Tabla).Clear()
            End If
        End If
        sqlda.Fill(ds, Tabla)
        If (ds.Tables(Tabla).Rows.Count > 0) Then
            dgDatos.DataSource = Nothing
            dgDatos.DataSource = ds.Tables(Tabla)
        End If
    Catch ex As Exception
        MessageBox.Show(ex.Message)
    End Try
End Sub
```

```
Private Sub mnuNuevo_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles mnuNuevo.Click
    Dim frmAdd As New frmAgregarDatos
    frmAdd.TipoOperacion = "Insertando"
    frmAdd.ValorInicial = ""
    frmAdd.IdActualizar = ""
    frmAdd.NombreTabla = NombreTabla
    frmAdd.NombreCampo = ds.Tables(NombreTabla).Columns(1).ColumnName
    frmAdd.ShowDialog()
    ConfigurarDatos(NombreTabla)
End Sub
```

```
Private Sub mnuModificar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles mnuModificar.Click
    If (Id <> Nothing) Then
        Dim frmAdd As New frmAgregarDatos
        frmAdd.TipoOperacion = "Actualizando"
        frmAdd.ValorInicial = Valor
        frmAdd.IdActualizar = Id
        frmAdd.NombreTabla = NombreTabla
    End If
End Sub
```



Sistema de control de Becas UACH

```
frmAdd.NombreCampo = ds.Tables(NombreTabla).Columns(1).ColumnName
frmAdd.ShowDialog()

ConfigurarDatos(NombreTabla)
Else
MessageBox.Show("Debe seleccionar una celda o cambiar la celda activa!", "Modificar registros", MessageBoxButtons.OK,
MessageBoxIcon.Information)
End If
End Sub

Private Sub dgDatos_CurrentCellChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles
dgDatos.CurrentCellChanged
If Not (dgDatos.Item(dgDatos.CurrentRowIndex, 0) Is System.DBNull.Value) Then
Id = dgDatos.Item(dgDatos.CurrentRowIndex, 0)
Valor = dgDatos.Item(dgDatos.CurrentRowIndex, 1)
Else
Id = ""
Valor = ""
End If
End Sub

Private Sub mnuCerrar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles mnuCerrar.Click
Me.Close()
End Sub

Private Sub dgDatos_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles dgDatos.Click
If Not (dgDatos.Item(dgDatos.CurrentRowIndex, 0) Is System.DBNull.Value) Then
Id = dgDatos.Item(dgDatos.CurrentRowIndex, 0)
Valor = dgDatos.Item(dgDatos.CurrentRowIndex, 1)
Else
Id = ""
Valor = ""
End If
End Sub
End Class
```



Formulario de Datos Generales:

Nueva Solicitud

Sub LlenarCbos()

Dim ds As New DataSet

Dim da As New SqlDataAdapter("Select IdEstado, Estado From EstadoBeca", sqlCn)

da = New SqlDataAdapter("Select '-1' IdTipo, " Tipo UNION Select IdTipo, Tipo From Tipo ORDER BY Tipo", sqlCn)

da.Fill(ds, "Tipo")

cboTipoBeca.DataSource = ds.Tables("Tipo")

cboTipoBeca.DisplayMember = "Tipo"

cboTipoBeca.ValueMember = "IdTipo"

da = New SqlDataAdapter("Select '-1' Identificacion, " Nombre UNION Select Identificacion, (Identificacion + ' - ' + ISNULL(Nombre1, ") + ' ' + ISNULL(Nombre2, ") + ' ' + ISNULL(Apellido1, ") + ' ' + ISNULL(Apellido2, ")) Nombre From ProveedorBeca WHERE IdTipo = 1 ORDER BY Nombre --INTERNOS", sqlCn)

da.Fill(ds, "ProveedorBeca")

cbIdentProveedor.DataSource = ds.Tables("ProveedorBeca")

cbIdentProveedor.DisplayMember = "Nombre"

cbIdentProveedor.ValueMember = "Identificacion"

da = New SqlDataAdapter("Select '-1' IdTipoSolicitud, " Descripcion UNION Select IdTipoSolicitud, Descripcion From TipoSolicitud ORDER BY Descripcion", sqlCn)

da.Fill(ds, "TipoSolicitud")

cboTipoSolicitud.DataSource = ds.Tables("TipoSolicitud")

cboTipoSolicitud.DisplayMember = "Descripcion"

cboTipoSolicitud.ValueMember = "IdTipoSolicitud"

End Sub

Sub ObtenerConsecutivo()

sqlCmd = New SqlCommand("EXEC UltimoNumeroSol", sqlCn)

Try

If (sqlCn.State = ConnectionState.Closed) Then

sqlCn.Open()

End If



Sistema de control de Becas UACH

```
txtNumeroSolicitud.Text = CInt(sqlCmd.ExecuteScalar()) + 1
Catch ex As Exception
MessageBox.Show(ex.Message)
Finally
sqlCn.Close()
End Try
End Sub
Private Sub frmSolicitud_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
ObtenerConsecutivo()
LlenarCbos()
CrearCK()
End Sub
Private Sub txtCarnet_Leave(ByVal sender As Object, ByVal e As System.EventArgs) Handles txtCarnet.Leave
If txtCarnet.Text <> "" Then
AsignarDatosEstudiante(txtCarnet.Text)
End If
End Sub

