UNIVERSIDAD NACIONAL AUTÓNOMA DE NICARAGUA

UNAN – LEÓN

Trabajo de Tesis de Maestría presentado para optar al título de Master en Computación con Énfasis en Gestión de la Información

Diseño de una aplicación con acceso a base de datos para llevar el control de las actividades económicas en los Cursos por Encuentro de la Facultad de Ciencias Económicas y Empresariales de la UNAN – León.

AUTOR:

Lic. Mario Alberto Talavera Flores

TUTOR:

M.Sc. Ernesto Espinoza

León, Julio 2008

Dedicatoria

Dedico mi trabajo de tesis a mis madre, Lic. Consuelo Flores Montalván, por ser la razón de mi existencia y fuente de inspiración y superación personal y profesional.

A mi esposa, Lic. Greetcheng Anielka Delgado Sirias, por compartir conmigo este largo camino de casi dos años y medio. Dos años y medio lleno de cansancio y de mucho esfuerzo, donde ella siempre estaba dándome apoyo moral.

A mi hijo, Diego Alberto Talavera Flores por ser la principal fuente de deseo de superación.

A mi tía, Lic. Martha Flores por demostrarme siempre un cariño de madre. Gracias.

Lic. Mario Talavera

Agradecimiento

Al creador, por ser mi guía espiritual, por darme fortaleza, prudencia y fe para llevar a cabo esta obra y así compartir mis conocimientos con la sociedad.

A mi tutor, M.Sc. Ernesto Espinoza, por ayudarme y darnos todo su tiempo para culminar con éxito este proyecto.

Lic. Mario Talavera

ÍNDICE PRINCIPAL

INTRODUCCION 1

ANTECEDENTES 2

DEFINICIÓN DEL PROBLEMA 3

JUSTIFICACIÓN Y ALCANCE DE LA INVESTIGACIÓN 4

OBJETIVOS 6

Objetivo General 6

Objetivos Específicos 6

RESULTADOS ESPERADOS 7

MARCO TEÓRICO 8

I. Ingeniería del Software 8

II. Programación Orienta a Objetos (POO) 11

III. Gestor de Base de Datos PostGreSQL 15

IV. Programación orientada a objetos en Visual Basic 18

V. Consideraciones generales del área de estudio 20

Elementos financieros y contables necesarios para un completo y eficiente control de las actividades económicas de los Cursos por Encuentros 26

Del proceso de pago 30

Esquema para una óptima cartera 32

Diagnóstico de la situación actual de los Cursos por Encuentro de la Facultad de Ciencias Económicas y Empresariales, UNAN – León 34

MARCO METODOLÓGICO 39

I. Descripción del Proyecto 39

II. Material y Método 39

III. Descripción del ciclo de vida 40

Modelo en Cascada 41
Actividades del Ciclo de vida en Cascada 41
Análisis 43
Especificación de Requisitos del Software 44
Arquitectura SQL de la Base de Datos. 73
CONCLUSIONES 82
RECOMENDACIONES 84
Anexos 85

INTRODUCCION

La Universidad Nacional Autónoma de Nicaragua, UNAN – León, es una institución de la educación superior con mucho prestigio dentro de las universidades públicas del país. Esto le ha permitido contar con una gran demanda en cada una de sus carreras ofertadas.

La Universidad Nacional Autónoma de Nicaragua tiene como misión el “contribuir a la transformación y desarrollo de la sociedad con vocación centro americanista, a través de la formación integral de las personas…”. Como parte de este compromiso con la sociedad, la Universidad Nacional Autónoma de Nicaragua – León, se ha preocupado por brindar la mejor educación superior, ofertando carreras de interés para todos los niveles y sectores sociales.

La Facultad de Ciencias Económicas es nueva dentro de las demás facultades de la UNAN – León. Pero esta no le ha impedido ofertar carreras de interés para la población lo que la ha llevado ha convertirse en una de las facultades con mayor población estudiantil dentro de la institución. Esta tiene como ofertas académicas carreras en modalidades Regulares y modalidad Curso por Encuentros (Sabatino).

Gracias a las herramientas actuales y a los avances en términos tecnológicos que la Universidad ha alcanzado en los últimos años, es que surge la necesidad de hacer uso de dichas herramientas y crear un proyecto en la Facultad de Ciencias Económicas y Empresariales que permita a los estudiantes de los Cursos por Encuentro y a sus autoridades proporcionar información actualizada, fiable y disponible en el menor tiempo posible sobre sus gestiones de pago de mensualidades, estados de cuenta y demás.

ANTECEDENTES

La Facultad de Ciencias Económicas y Empresariales anteriormente no contaba con ningún sistema automatizado que le permitiera llevar un mejor control de las actividades arancelarias y control de ingresos en la modalidad de Curso por Encuentros. Todos los procesos se llevaban a cabo de manera manual lo que incurría en pérdida de tiempo y dinero, ya que la cartera de pago generalmente no era recuperada por no llevar un control adecuado y ágil de la misma.

Además, los estudiantes generalmente a la hora de realizar sus respectivos pagos, se veían envueltos en procesos bastante engorrosos en los que se les debía solicitar primeramente contar con cada uno de sus recibos en los cuales se les pudiera constatar que mes le correspondía pagar y al mismo tiempo que la misma Facultad pudiera conocer el estado de cuenta de dicho estudiante. Esto incurría en largas filas de espera para poder lograr cancelar, abonar o solicitar algún servicio administrativo en el cual se pudiera reflejar el correcto cumplimiento de las obligaciones de pago de un estudiante.

Es por esta razón que la Facultad de Ciencias Económicas y Empresariales se ve en la necesidad de desarrollar y contar con un sistema que le permita estandarizar las actividades envueltas en la recuperación de la cartera y que facilite:

- A los estudiantes conocer de forma rápida, segura y confiable el estado de sus cuentas.
- A las autoridades y personal administrativo de la facultad, brindar un mejor servicio administrativo a los estudiantes que cursan cada una de las carreras ofertadas en esta modalidad. De igual manera, asegurar el correcto control de estos ingresos facilitando así la gestión administrativa y contable de los mismos y poder visualizar el movimiento monetario que se genera al poder contar con una cartera actualizada y confiable.