Function ObtenerAyoAcademico(ByVal id As Integer) As String
Dim cmd As New SqlCommand("Select AyoAcademico from vwListaBecasHdr Where IdBeca = " & id.ToString(), sqlCn)
Try
If (sqlCn.State = ConnectionState.Closed) Then
sqlCn.Open()
End If
ObtenerAyoAcademico = cmd.ExecuteScalar()
Catch ex As Exception
ObtenerAyoAcademico = ""
End Try
If ObtenerAyoAcademico Is Nothing Then
ObtenerAyoAcademico = ""
End If
End Function

Function ObtenerIdBecaUltima() As Integer
Dim cmd As New SqlCommand("Exec SeleccionarBecaXEstudiante " + txtIdEstudiante.Text, sqlCn)
Try
If sqlCn.State = ConnectionState.Closed Then
sqlCn.Open()
End If
Dim id As Integer
ObtenerIdBecaUltima = cmd.ExecuteScalar 'Identificador de la beca
Catch ex As Exception
ObtenerIdBecaUltima = 0
Finally
sqlCn.Close()
End Try
End Function

Sub AsignarDatosEstudiante(ByVal Carnet As String)
sqlda = New SqlDataAdapter("EXEC SeleccionarEstudianteXCarnet " & Carnet & "", sqlCn)
Try
dtEstudiante = New DataTable
sqlda.Fill(dtEstudiante)
If (dtEstudiante.Rows.Count > 0) Then
txtIdEstudiante.Text = dtEstudiante.Rows(0)("IdEstudiante")
txtNombreEstudiante.Text = dtEstudiante.Rows(0)("NombreCompleto")
txtCarrera.Text = dtEstudiante.Rows(0)("Carrera")
txtModalidad.Text = dtEstudiante.Rows(0)("Modalidad")
txtAyoAcademicoActual.Text = ObtenerAyoAcademico(ObtenerIdBecaUltima()).ToString()
Else
```



Sistema de control de Becas UACH

```
MessageBox.Show("No existe estudiante con el carnet suministrado", "Error al buscar el estudiante",
MessageBoxButtons.OK, MessageBoxIcon.Error)
txtCarnet.Text = ""
txtNombreEstudiante.Text = ""
txtCarrera.Text = ""
txtModalidad.Text = ""
txtAyoAcademicoActual.Text = ""
End If
Catch ex As Exception
MessageBox.Show(ex.Message)
End Try
End Sub

Private Sub cmdBuscarEstudiante_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
cmdBuscarEstudiante.Click
Dim frmBuscar As New frmBuscador
frmBuscar.NombreTabla = "Estudiante"
frmBuscar.ShowDialog()
If (frmBuscar.LlaveReturn <> "") Then
txtCarnet.Text = frmBuscar.LlaveReturn
txtCarnet_Leave(Nothing, Nothing)
End If
End Sub

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
Dim frmBuscar As New frmBuscador
frmBuscar.NombreTabla = "ProveedorBeca"
frmBuscar.MostrarSol = True
frmBuscar.ShowDialog()
If Not (frmBuscar.LlaveReturn Is Nothing) Then
cbolidentProveedor.SelectedValue = frmBuscar.LlaveReturn
End If
End Sub

Sub AsignarDatosProveedor(ByVal Identificacion As String)
sqlda = New SqlDataAdapter("EXEC SeleccionarProveedorXIdentificacion " & Identificacion & "", sqlCn)
Try
Dim tabla As New DataTable
sqlda.Fill(tabla)
If (tabla.Rows.Count > 0) Then
txtIdProveedor.Text = tabla.Rows(0)("IdProveedor")
Else
MessageBox.Show("No existe proveedor con la identificación suministrada", "Error al buscar el proveedor",
MessageBoxButtons.OK, MessageBoxIcon.Error)
cbolidentProveedor.Text = ""
End If
Catch ex As Exception
MessageBox.Show(ex.Message)
End Try
End Sub

Private Sub cbolidentProveedor_Leave(ByVal sender As Object, ByVal e As System.EventArgs) Handles
cbolidentProveedor.Leave
If cbolidentProveedor.Text <> "" Then
AsignarDatosProveedor(cbolidentProveedor.SelectedValue)
End If
End Sub

Function ValidarDatos() As Boolean
If (txtIdEstudiante.Text = "") Or (txtIdEstudiante.Text Is Nothing) Then
ValidarDatos = False
MessageBox.Show("Debe de escoger un estudiante para la solicitud")
```



Sistema de control de Becas UACH

```
Exit Function
End If
If (txtIdProveedor.Text = "") Or (txtIdProveedor.Text Is Nothing) Then
ValidarDatos = False
MessageBox.Show("Debe de escoger un proveedor para la solicitud")
Exit Function
End If
If (txtAyoLectivo.Text = "") Or (txtAyoLectivo.Text Is Nothing) Then
ValidarDatos = False
MessageBox.Show("Debe escribir el año lectivo de la solicitud")
Exit Function
End If
If (cboAyoAcademico.Text = "") Or (cboAyoAcademico.Text Is Nothing) Then
ValidarDatos = False
MessageBox.Show("Se necesita saber a que año academico pertenece el estudiante")
Exit Function
End If
If (cboTipoBeca.Text = "") Or (cboTipoBeca.Text Is Nothing) Then
ValidarDatos = False
MessageBox.Show("Se necesita saber a que tipo de beca se esta solicitando")
Exit Function
End If
ValidarDatos = True
End Function

Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnAceptar.Click
If ValidarDatos() Then
Try
sqlCmd = New SqlCommand("InsertarSolicitud", sqlCn)
sqlCmd.CommandType = CommandType.StoredProcedure
sqlCmd.Parameters.Add("@NumeroSolicitud", txtNumeroSolicitud.Text)
sqlCmd.Parameters.Add("@FechaSolicitud", dtpFechaNac.Value)
sqlCmd.Parameters.Add("@AyoLectivo", txtAyoLectivo.Text)
sqlCmd.Parameters.Add("@AyoAcademico", cboAyoAcademico.Text)
sqlCmd.Parameters.Add("@TipoSolicitud", cboTipoSolicitud.SelectedValue)
sqlCmd.Parameters.Add("@TipoBeca", cboTipoBeca.SelectedValue)
sqlCmd.Parameters.Add("@IdProveedor", txtIdProveedor.Text)
sqlCmd.Parameters.Add("@IdEstudiante", txtIdEstudiante.Text)
sqlCmd.Parameters.Add("@Estado", 1) 'ESTADO PENDIENTE.
sqlCmd.Parameters.Add("@Observacion", txtObservacion.Text)
If (sqlCn.State = ConnectionState.Closed) Then
sqlCn.Open()
End If

Dim id As Integer
id = CInt(sqlCmd.ExecuteScalar())
GuardarCk(id)
If (MessageBox.Show("Se ha insertado satisfactoriamente el nuevo registro" & Chr(13) & "¿Desea insertar una nueva
solicitud?", "Proceso realizado satisfactoriamente", MessageBoxButtons.YesNo) = DialogResult.Yes) Then
LimpiarControles()
ObtenerConsecutivo()
Else
Me.Close()
End If
Catch ex As Exception
MessageBox.Show("Se produjo el siguiente error al insertar:" & Chr(13) & ex.Message)
Finally
sqlCn.Close()
End Try

End If
End Sub
```



Sistema de control de Becas UACH

```
Sub LimpiarControles()  
txtNumeroSolicitud.Text = ""  
dtpFechaNac.Value = DateTime.Now  
txtAyoLectivo.Text = ""  
cboAyoAcademico.SelectedIndex = 0  
cboTipoBeca.SelectedIndex = 0  
cbolidentProveedor.SelectedIndex = 0  
txtIdProveedor.Text = ""  
txtIdEstudiante.Text = ""  
txtObservacion.Text = ""  
txtCarnet.Text = ""  
txtNombreEstudiante.Text = ""  
txtCarrera.Text = ""  
txtModalidad.Text = ""  
CrearCK()  
End Sub
```

```
Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnCancelar.Click  
Close()  
End Sub
```

```
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click  
Dim frmEst As New frmEstudiante  
frmEst.Sol = True  
frmEst.ShowDialog()  
If (frmEst.Carnet <> "") Then  
txtCarnet.Text = frmEst.Carnet  
txtCarnet_Leave(Nothing, Nothing)  
End If  
End Sub  
End Class
```



Estudiante

Datos de los estudiantes

Carnet:	<input type="text" value="T1"/>	<input type="button" value="Renovar Beca"/>		
Primer nombre:	<input type="text" value="Ana"/>	Segundo nombre:	<input type="text" value="Patricia"/>	<input type="button" value="Cancelar Beca"/>
Primer apellido:	<input type="text" value="Escobar"/>	Segundo apellido:	<input type="text"/>	
Fecha de Nacimiento:	<input type="text" value="11/12/2008"/>	Edad:	<input type="text" value="0"/>	
Sexo:	<input type="text" value="F"/>	Teléfono:	<input type="text"/>	
Dirección domiciliar:	<input type="text"/>			
Modalidad:	<input type="text" value="Diario"/>	Carrera:	<input type="text" value="Turismo"/>	
Proveedor de la beca:	<input type="text" value="Antonio Jose Cuadra Rivera"/>	Año Académico:	<input type="text" value="4"/>	
Tipo de Beca:	<input type="text" value="25%"/>	Fecha otorgada:	<input type="text" value="10/12/2008"/>	

```
Private Sub frmEstudiante_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
If OpReg.EstadoOp = "" Then
EnlazarDatos()
OpReg.BloquearControles(Me)
Else
ConDb.AsignarConexion(sqlCnEst)
Obtenerdatos()
btnGuardarCambios.Visible = True
End If
ConfigurarInsertar()
ConfigurarActualizar()
ConfigurarEliminar()
ConfMovimiento()
If Sol Then
ConfigurarParaSol()
End If
End Sub
Private Sub EnlazarDatos()
Dim dstemp As New dsRegistroBecas
ConDb.AsignarConexion(sqlCnEst)
sqlIDA = New SqlDataAdapter(ConDb.ComandoSeleccionarEstado(NombreTabla), sqlCnEst)
dstemp.EnforceConstraints = False
Try
sqlCnEst.Open()
sqlIDA.SelectCommand.ExecuteNonQuery()
sqlIDA.Fill(dstemp, NombreTabla)
Catch ex As Exception
MessageBox.Show(ex.Message)
Finally
dstemp.EnforceConstraints = True
sqlCnEst.Close()
objDSEstudiantes.Clear()
objDSEstudiantes.Merge(dstemp)
Obtenerdatos()
End Try
End Sub
Private Sub dtpFechaNac_LostFocus(ByVal sender As Object, ByVal e As System.EventArgs) Handles
dtpFechaNac.LostFocus
txtEdad.Text = ConDb.CalcularEdad(Convert.ToDateTime(dtpFechaNac.Text))
End Sub
```



Sistema de control de Becas UACH

```
Private Sub frmEstudiante_Closing(ByVal sender As Object, ByVal e As System.ComponentModel.CancelEventArgs) Handles MyBase.Closing
Try
OpReg.VerCambioDatos("dsRegistroBecas", Me, objDSEstudiantes, NombreTabla, sqlCnEst, sqlIDA)
If (btnGuardarCambios.Enabled) And (OpReg.EstadoOp = "Agregando") Then
ID = Val(ConDb.UltimoIDEst())
Else
ID = objDSEstudiantes.Tables(NombreTabla).Rows(Me.BindingContext(objDSEstudiantes,
NombreTabla).Position).Item("IdEstudiante")
End If
Catch ex As Exception
End Try
End Sub
Private Sub btnGuardarCambios_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
btnGuardarCambios.Click
Try
If (txtIdCarrera.Text <> "") Then
If (txtIdModalidad.Text <> "") Then
clsControlRegistro.OperacionReg(Me, "dsRegistroBecas", Me.objDSEstudiantes, Me.NombreTabla, "Actualizar",
Me.sqlCnEst, Me.sqlIDA)
clsControlRegistro.BloquearControles(Me)
Carnet = txtCarnet.Text
Close()
Else
MessageBox.Show("Debe escoger una modalidad", "Faltan datos")
End If
Else
MessageBox.Show("Debe escoger una carrera", "Faltan datos")
End If
Catch ex As Exception
MessageBox.Show(ex.Message)
End Try
End Sub

Private Sub frmEstudiante_Resize(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.Resize
Me.WindowState = 0
End Sub
Private Sub cboModalidad_SelectedValueChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles
cboModalidad.SelectedValueChanged
If (cboModalidad.SelectedValue.GetType.ToString() <> "System.Data.DataRowView") And Not ObteniendoDatos Then
txtIdModalidad.Text = CStr(cboModalidad.SelectedValue)
End If
End Sub
Private Sub cboCarrera_SelectedValueChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles
cboCarrera.SelectedValueChanged
If (cboCarrera.SelectedValue.GetType.ToString() <> "System.Data.DataRowView") And Not ObteniendoDatos Then
txtIdCarrera.Text = CStr(cboCarrera.SelectedValue)
End If
End Sub
Function ObtenerIdBecaUltima() As Integer
Dim cmd As New SqlCommand("Exec SeleccionarBecaXEstudiante " + txtIdEstudiante.Text, sqlCnEst)
Try
If sqlCnEst.State = ConnectionState.Closed Then
sqlCnEst.Open()
End If

Dim id As Integer
ObtenerIdBecaUltima = cmd.ExecuteScalar 'Identificador de la beca
Catch ex As Exception
MessageBox.Show(ex.Message)
```



Sistema de control de Becas UACH

```
ObtenerIdBecaUltima = 0
Finally
sqlCnEst.Close()
End Try
End Function
Private Sub btnOtorgarBeca_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
btnOtorgarBeca.Click
Dim frmBeca As New frmBecas
If (ObtenerIdBecaUltima() > 0) Then
If (MessageBox.Show("El estudiante tiene asignada una beca" & Chr(13) & "¿Desea de todas formas otorgar una nueva?",
"Otorgar nueva beca", MessageBoxButtons.YesNo, MessageBoxIcon.Question) = DialogResult.Yes) Then
frmBeca.TipoOperacion = "Nuevo"
frmBeca.txtIdEstudiante.Text = Me.txtIdEstudiante.Text
frmBeca.pnEstudiante.Visible = False
frmBeca.ShowDialog()
End If
Else
frmBeca.TipoOperacion = "Nuevo"
frmBeca.txtIdEstudiante.Text = Me.txtIdEstudiante.Text
frmBeca.pnEstudiante.Visible = False
frmBeca.ShowDialog()
End If
ConfMovimiento()
End Sub
Private Sub btnCancelarBeca_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
btnCancelarBeca.Click
If (MessageBox.Show("¿Desea cancelar la beca activa?", "Cancelación de beca", MessageBoxButtons.YesNo,
MessageBoxIcon.Question) = DialogResult.Yes) Then
Dim frmCancelar As New frmCancelarBeca
Dim id As Integer
id = ObtenerIdBecaUltima()
If id > 0 Then
frmCancelar.IdBeca = id
frmCancelar.TipoObservacion = "Beca"
frmCancelar.ShowDialog()
Else
MessageBox.Show("El estudiante no tiene ninguna beca activa, Verifique los datos.")
End If
End If
ConfMovimiento()
End Sub
Function ObtenerAyoAcademico(ByVal id As Integer) As String
Dim cmd As New SqlCommand("Select AyoAcademico from vwListaBecasHdr Where IdBeca = " & id.ToString(), sqlCnEst)
Try
If (sqlCnEst.State = ConnectionState.Closed) Then
sqlCnEst.Open()
End If
ObtenerAyoAcademico = cmd.ExecuteScalar()
Catch ex As Exception
MessageBox.Show(ex.Message)
End Try
End Function