DEFINICIÓN DEL PROBLEMA

Debido a la demanda que estos Cursos por Encuentros han tenido, han sido una gran fuente de ingresos propios para las Facultad de Ciencias Económicas y Empresariales, ya que estos son autofinanciados por los mismos estudiantes. Es debido a esto que ellos se ven en el derecho de exigir no solo calidad en la enseñanza docente sino también calidad en los servicios administrativos que manejan estos cursos por encuentros.

Esta calidad en los servicios administrativos implica el deber de cada dependencia administrativa de ofrecer a cada estudiante un estado de cuenta que le permita reflejar cómo se encuentra su situación en lo referente a solvencia de cada una de sus mensualidades. Además, en la facultad existe una gran población estudiantil en esta modalidad, por lo que una situación muy común es ver la gran cantidad de estudiantes tratando de cumplir con sus deberes arancelarios y un débil sistema de atención en caja lo que da como resultado grandes filas y un número enorme de estudiantes inconformes y demandantes de un servicio más ágil y eficiente.

Es por esto que se plantea la siguiente pregunta: “¿Un sistema automatizado para el control de las actividades económicas de los Cursos por Encuentros de la Facultad de Ciencias Económicas y Empresariales ayudaría a obtener una mejora en los servicios administrativos brindados a los estudiantes y a un manejo más eficiente de la cartera generada por los mismos?”.

JUSTIFICACIÓN Y ALCANCE DE LA INVESTIGACIÓN

Para toda empresa, sea cual sea su sector, se le es necesaria una correcta administración de sus recursos. Esa administración adecuada de los recursos generalmente se ve apoyada en una cartera con un control muy eficiente. A la hora de gestionar una cartera hay que tener en cuenta muchos factores que van a influir en la toma de decisiones. Esto le permitirá a la empresa conocer en todo momento el capital con el que cuenta permitiéndole así poder circular ese dinero en futuras inversiones que mejoren su competitividad con otras empresas de su mismo sector.

Con relación a lo anterior, las empresas deben de competir en un mercado altamente demandante de servicios de calidad, que sean ágiles, eficientes y que principalmente se les permita a los futuros clientes o demandantes de estos servicios ahorrar tiempo y dinero.

La Facultad de Ciencias Económicas y Empresariales, UNAN – León, por ser una facultad joven y con una gran demanda académica se le es necesario proyectarse como una facultad a la vanguardia de los procesos tecnológicos que le permitirán automatizar sus actividades y ahorrarle tiempo y dinero que hasta el día de hoy se han estado mal gastando.

Esto le permitirá darle respuesta a la demanda de los estudiantes de mejor calidad en la atención a la hora de cumplir con sus obligaciones económicas que adquirieron a la hora de matricularse en algunas de las carreras ofertadas. De igual forma a la administración se le hará más fácil conocer en todo momento el movimiento de los ingresos que tienen en esta modalidad de estudio.

El resultado será un sistema robusto y flexible que dé solución a las problemáticas antes mencionadas. Este podrá ser de gran ayuda para los estudiantes, ya que se

les permitirá conocer y entregar en tiempo y forma sus estados de cuentas, calendarios de pagos, costos de los servicios y de nuevas ofertas; y para las autoridades y encargados de manejar estos ingresos, ya que se les permitirá agilizar sus tareas de control y administración de los mismos. Todo esto al final llevará ha convertir a la Facultad en una dependencia preocupada por darle solución a las necesidades y demandas de sus estudiantes y le dará más robustez a sus actividades administrativas.

OBJETIVOS

Objetivo General

Diseñar una aplicación con acceso a base de datos para el control de las actividades económicas de los Cursos por Encuentros de la Facultad de Ciencias Económicas y Empresariales de la UNAN – León.

Objetivos Específicos

- Diagnosticar la situación actual de la recuperación de cartera de cobro de los Cursos por Encuentros de la Facultad de Ciencias Económicas y Empresariales de la Universidad Nacional Autónoma de Nicaragua – León.

- Especificar los requerimientos necesarios que el sistema deberá tener para poder llevar a cabo cada una de las funciones específicas que ayuden a un mejor control y recuperación de la cartera de cobro.

- Diseñar la estructura adecuada de la base de datos del sistema para que pueda para que pueda albergar toda la información necesaria.

- Desarrollar una interfaz amigable para que pueda ser utilizada por cualquier usuario con la mínima experiencia en sistemas informáticos.

RESULTADOS ESPERADOS

- Contar con una estructura estándar que permita el Control de todas las actividades económicas en las que se ven envueltos los Cursos por Encuentros de la Facultad de Ciencias Económicas y Empresariales de la Universidad Nacional Autónoma de Nicaragua - León.

- Presentar una panorámica actualizada de la situación en la que actualmente se encuentran los Cursos por Encuentros de la Facultad de Ciencias Económicas y Empresariales de la Universidad Nacional Autónoma de Nicaragua – León.

- Describir los elementos financieros y contables que envuelven las actividades económicas de estos Cursos por Encuentros y de esta manera comprender que elementos son indispensables para que el sistema sea óptimo.

- Caracterizar los diferentes tipos de usuarios que se ven envueltos en el desarrollo de las actividades económicas de los Cursos por Encuentros, qué funciones tienen en el proceso manual y cuál será su rol dentro del sistema.

- Tomar toda la información obtenida y determinar un sistema de control de las actividades económicas de los Cursos por Encuentros de la Facultad de Ciencias Económicas y Empresariales de la UNAN – León, mediante el uso de herramientas tecnológicas de diseño de software y bases de datos.