Private Sub btnRenovarBeca_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
btnRenovarBeca.Click
Dim frmRen As New frmRenovarBeca
If (ObtenerIdBecaUltima() > 0) Then
frmRen.IdBeca = ObtenerIdBecaUltima() 'Identificador de la beca
frmRen.Estudiante = txtNombre1.Text & " " & txtNombre2.Text & " " & txtApellido1.Text & " " & txtApellido2.Text & " " & Año
academico: " & ObtenerAyoAcademico(ObtenerIdBecaUltima()) 'Nombre completo del estudiante
```



Sistema de control de Becas UACH

```
frmRen.ShowDialog()
Else
MessageBox.Show("El estudiante no tiene ninguna beca activa, Verifique los datos.")
End If
ConfMovimiento()
End Sub
Public Sub ConfMovimiento()
If txtIdEstudiante.Text <> "" Then
If.dtpFechaNac.Value.ToString() <> "" Then
txtEdad.Text = ConDb.CalcularEdad(Convert.ToDateTime(dtpFechaNac.Text))
End If
End If
If (txtIdEstudiante.Text <> "") Then
Dim da As New SqlDataAdapter("Exec SeleccionarDetalleBecaXEstudiante " & txtIdEstudiante.Text, sqlCnEst)
Dim dtDetalleBeca As New DataTable
Try
da.Fill(dtDetalleBeca)
If dtDetalleBeca.Rows.Count > 0 Then
btnOtorgarBeca.Visible = False
btnRenovarBeca.Visible = True
btnCancelarBeca.Visible = True
For Each f As DataRow In dtDetalleBeca.Rows
txtProveedorBeca.Text = f("ProveedorBeca")
txtAyoAcademico.Text = f("AyoAcademico")
txtTipoBeca.Text = f("Tipo")
dtpFechaOtorgada.Value = f("FechaEmision")
Next
Else
btnOtorgarBeca.Visible = True
btnRenovarBeca.Visible = False
btnCancelarBeca.Visible = False
txtProveedorBeca.Text = ""
txtAyoAcademico.Text = ""
txtTipoBeca.Text = ""
dtpFechaOtorgada.Value = DateTime.Now
End If
Catch ex As Exception
MessageBox.Show(ex.Message)
End Try
End If
End Sub

Public Sub ConfigurarParaSol()
btnOtorgarBeca.Visible = False
btnCancelarBeca.Visible = False
btnRenovarBeca.Visible = False
pnDetalleBeca.Visible = False
btnGuardarCambios.Visible = True

btnCancelar.Visible = True
clsControlRegistro.OperacionReg(Me, "dsRegistroBecas", Me.objDSEstudiantes, Me.NombreTabla, "Nuevo", Me.sqlCnEst, Me.sqlDA)
clsControlRegistro.DesbloquearControles(Me)
End Sub

Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnCancelar.Click
Close()
End Sub
End Class
```




Sistema de control de Becas UACH



Proveedor de Becas

Proveedores de la beca

Identificación: 6568-565

Primer nombre: Edwin Segundo nombre: A

Primer apellido: Melendez Segundo apellido: A

Institución: UACH

Cargo: Vice Rector

Teléfono: 2466 Tipo: INTERNO

Dirección domiciliar: Chinandega.

```
Private Sub frmProveedorBeca_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
    If OpReg.EstadoOp = "" Then
        EnlazarDatos()
        OpReg.BloquearControles(Me)
    Else
        ConDb.AsignarConexion(sqlCnPBeca)
        Obtenerdatos()
        btnGuardarCambios.Visible = True
    End If
    ConfigurarInsertar()
    ConfigurarActualizar()
    ConfigurarEliminar()
End Sub
```

```
Private Sub EnlazarDatos()
    Dim dstemp As New dsRegistroBecas
    ConDb.AsignarConexion(sqlCnPBeca)
    sqlDA = New SqlDataAdapter(ConDb.ComandoSeleccionar("ProveedorBeca"), sqlCnPBeca)
    dstemp.EnforceConstraints = False
    Try
        sqlCnPBeca.Open()
        sqlDA.SelectCommand.ExecuteNonQuery()
        sqlDA.Fill(dstemp, "ProveedorBeca")
    Catch ex As Exception
        MessageBox.Show(ex.Message)
    Finally
        dstemp.EnforceConstraints = True
        sqlCnPBeca.Close()
        objDSProveedorBeca.Clear()
        objDSProveedorBeca.Merge(dstemp)
        Obtenerdatos()
    End Try
End Sub
```



Sistema de control de Becas UACH

```
Private Sub frmProveedorBeca_Closing(ByVal sender As Object, ByVal e As System.ComponentModel.CancelEventArgs) Handles MyBase.Closing
Try
OpReg.VerCambioDatos("dsRegistroBecas", Me, objDSPveedorBeca, "ProveedorBeca", sqlCnPbeca, sqlDA)
If (btnGuardarCambios.Enabled) And (OpReg.EstadoOp = "Agregando") Then
ID = Val(ConDb.UltimoIDEst())
Else
ID = objDSPveedorBeca.Tables("ProveedorBeca").Rows(Me.BindingContext(objDSPveedorBeca, "ProveedorBeca").Position).Item("IdProveedor")
End If
Catch ex As Exception
End Try
End Sub