MARCO TEÓRICO

En todo proceso de investigación, se debe de pasar por una fase que es la antesala al desarrollo del proyecto. Esta fase consiste en la recolección de toda la información teórica con la que se debe de contar para poder desarrollar y culminar la investigación. Esta parte se resume en una serie de conceptos sobre los puntos específicos que se abarcarán en la investigación y que necesitan esclarecerse para poder comprender a plenitud la magnitud y el alcance de la investigación.

Para comprender mejor la elaboración de este proyecto debemos conocer ciertos criterios importantes aplicados dentro del desarrollo de la investigación, los cuales se describen a continuación.

I. Ingeniería del Software

a. Concepto de DBMS

DBMS (**D**ata **B**ase **M**anagement **S**ystem): Sistemas generadores de Bancos de Datos mejor conocidos como Bases de Datos. Con programas que se ocupan de acceder y actualizar las bases de datos, como Microsoft Access, SQL Server, Oracle, MySql, PostgreSql. Actúan como una interfaz entre el programa de aplicación y la Base de Datos. Por ejemplo, cuando un programa debe de actualizar una información en la Base de Datos, no lo hace directamente, sino que le pide al **DBMS** hacerlo.

Para llevar a cabo el desarrollo de nuestra aplicación web utilizamos el “Ciclo de Vida Básico” o “Modelo en Cascada”.

b. Etapa del Desarrollo del Software

Este modelo descompone el desarrollo en varias etapas:

Ingeniería y modelado de sistemas / información: El software siempre forma parte de un sistema más grande, el trabajo comienza estableciendo requisitos de todos los elementos del sistema y asignando al software algún grupo de estos requisitos.

Análisis de los requisitos del software: permite especificar la función y el rendimiento del software, indica la interfaz del software con otros elementos del sistema y establece las restricciones que debe de cumplir el software, proporciona modelos que pueden traducirse en el diseño de datos, arquitectónico, de interfaz y procedimental. Finalmente proporcionan al diseñador y al cliente los medios para valorar la calidad una vez que se ha construido el software.

Diseño: Se centra en cuatro atributos distintos de un programa: estructura de datos, arquitectura del software, representaciones de interfaz y detalle procedimental.

Generación de código: Realiza la tarea de traducir el diseño en una forma legible por la máquina.

Pruebas: Se centra en los procesos lógicos internos del software asegurando que todas las sentencias se han comprobado, la realización de las pruebas es para la detección de errores y sentirse seguro de que la entrada definida produzca resultados reales de acuerdo con los resultados requeridos.

Mantenimiento: El software puede sufrir cambios una vez entregado al cliente, por que se pueden encontrar errores, o por que el software debe adaptarse a los cambios de su entorno externo o por que el cliente desea hacerle cambios.

c. Fases del Diseño

Diseño de Datos: La actividad principal de diseño de datos es seleccionar representaciones lógicas de objetos de datos identificados durante la fase de definición de requisitos.

Diseño Arquitectónico: Su objetivo primario es desarrollar una estructura de programa modular y representar las relaciones de control entre otros módulos. Combina la estructura del programa y las estructuras de datos definiendo interfaces que permiten el flujo de datos a través del programa.

Diseño de Interfaz: El diseño de interfaz se centra en tres áreas importantes: el diseño de interfaz entre los módulos software, diseño de interfaz entre el software y otros productos y consumidores no humanos de información y el diseño de la interfaz entre el hombre y la computadora.

Diseño procedimental: Debe de especificar los detalles procedimentales sin ambigüedades, por lo que no se recomienda utilizar solamente el lenguaje natural, sino que se debe de utilizar un modo más restringido de representar el detalle procedimental ya sea con algoritmos o diagramas de flujos.

Diagrama de flujo de datos (DFD): Es una técnica que representa el flujo de la información y las transformaciones que se aplican a los datos al moverse desde la entrada hasta la salida. Se puede utilizar el DFD para representar un Sistema o Software a cualquier nivel de abstracción.

Diccionario de Datos: Es un listado organizado de todos los elementos de datos que son necesarios para el sistema, con definiciones precisas y rigurosas que permiten que el usuario y el analista del sistema tengan una misma comprensión de las entradas, salidas, de los componentes, de los almacenes y de los cálculos intermedios.

II. Programación Orienta a Objetos (POO)

La programación orientada a objetos es un *paradigma de programación*[1] que usa objetos y sus interacciones para diseñar aplicaciones y programas de computadora. Este consta de varias técnicas entre las cuales va incluida: la herencia, modularidad, polimorfismo, y encapsulamiento.

En la programación orientada objetos existe lo que conocemos como objetos, los cuales son entidades que combinan *estado, comportamiento* e identidad.

La programación orientada a objetos expresa un programa como un conjunto de estos objetos, que colaboran entre ellos para realizar tareas. Esto permite hacer los programas y módulos más fáciles de escribir, mantener y neutralizar.

De esta forma, un objeto contiene toda la información que permite definirlo e identificarlo frente a otros objetos pertenecientes a otras clases e incluso frente a objetos de una misma clase, al poder tener valores bien diferenciados en sus *atributos*. A su vez, los objetos disponen de mecanismos de interacción llamados *métodos*[2] que favorecen la comunicación entre ellos. Esta comunicación favorece a su vez el cambio de estado en los propios objetos. Esta característica llega a tratarlos como unidades indivisibles, en las que no se separan ni deben separarse el estado y el comportamiento.

a. Conceptos fundamentales de la Programación Orientada a Objetos

[1] Representa un enfoque particular o filosofía para la construcción de software. No existe uno mejor que otro sino que cada uno tiene ventajas y desventajas. También hay situaciones en las que un paradigma resulta más apropiado que el otro.

[2] El término se refiere a las porciones de código asociadas exclusivamente con una clase (se les denomina entonces **métodos de clase** o **métodos estáticos**) o con un objeto (en este caso **métodos de instancia**)

La programación orientada a objetos es una nueva forma de programar que trata de encontrar una solución a estos problemas. Introduce nuevos conceptos, que superan y amplían conceptos antiguos ya conocidos. Entre ellos destacan los siguientes:

- Clases: definiciones de las propiedades y comportamiento de un tipo de objeto concreto.
- Objeto: Entidad provista de un conjunto de propiedades o atributos (datos) y de comportamiento o funcionalidad (métodos). Se corresponde con los objetos reales del mundo que nos rodea, o a objetos internos del sistema (del programa). Es una instancia a una clase.
- Método: algoritmo asociado a un objeto (o a una clase de objetos), cuya ejecución se desencadena tras la recepción de un “mensaje”.
- Evento: Un suceso en el sistema (tal como la interacción de un usuario con la máquina, o un mensaje enviado por un objeto). El sistema maneja el evento enviando el mensaje adecuado al objeto pertinente. También se puede definir como evento, a la reacción que puede desencadenar un objeto, es decir la acción que genera.
- Mensaje: Una comunicación dirigida a un objeto, que le ordena que ejecute uno de sus métodos con ciertos parámetros asociados al evento que lo generó.
- Propiedad o atributo: Contenedor de un tipo de datos asociados a un objeto (o a una clase de objetos), que hace los datos visibles desde afuera del objeto y esto se define como sus características predeterminadas, y cuyo valor puede ser alterado por la ejecución de algún método.
- Estado interno: Es una variable que se declara privada, que puede ser únicamente accedida y alterada por un método del objeto, y que se utiliza para indicar distintas situaciones posibles para el objeto (o clases de objetos), que hace los datos visibles desde afuera del objeto y esto se define como sus características predeterminadas, y cuyo valor puede ser alterado por la ejecución de algún método.

- Componentes de un objeto: atributos, identidad, relaciones y métodos.
- Representación de un objeto: un objeto se representa por medio de una tabla o entidad que esté compuesta por sus atributos y funciones correspondientes.
-

b. Características principales

Existe un cierto desacuerdo sobre cuales son exactamente las características de un método de programación o lenguaje le definen como "orientado a objetos", pero hay un consenso general en que las características siguientes son las más importantes:

- Abstracción: Cada objeto en el sistema sirve como modelo de un "agente" abstracto que puede realizar trabajo, informar y cambiar su estado, y "comunicarse" con otros objetos en el sistema sin revelar cómo se implementan estas características. Los procesos, las funciones o los métodos pueden también ser abstraídos y cuando lo están, una variedad de técnicas son requeridas para ampliar una abstracción.
- Encapsulamiento: Significa reunir a todos los elementos que pueden considerarse pertenecientes a una misma entidad, al mismo nivel de abstracción. Esto permite aumentar la cohesión[3] de los componentes del sistema.
- Principio de ocultación: Cada objeto está aislado del exterior, es un módulo natural, y cada tipo de objeto expone una interfaz a otros objetos que específica cómo pueden interactuar con los objetos de una clase. El aislamiento protege a las propiedades de un objeto contra su modificación por quien no tenga derecho a acceder a ellas, solamente los propios métodos internos del objeto pueden acceder a su estado. Esto asegura que otros objetos no pueden cambiar el estado interno de un objeto de maneras inesperadas, limitando el grado de abstracción, eliminando efectos

[3] Nos dice que la información que almacena una clase debe de ser coherente y está en la mayor medida de lo posible relacionada con la clase.

secundarios e interacciones inesperadas. Algunos lenguajes relajan esto, permitiendo un acceso directo a los datos internos del objeto de una manera controlada y limitando el grado de abstracción.

- Polimorfismo: Comportamientos diferentes asociados a objetos distintos, pueden compartir el mismo nombre, al llamarlos por ese nombre se utilizará el comportamiento correspondiente al objeto que esté usando. O dicho de otro modo, las referencias y las colecciones de objetos pueden contener objetos de diferentes tipos, y la innovación de un comportamiento en una referencia producirá el comportamiento correcto para el tipo real del objeto referenciado. Cuando esto ocurre en "tiempo de ejecución"[4] , esta última característica se llama *asignación tardía* o *asignación dinámica*. Algunos lenguajes proporcionan medios más estáticos (en "tiempo de compilación"[5]) de polimorfismo, tales como las plantillas y la *sobrecarga de operadores*[6] de C++.
- Herencia: Las clases no están aisladas, sino que se relacionan entre sí, formando una jerarquía de clasificación. Los objetos heredan las propiedades y el comportamiento de todas las clases a las que pertenecen. La herencia organiza y facilita el polimorfismo y el encapsulamiento permitiendo a los objetos ser definidos y creados como tipos especializados de objetos preexistentes. Estos pueden compartir y extender su comportamiento sin tener que volver a implementarlo. Esto suele hacerse habitualmente agrupando los objetos en clases y estas en árboles que reflejan un comportamiento en común. Cuando un objeto hereda de más de una clase se dice que hay *herencia múltiple*.

[4] Se denomina tiempo de ejecución (*Runtime*) al intervalo de tiempo en el que un programa de computadora se ejecuta en un sistema operativo.
[5] Se denomina tiempo de compilación (*compile-time*) al intervalo de tiempo en el que un compilador compila código escrito en lenguaje de programación a una forma de código ejecutable por una máquina.
[6] La sobrecarga de operadores es utilizada para dar más de una implementación a un operador.

III. Gestor de Base de Datos PostGreSQL

PostGreSQL es un servidor de base de datos relacional orientada a objetos de *software libre*, liberado bajo la licencia BSD.

Como muchos otros proyectos *open source*, el desarrollo de PostGreSQL no es manejado por una sola compañía sino que es dirigido por una comunidad de desarrolladores y organizaciones comerciales las cuales trabajan en su desarrollo. Dicha comunidad es denominada PGDG (PostgreSQL Global Development Group).