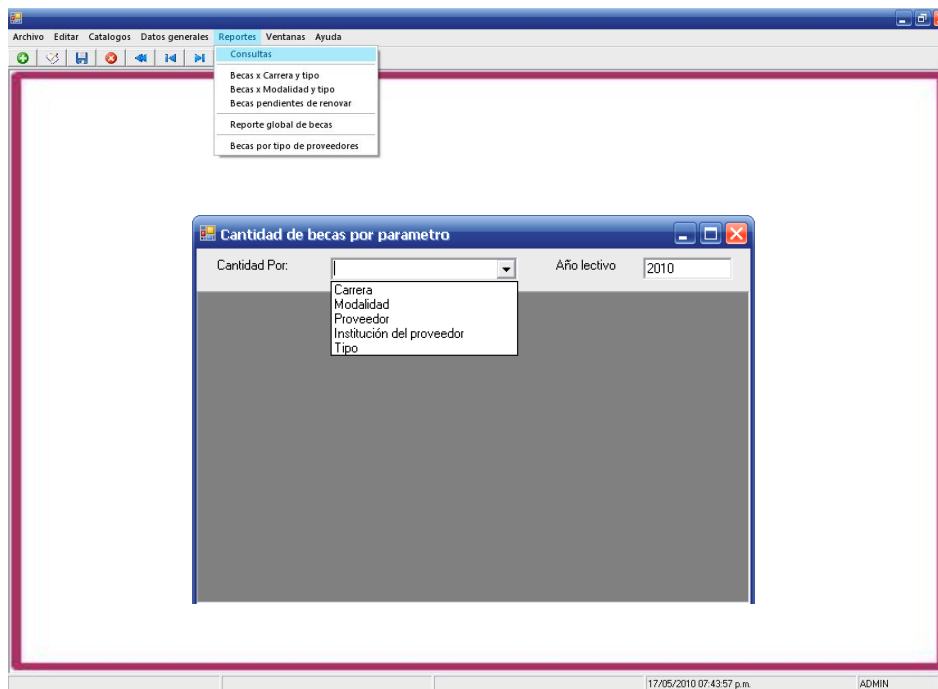
Private Sub btnGuardarCambios_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnGuardarCambios.Click
Close()
End Sub

Private Sub frmProveedorBeca_Resize(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.Resize
Me.WindowState = 0
End Sub

Private Sub cboldTipo_SelectedValueChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles cboldTipo.SelectedValueChanged
If (cboldTipo.SelectedValue.GetType.ToString() <> "System.Data.DataRowView") And Not ObteniendoDatos Then
txtIdTipo.Text = CStr(cboldTipo.SelectedValue)
End If
End Sub
End Class
```



Formulario de Reportes:



Consultas

```
Private Sub frmListaBecasXCantidad_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
    dgDatos.ReadOnly = True
    txtAyoLectivo.Text = Now.Date.Year
End Sub
```

```
Sub ConfigurarDatos(ByVal AyoLectivo As Integer, ByVal Procedimiento As String)
    Dim sqlda As New SqlDataAdapter("Exec " & Procedimiento & " " & AyoLectivo, cn)
    Try
        If (ds.Tables.Count > 0) Then
            If Not ds.Tables(Procedimiento) Is Nothing Then
                ds.Tables(Procedimiento).Clear()
            End If
        End If
        sqlda.Fill(ds, Procedimiento)
        If (ds.Tables(Procedimiento).Rows.Count > 0) Then
            dgDatos.DataSource = Nothing
            dgDatos.DataSource = ds.Tables(Procedimiento)
        End If
    Catch ex As Exception
        MessageBox.Show(ex.Message)
    End Try
End Sub
```

```
Sub ObtenerDetalle()
    Dim Title As String
```



Sistema de control de Becas UACH

```
Dim daDtl As New SqlDataAdapter(NombreSPDtl, cn)
daDtl.SelectCommand.CommandType = CommandType.StoredProcedure
Select Case NombreSPDtl
Case "dtlFiltrarCantidadXCarrera"
Prm1 = ds.Tables(TablaHdr).Rows(IndiceDbClick)("Carrera")
NombreRpt = "rptCantidadXCarrera"
daDtl.SelectCommand.Parameters.Add("@Carrera", ds.Tables(TablaHdr).Rows(IndiceDbClick)("Carrera"))
Title = "Detalle de becas por CARRERA"
Case "dtlFiltrarCantidadXModalidad"
Prm1 = ds.Tables(TablaHdr).Rows(IndiceDbClick)("Modalidad")
NombreRpt = "rptCantidadXModalidad"
daDtl.SelectCommand.Parameters.Add("@Modalidad", ds.Tables(TablaHdr).Rows(IndiceDbClick)("Modalidad"))
Title = "Detalle de becas por MODALIDAD"
Case "dtlFiltrarCantidadXProveedor"
Prm1 = ds.Tables(TablaHdr).Rows(IndiceDbClick)("Proveedor")
NombreRpt = "rptCantidadXProveedor"
daDtl.SelectCommand.Parameters.Add("@Proveedor", ds.Tables(TablaHdr).Rows(IndiceDbClick)("Proveedor"))
Title = "Detalle de becas por PROVEEDOR"
Case "dtlFiltrarCantidadXInstitucion"
Prm1 = ds.Tables(TablaHdr).Rows(IndiceDbClick)("Institucion")
NombreRpt = "rptCantidadXInstitucion"
daDtl.SelectCommand.Parameters.Add("@Institucion", ds.Tables(TablaHdr).Rows(IndiceDbClick)("Institucion"))
Title = "Detalle de becas por INSTITUCION"
Case "dtlFiltrarCantidadXTipo"
Prm1 = ""
NombreRpt = "rptCantidadXTipoBeca"
Title = "Detalle de becas por TIPO DE BECA"
End Select
daDtl.SelectCommand.Parameters.Add("@Tipo", ds.Tables(TablaHdr).Rows(IndiceDbClick)("Tipo"))
daDtl.SelectCommand.Parameters.Add("@AyoLectivo", txtAyoLectivo.Text)
Prm2 = ds.Tables(TablaHdr).Rows(IndiceDbClick)("Tipo")
Prm3 = txtAyoLectivo.Text
dtDtl = New DataTable
daDtl.Fill(dtDtl)
If (dtDtl.Rows.Count > 0) Then
Dim frmDtl As New frmMostrarDetalle
frmDtl.Titulo = Title
frmDtl.NombreRpt = NombreRpt
frmDtl.P1 = Prm1
frmDtl.P2 = Prm2
frmDtl.P3 = Prm3
frmDtl.dt = dtDtl
frmDtl.ShowDialog()
End If
End Sub