Para ser más entendible el hecho de que PostgreSQL es *software libre*, a continuación se dará una breve explicación sobre este calificativo, luego se mencionará algunas de las características más importantes de este Gestor de Base de Datos.

a. Software Libre

Software Libre es la denominación del software que brinda libertad a los usuarios sobre su producto adquirido y por tanto, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente. Según la *Free Software Foundation*, el software libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software; de modo más preciso, se refiere a cuatro libertades de los usuarios del software: la libertad de usar el programa, con cualquier propósito; de estudiar el funcionamiento del programa, y adaptarlo a las necesidades; de distribuir copias, con lo que puede ayudar a otros; de mejorar el

programa y hacer públicas las mejoras, de modo que toda la comunidad se beneficie.

El software libre suele estar disponible gratuitamente, o a precio del coste de distribución a través de otros medios; sin embargo no es obligatorio que sea así, por ende no hay que asociar software libre a "software gratuito" (denominado usualmente como *freeware*), ya que, conservando su carácter de libre, puede ser distribuido comercialmente ("software comercial"). Análogamente, el "software gratis" o "gratuito" incluye en algunas ocasiones el código fuente; no obstante este tipo de software no es libre en el mismo sentido que el *software libre*, a menos que garanticen los derechos de modificación y redistribución de dichas versiones modificadas del programa.

b. Características de PostGreSQL

PostGreSQL ha tenido una larga evolución, comenzando con el proyecto *Ingres* en la *Universidad de Berkeley*. Con el pasar del tiempo muchos desarrolladores entusiastas de los motores de base de datos se unieron al proyecto y entre todos comenzaron a incorporar muchas características al motor. Algunas de sus características principales tenemos:

- **Alta concurrencia**

Mediante un sistema denominado MVCC (Acceso concurrente multiversión, por sus siglas en inglés) PostGreSQL permite que mientras un proceso escribe una tabla, otros accedan a la misma tabla sin necesidad de bloqueos

- **Amplia variedad de tipos nativos**

  PostgreSQL provee nativamente soporte para:

  - Números de *precisión arbitraria*[7]
  - Texto de largo ilimitado.
  - Figuras geométricas (con una variedad de funciones asociadas)
  - Direcciones IP (IPv4 e IPv6)
  - Bloques de direcciones estilo CIDR
  - Direcciones MAC[8]
  - Arrays[9]

[7] Es un método que permite la representación, en un programa informático, de números ya sean enteros o racionales con tantos dígitos de precisión como cuanto sea deseado y además posibilita la realización de operaciones aritméticas sobre dichos números.
[8] Es un identificador de 48 bits (6 bytes) que corresponde de forma única a una tarjeta o interfaz de red.
[9] Conjunto de agrupación de variables del mismo tipo cuyo acceso se realiza por índices.

IV. Programación orientada a objetos en Visual Basic

a. Generalidades del sistema

El sistema fue desarrollado por Alan Cooper para Microsoft. El lenguaje de programación es un dialecto de BASIC, con importantes añadidos. Visual Basic constituye un IDE[10] que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un depurador, un compilador, y un constructor de interfaz gráfica o GUI.

Es un lenguaje de fácil aprendizaje centrado en un motor de formularios que facilita el rápido desarrollo de las aplicaciones gráficas.

Cabe señalar que se ha agregado una implementación limitada de la programación orientada a objetos, aunque si admite el polimorfismo mediante el uso de interfaces, no admite la herencia. No requiere de manejo de punteros y posee un manejo sencillo de cadena de caracteres. Posee varias bibliotecas para el manejo de bases de datos, pudiendo conectar con cualquier base de datos a través de ODBC, entre ellas PostgreSQL, a través de ADO.

- **Ventajas**

Como todo lenguaje de programación, Visual Basic consta de una serie de ventajas que lo hacen una herramienta de programación aceptable y la razón por la cual se decidió usar esta herramienta para diseñar la interfaz de la aplicación.

[10] Integrated Development Enviroment

1. Es un lenguaje RAD[11]
2. Posee una curva de aprendizaje muy rápida
3. Integra el diseño e implementación de formularios Windows

b. Características generales de la programación en Visual Studio 6.0

- La posibilidad de acceder a datos de la base de datos, aplicaciones cliente/servidor y aplicaciones escalables.
- Tecnología ActiveX.
- Controles ActiveX.
- Programación en Internet, servidores web, DHTML, etc.

[11] Rapid Application Development

V. Consideraciones generales del área de estudio

a. ¿Qué son los Cursos por Encuentros?

Desde el punto de vista de la metodología de estudio, los cursos por encuentros son una modalidad de estudio en la cuales los estudiante se hacen presente un día a la semana para recibir clases. En la UNAN – León estos Cursos por Encuentro generalmente son llamados Cursos Sabatinos, pero realmente, un curso por encuentro puede darse cualquier día de la semana. El nombre más acertado para estos cursos por encuentro es *Curso para Trabajadores*.

b. ¿A quiénes van dirigidos?

Los Cursos para Trabajadores, como su nombre lo dice, van dirigidos a personas profesionales que generalmente tienen ya una carrera y desean aumentar sus conocimientos y mejorar su calidad profesional.

Aunque esto debería ser así, y está pensado que debe ser así, la realidad es otra. Actualmente en los cursos por encuentro ofertados por las diferentes Facultades de la UNAN – León se puede observar una tendencia marcada de que los estudiantes de estas modalidades la mayoría acaban de salir de los colegios públicos o privados y que por algún motivo no pudieron clasificar en carreras de modalidad regular y decidieron tomar una carrera de modalidad sabatina. Esta es una de las situaciones más comúnmente vista.

De cierto modo esto representa una desventaja para este tipo de estudiante porque la forma en que los cursos por Encuentros están

estructurados no va de acorde al nivel de conocimiento y a la experiencia profesional que estos estudiantes llevan al salir de sus centros de estudio secundario. Actualmente se están estudiando las políticas de ingreso para tratar de controlar este tipo de situación sin perjudicar la calidad de la enseñanza.

c. *¿Cómo funcionan los Cursos por Encuentro?*

Los Cursos por Encuentro tienen una metodología de estudio muy diferente a los cursos Regulares. Como se dijo anteriormente, esto se debe a que son cursos orientados específicamente para trabajadores que ya constan con una experiencia profesional previa. Vamos a explicar un poco la forma de trabajo de los Cursos por Encuentro de la Facultad de Ciencias Económicas y Empresariales.

Estos están divididos en Trimestres, en cada trimestre el estudiante se debe de presentar un día a la semana, que en este caso es el día sábado.

El primer día de clase el estudiante se presenta a su respectiva aula de clase, donde el profesor le deberá de indicar de que trata el curso, como va estar compuesto el contenido temático de la asignatura que se va a impartir, que ponderación tendrán las distintas evaluaciones que se realicen, entre otras cosas. En resumen, el docente el primer día de clase está en la obligación de darle al estudiante una panorámica detallada de todo lo que verá durante el trimestre que dure dicha asignatura. Luego de dar todas estas indicaciones al estudiante, el profesor comienza impartiendo el primer tema el cual ha sido desarrollado de acuerdo a la forma de trabajo de los cursos por encuentros. Seguidamente, el profesor debe de indicar la tarea que debe de realizar el estudiante si es que hay alguna y además debe de

indicarle que es lo que se debe de leer el estudiante para el próximo encuentro. Cabe señalar que la característica principal de estos cursos es el auto aprendizaje. El estudiante está en la obligación de leer el material que se va a impartir en el encuentro siguiente y el profesor en la obligación de entregar dicho material con anterioridad al siguiente encuentro.

En el segundo día de clase, el estudiante se presenta a su respectiva aula de clase donde el profesor deberá de aclarar las dudas que le hayan suscitado al estudiante producto de la lectura realizada antes del encuentro. Luego el docente explica el contenido de la clase que se verá en el encuentro siguiente, entrega las tareas correspondientes que según como lo decida el profesor deben de ser entregadas en encuentros posteriores.

Esto se repite de manera consecutiva hasta terminar el trimestre.

d. Relación entre el problema de estudio y las tecnologías informáticas a utilizar

En esta sección trataremos de dar las consideraciones necesarias que se tomaron como referencia para desarrollar la aplicación usando Visual Basic 6.0 y PostgreSQL como tecnologías de desarrollo y gestión de datos respectivamente.

En apartados anteriores se mostraron las ventajas y desventajas de Visual Basic 6.0 y si bien es cierto las ventajas que tiene el desarrollo de aplicaciones con Visual Basic 6.0 se ven sufragadas debido a la gran cantidad de desventajas que este lenguaje presenta, es necesario hacer las siguientes consideraciones.

Como se dijo anteriormente, Visual Basic es un lenguaje de fácil aprendizaje pensado tanto para programadores principiantes como para aquellos con cierta experiencia, guiado por eventos, y centrado en un motor de formularios que facilita el rápido desarrollo de aplicaciones gráficas.

Visual Basic es utilizado principalmente para aplicaciones de gestión de empresas, debido a la rapidez con la que puede hacerse un programa que utilice una base de datos sencilla.

La Facultad de Ciencias Económicas y Empresariales, es la facultad más joven que existe en la Universidad Nacional Autónoma de Nicaragua – León. Con apenas dos años de haber sido nombrada Facultad, esta alberga la cantidad de 2780 estudiantes. Ver tabla adjunta.

Carrera / Facultad	Estatal	Privado	Total
GESTION EMPRESA TURIST.	51	130	181
ECONOMIA	76	167	243
ADMINISTRACION DE EMP.	69	194	263
CONTADURIA PUB. Y FINANZA	110	149	259
MERCADOTECNIA	69	141	210
TURISMO (SABATINO)	41	55	96
ADMON. EMPRESAS (SABATINO	195	292	487
ECONOMIA (SABATINA)	81	73	154
CONTADURIA (SABATINA)	287	486	773

MERCADOTECNIA (SABATINO)	39	75	114
CIENCIAS ECONOMICAS	1018	1762	2780

Esto la convierte en la Facultad con la mayor población estudiantil dentro de esta casa de estudio. Como se puede observar en la tabla anterior la cantidad de alumnos que estudian en la modalidad de Cursos por Encuentros, también conocida como Cursos por Sabatinos (esto debido a que las clases de los Cursos por Encuentros en la UNAN – León son impartidos los días Sábado), representan el sustento económico de dicha Facultad. Para ver más detalladamente esto, observemos las siguientes tablas.

Modalidad Curso por Encuentros

Carrera / Facultad	Estatal	Privado	Total
TURISMO (SABATINO)	41	55	96
ADMON. EMPRESAS (SABATINO	195	292	487
ECONOMIA (SABATINA)	81	73	154
CONTADURIA (SABATINA)	287	486	773
MERCADOTECNIA (SABATINO)	39	75	114
		Total	1624

Modalidad Cursos Regulares			
Carrera / Facultad	**Estatal**	**Privado**	**Total**
GESTION EMPRESA TURIST.	51	130	181
ECONOMIA	76	167	243
ADMINISTRACION DE EMP.	69	194	263
CONTADURIA PUB. Y FINANZA	110	149	259
MERCADOTECNIA	69	141	210
		Total	1156

La cantidad de estudiantes de los cursos por encuentros prácticamente sobrepasa a la cantidad de estudiantes de los cursos regulares.

Como se ha hecho mención anteriormente, los cursos por encuentro son autofinanciados y representan la mayor fuente de ingresos propios de una Facultad. Es por esto que urgía un sistema que ayudara al control administrativo de cada uno de los ingresos generados por los estudiantes de dichos cursos.

Es por esto que se decidió tomar una herramienta de desarrollo fácil de usar y que prestaras las condiciones para poder controlar y administrar el flujo de información que se genera en dichos cursos por encuentros.