Private Sub cboCantidadX_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles cboCantidadX_SelectedIndexChanged
If txtAyoLectivo.Text <> "" Then
Select Case cboCantidadX.Text
Case "Carrera"
ConfigurarDatos(txtAyoLectivo.Text, "SeleccionarCantidadBecaXCarrera")
NombreSPDtl = "dtlFiltrarCantidadXCarrera"
TablaHdr = "SeleccionarCantidadBecaXCarrera"
Case "Modalidad"
ConfigurarDatos(txtAyoLectivo.Text, "SeleccionarCantidadBecaXModalidad")
NombreSPDtl = "dtlFiltrarCantidadXModalidad"
TablaHdr = "SeleccionarCantidadBecaXModalidad"
Case "Proveedor"
ConfigurarDatos(txtAyoLectivo.Text, "SeleccionarCantidadBecaXProveedor")
NombreSPDtl = "dtlFiltrarCantidadXProveedor"
```



Sistema de control de Becas UACH

```
TablaHdr = "SeleccionarCantidadBecaXProveedor"
Case "Institución del proveedor"
ConfigurarDatos(txtAyoLectivo.Text, "SeleccionarCantidadBecaXInstitucion")
NombreSPDtl = "dtlFiltrarCantidadXInstitucion"
TablaHdr = "SeleccionarCantidadBecaXInstitucion"
Case "Tipo"
ConfigurarDatos(txtAyoLectivo.Text, "SeleccionarCantidadBecaXTipo")
NombreSPDtl = "dtlFiltrarCantidadXTipo"

TablaHdr = "SeleccionarCantidadBecaXTipo"
End Select
Else
MessageBox.Show("Debe de escribir el año lectivo")
End If
End Sub

Private Sub txtAyoLectivo_Leave(ByVal sender As Object, ByVal e As System.EventArgs) Handles txtAyoLectivo.Leave
If (txtAyoLectivo.Text >= 2000 And txtAyoLectivo.Text <= Now.Date.Year + 1) And txtAyoLectivo.Text <> "" Then
cboCantidadX_SelectedIndexChanged(Nothing, Nothing)
Else
MessageBox.Show("Debe de ingresar un año válido")
End If
End Sub

Private Sub dgDatos_DoubleClick(ByVal sender As Object, ByVal e As System.EventArgs) Handles dgDatos.DoubleClick
If (dgDatos.CurrentRowIndex >= 0) Then
If (Not (dgDatos.Item(dgDatos.CurrentRowIndex, 0)) Is Nothing) Then
IndiceDbClick = dgDatos.CurrentRowIndex
ObtenerDetalle()
End If
End If
End Sub

Private Sub dgDatos_Navigate(ByVal sender As System.Object, ByVal ne As System.Windows.Forms.NavigateEventArgs)
Handles dgDatos.Navigate
End Sub
End Class
```



Becas x Carrera y Tipo

Filtros del reporte.

Año lectivo

Tipo

Carrera

```
Private Sub frmParametrosRpt_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
Select Case NombreRpt
Case "rptBecasXCarrera"
LlenarcborptBecasXCarrera()
Case "rptBecasXRenovar"
LlenarcborptBecasXCarrera()
Case "rptBecasXModalidad"
LlenarcborptBecasXModalidad()
lblCarreraModalidad.Text = "Modalidad"
Case "rptGlobal"
lblTipo.Visible = False
lblCarreraModalidad.Visible = False
cboCarreraModalidad.Visible = False
cboTipo.Visible = False
Case "rptEstudiantesXTipoBeca"
lblCarreraModalidad.Visible = False
cboCarreraModalidad.Visible = False
LlenarcborptTipo(2)
End Select
If (NombreRpt = "rptBecasXRenovar") Then
lblAyo.Text = "Año academico"
End If
End Sub
```

```
Sub LlenarcborptBecasXCarrera()
Dim ds As New DataSet
Dim da As New SqlDataAdapter("Select Tipo From Tipo", sqlcn)
da.Fill(ds, "Tipo")
cboTipo.DataSource = ds.Tables("Tipo")
cboTipo.DisplayMember = "Tipo"
da = New SqlDataAdapter("Select Descripcion From Carrera", sqlcn)
da.Fill(ds, "Carrera")
cboCarreraModalidad.DataSource = ds.Tables("Carrera")
cboCarreraModalidad.DisplayMember = "Descripcion"
End Sub
```