Elementos financieros y contables necesarios para un completo y eficiente control de las actividades económicas de los Cursos por Encuentros

Actualmente, en las empresas existe la necesidad de una armonización financiera y contable que permita la obtención de los objetivos de la misma y que permite ofrecer una mejor calidad en los servicios que se brindan.

Por armonización podemos entender el proceso de adaptación de las normas y políticas de la empresa a la forma de trabajo de una aplicación informática.

La necesidad de hacer valer las normas de control interno de estos Cursos por Encuentro han hecho llegar a las autoridades superiores de la Facultad de Ciencias Económicas y Empresariales a la necesidad de contar con herramientas tecnológicas que ayuden al tratamiento y procesamiento de la información, agilizando de esta manera el trabajo que actualmente se realiza de forma manual y permitiendo que los estudiantes de dichos cursos por encuentros puedan recibir toda la información necesaria sobre sus estados de cuenta y demás actividades económicas que en dicha modalidades de estudio realizan.

Pero, como es conocimiento de todo diseñador de aplicaciones, es necesario indagar en los distintos conceptos que de manera obligatoria llevarían a cumplir a cabalidad con las tareas que una buena y completa herramienta de procesamiento de información contable debe de realizar.

Las Normas Técnicas de Control Interno establecen que las Tecnologías de la Información que puedan ayudar a la automatización de las actividades contables y administrativas de cualquier empresa deben de cumplir con controles de validación de los datos al ser ingresados para procesamiento, por lo que en nuestro caso se establecieron medidas de control tales como:

1. Controles de edición y validación.

2. Doble digitación de campos críticos.

Además se especifica que todas aquellas transacciones que vayan a ser anuladas o rechazadas, ya sea que no cumplan con las características establecidas para su ingreso a la base de datos del sistema serán devueltas al usuario o incluidas en un archivo de transacciones anuladas o rechazadas para su posterior corrección en caso de ser posible. Esto aplica a la hora de ingresar un recibo que por cierta razón ha sido anulado, el sistema permite conocer y llevar un control del mismo.

Con respecto al procesamiento de los datos, las NTCI especifican que deben de establecerse controles para asegurar que los datos procesados y la información obtenida sean consistentes, completos y correspondan al período correcto.

Estos controles podrían ser los siguientes:

1. Formulario pre numerado y rutinas de control de secuencia.

   Los recibos que actualmente se manejan en la Facultad y que se entregan como soporte de que la transacción fue realizada, se encuentran pre numerados. El sistema debió adecuarse a esta característica especial. Los recibos son solicitados a la imprenta cuando se detecta que están por agotarse. Estos vienen en una presentación de pequeñas resmas de 50 recibos cada una, donde cada recibo lleva incluido las dos copias necesarias para cumplir con el control interno. Cada recibo consta de una serie y un número de recibo. Las series que son manejadas corresponden a las siguientes:

   - Serie A – Recibos para pagos en córdobas.
   - Serie B – Recibos para pagos en dólares.

   El sistema fue diseñado para que de manera manual se alimentara con la serie de recibos y el número de recibo con la que inicia la resma de 50

recibos. Automáticamente el sistema calcula el número de recibo final a partir de sumar al número de recibo inicial los 49 recibos que le proceden.
De esta manera se crea una rutina que permite controlar la secuencia de uso de cada recibo, permitiendo ahorrar tiempo, reducir las probabilidades de error a la hora de generar un recibo y permitir una mayor automatización en esta actividad.

2. Totalización de valores críticos antes y después del procesamiento.

   Según la forma de trabajo de la cajera a la hora de generar un recibo de forma manual, un solo recibo puede contener múltiples conceptos, los cuales son plasmados uno a uno, luego se suma el total de cada uno de los conceptos y se obtiene el total del monto a pagar en el recibo.

   El sistema emula esta misma característica y lo que hace es tomar el valor de cada concepto, los cuales previamente han sido ingresados al sistema esto con el objetivo de manejar un Catálogo de Conceptos según la normas contabilidad, y cada uno de los valores de los conceptos son totalizados antes del procesamiento de los datos y revisados nuevamente a la hora de registrar el recibo en la base de datos del sistema.

3. Validez de los datos generados automáticamente.

   Todos los datos que son generados automáticamente, tales como los números de recibo, son cuidadosamente validados ya que es el usuario quien primeramente ingresa la serie del recibo que se está usando y el número de recibo inicial y el número de recibo final. A la hora de que el usuario inicia el proceso de generación del recibo, el sistema consulta la serie de recibo que se encuentra en uso y luego revisa la tabla correspondiente a los recibos válidos anteriormente generados para consultar cuál fue el último recibo emitido, posteriormente revisa la tabla de los recibos anulados para

consultar si el último número de recibo emitido es mayor al número de recibo encontrado en la tabla de los recibos válidos, si es así, se toma este como el último recibo usado y se aumenta al siguiente número de recibo de forma automática, de lo contrario se toma el número de recibo de la tabla de recibos válidos anteriormente generados y se aumenta en uno para pasar al siguiente recibo a utilizar.

Además, el sistema tiene la capacidad de detectar cuando una serie de recibo se encuentra por concluir e informa al usuario de ello para que este proceda a realizar el registro del nuevo rango de recibos de esa serie a utilizar.

Cómo se puede observar en todo momento nos encontramos con procesos de validación para confirmar que realmente el dato generado de manera automática corresponde al correcto.

4. Generar datos o pistas de auditoría.

Este apartado es muy importante para el control interno de una empresa. Los sistemas que forman parte del soporte y tratamiento de la información correspondiente a los ingresos o egresos de la empresa, así como otros tipos de sistema de información crítica, deben de cumplir con este control.

El sistema lleva un control de cada uno de los recibos anulados los cuales deben ser revisados y auditados para conocer el porqué de la anulación de dichos recibos. Recordemos que los recibos tienen un costo para la empresa y esta no puede muchas veces darse el lujo de estar perdiendo grandes cantidades de recibos por cualquiera que sea la razón. Es por esto que la cajera debe de responder y brindar una explicación del porqué de los recibos anulados.