```
Sub LlenarcborptTipo(ByVal Cual As Integer)
If (Cual = 1) Then 'Becas por modalidad
Dim ds As New DataSet
Dim da As New SqlDataAdapter("Select Tipo From Tipo", sqlcn)
da.Fill(ds, "Tipo")
cboTipo.DataSource = ds.Tables("Tipo")
```



Sistema de control de Becas UACH

```
cboTipo.DisplayMember = "Tipo"
Elseif (Cual = 2) Then 'Estudiantes por tipo de becas.
Dim ds As New DataSet
Dim da As New SqlDataAdapter("Select Descripcion From TipoProveedor", sqlcn)
da.Fill(ds, "TipoProveedor")
cboTipo.DataSource = ds.Tables("TipoProveedor")
cboTipo.DisplayMember = "Descripcion"
End If
End Sub

Sub LlenarcborptBecasXModalidad()
Dim ds As New DataSet
Dim da As New SqlDataAdapter("Select Descripcion From Modalidad", sqlcn)
da.Fill(ds, "Modalidad")
cboCarreraModalidad.DataSource = ds.Tables("Modalidad")
cboCarreraModalidad.DisplayMember = "Descripcion"
LlenarcborptTipo(1)
End Sub

Private Sub cmdAceptar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles cmdAceptar.Click
Dim frm As New frmVisorReportes
frm.NombreRpt = NombreRpt
If NombreRpt = "rptBecasXRenovar" Then
frm.Titulo = "Becas pendiente de renovar por carrera"
frm.ParametrosRpt(0) = cboTipo.Text
frm.ParametrosRpt(1) = txtAyoLectivo.Text
frm.ParametrosRpt(2) = cboCarreraModalidad.Text
Elseif NombreRpt = "rptGlobal" Then
frm.Titulo = "Listado global de becas"
frm.ParametrosRpt(0) = txtAyoLectivo.Text
Elseif NombreRpt = "rptEstudiantesXTipoBeca" Then
frm.Titulo = "Becas otorgadas por proveedores: " & cboTipo.Text
frm.ParametrosRpt(0) = cboTipo.Text
frm.ParametrosRpt(1) = txtAyoLectivo.Text
Else
If NombreRpt = "rptBecasXCarrera" Then
frm.Titulo = "Becas por carrera"
Elseif NombreRpt = "rptBecasXModalidad" Then
frm.Titulo = "Becas por modalidad"
End If
frm.ParametrosRpt(0) = cboTipo.Text
frm.ParametrosRpt(1) = cboCarreraModalidad.Text
frm.ParametrosRpt(2) = txtAyoLectivo.Text
End If
Me.Hide()
frm.ShowDialog()
Me.Close()
End Sub

Private Sub cmdCancelar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
cmdCancelar.Click
Me.Close()
End Sub

Private Sub frmParametrosRpt_Resize(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.Resize
Me.WindowState = 0
End Sub
End Class
```



CONCLUSION

Al terminar con el desarrollo de este sistema, hemos determinado que logramos cumplir con los objetivos propuestos, los cuales tienen como finalidad la elaboración e implementación de un sistema computarizado que permita que el control de becas de estudiantes de la UACH se realice de manera rápida, precisa y eficaz, logrando disminuir el tiempo de operaciones de control de becas, mejorando la eficiencia en el manejo de la información, facilitando la generación de reportes estadísticos, ayudando a reducir espacio de almacenamiento de información, agilizando la disponibilidad de datos para el usuario responsable de su manejo y mejorando la calidad de servicios prestados a los estudiantes.

En el desarrollo de este trabajo hemos comprobado que el tiempo y los recursos son determinantes para el avance y finalización de un sistema.

Para nosotros fue una gran experiencia, porque nos ha permitido afianzar los conocimientos teóricos y prácticos que adquirimos en el transcurso de nuestra vida universitaria y además los nuevos conocimientos obtenidos durante la elaboración de este proyecto los que serán de gran ayuda en nuestra vida como profesional.

Esperamos que este documento ayude a futuras generaciones ya que incorporamos parte de la codificación que se escribió para el funcionamiento de esta aplicación.



RECOMENDACIONES

- ✚ Para evitar alteración de la información se recomienda tener en cuenta las medidas de seguridad, es decir establecer los permisos necesarios a los usuarios que tendrán acceso al sistema.
- ✚ Al borrar una carrera, modalidad, tipo de proveedor, tipo de beca, requisitos, tipo de solicitud, de la base de datos, se recomienda tener especial cuidado ya que una vez que haya eliminado un registro no podrá recuperarlo.
- ✚ Es recomendable realizar copia de seguridad de la base de datos de forma periódica, prestar mucha atención en el dispositivo de almacenamiento en el que se realizará.
- ✚ Al salir de la aplicación tratar de salir de forma correcta, esto evitará daños en el archivo de datos.



BIBLIOGRAFÍA

1. Blanco, Luis Miguel. Programacion en Visual. Net
2. Entrevista a Dirigente de la UACH

Referencia de Internet:

3. <http://www.elguille.info/NET/vs2005/Default.aspx>
4. <http://www.homeandlearn.co.uk/net/vbnet.html>
5. <http://www.scribd.com/doc/22739325/Manual-Visual-Basic-NET-SQL-Paso-a-Paso>
6. <http://msevents.microsoft.com/CUI/WebCastEventDetails.aspx?EventID=1032304899&EventCategory=3&culture=es-ES&CountryCode=ES>