Del proceso de pago

Para poder conocer realmente como se daban los procesos de recepción de pago de los estudiantes y las actividades que procedían a esta, fue necesario que estar presente durante se llevaran a cabo estas tareas.

El proceso inicia cuando el estudiante llega a la ventanilla de pago e informa a la cajera que hará efectivo el pago de algún o alguno de los conceptos posibles. Los diferentes conceptos de los cuales un estudiante puede llegar a pagar fueron catalogados por el administrador de la Facultad de Ciencias Económicas y Empresariales como los siguientes:

1. Pago de Mensualidad.
2. Pago de Matrícula.
3. Pago de Constancias o Certificados de Notas.
4. Pago de Alquiler de Equipo Informático.
5. Pago de Tutorías.
6. Pago de Cursos de Verano.
7. Pago de Reprogramaciones.
8. Inscripción de Protocolo.

Esto conformaría lo que se conoce como *Catálogo de Conceptos*, el cuál según las normas de contabilidad es necesario llevar para lograr un mejor control de los diferentes tipos de ingresos que se obtuvieron durante los estudiantes hicieron efectivo sus pagos.

Una vez que el horario de atención ha terminado, la cajera procede a arquearse y luego que esta tarea ha terminado procede a elaborar un informe de Caja en el cual se plasma los diferentes tipos de conceptos que se pagaron y el monto que se obtuvo por cada uno de ellos. A parte de eso, la cajera tiene la tarea de elaborar un informe que ellos llama *Alimentación de la Cartera* el cual hace referencia únicamente al concepto de pago de mensualidad e incluye el número de Carnet,

Nombres y Apellidos del estudiante, la cantidad de dinero que entregó, el mes al cual abono o canceló. Esto lo utilizan para alimentar el archivo de la cartera de los estudiantes y de esta manera llevar un control y permitir la emisión de los estados de cuenta de los mismos.

Una vez que la cajera ha terminado de elaborar los documentos que le corresponden, es obligación de la contadora proceder a realizar nuevamente un arqueo para corroborar que el informe de caja fue correctamente elaborado y el dinero con el que se cuenta es realmente el que se recibió usando como soporte los recibos que fueron emitidos.

Una vez concluido el arqueo, y todo fue encontrado en orden absoluto, la contadora de la Facultad de Ciencias Económicas y Empresariales procede a elaborar un informe de contabilidad donde hace entrega al administrador el dinero que será depositado. Cabe señalar que este informe de contabilidad es generado por el sistema de contabilidad validado por la UNAN – León que es el SOROYA. A la vez archiva los recibos que fueron emitidos en un folder correspondiente al mes en que se emitieron.

El administrador recibe el paquete enviado por la contadora y realiza las correspondientes revisiones de la información y si todo está en orden procede a elaborar un informe que incluye los datos del dinero recibido y envía una copia a la oficina de Contabilidad General y otra a la oficina de Auditoría General de la Universidad Nacional Autónoma de Nicaragua. Y el resto de información, junto con el informe de contabilidad es enviado a Tesorería General para que sean ellos quienes depositen el dinero en la cuenta respectiva.

A partir de este momento, el tratamiento del dinero y de la información generada en un día de atención queda fuera del alcance de la Facultad de Ciencias Económicas y Empresariales.

Esquema para una óptima cartera

¿Qué es la Cartera?

La cartera es el eje sobre el cual gira la liquidez de la empresa, es el componente principal del flujo de efectivo. En los cursos por encuentro, al ser estos autofinanciados por sus estudiantes, es necesario llevar un correcto control de dicha cartera ya que de lo contrario será difícil llevar el balance del flujo de efectivo que se maneja en dichos cursos. La cantidad masiva de estudiantes hace difícil obviar este elemento.

Según los resultados arrojados por las entrevistas realizadas al administrador de la Facultad de Ciencias Económicas y Empresariales, se logró recapitular y tomar por deducción cuales serían los diferentes elementos que conformarían una óptima cartera dentro de la empresa, que en este caso es la Facultad.

Antes de mencionarlo debemos de quedar claro que una óptima cartera si puede llegar a existir. Por lo menos eso es la opinión que comparten los distintos contadores que fueron consultados para efectuar este proyecto. Según ellos, la cartera es el componente más líquido del activo corriente. Existe una parte mínima de inversión que la empresa necesita en todo momento y ellos la llaman la *Cartera Constante*. Por otro lado y como consecuencia de un mayor crecimiento en facturación existe lo que se denomina cartera variable. Ellos opinan que esta es la que hay que controlar, puesto que una finanza sana, hay que determinar cuanta cartera debe de ser financiada con fuentes de largo plazo y cuanta con fuentes de corto plazo.

Según el administrador de la Facultad de Ciencias Económicas y Empresariales para él la cartera debe de ser:

1. Rotativa, no es el monto de la cartera lo que determina su importancia sino su rotación. A mayor rotación, mayor eficiencia.

2. La antigüedad de la cartera es otro factor importante para determinar si se trata de una cartera favorable. Esto quiere decir que es importante el grado de concentración porcentual o absoluta entre los distintos rangos de días.
3. La cartera debe de tener ecuación de liquidez, si vende de contado, compre de contado.
4. Un mayor aumento de estudiantes con arreglos de pagos, conlleva a un aumento de la cartera.

En resumen podemos decir que el manejo de la cartera exige políticas, normas y procedimientos. El control financiero de la misma requiere un manejo basado en los indicadores que permitan manejarla adecuadamente.

Diagnóstico de la situación actual de los Cursos por Encuentro de la Facultad de Ciencias Económicas y Empresariales, UNAN – León

Como es del conocimiento público, todas las empresas que se involucran en el área de oferta de servicios, como es el caso de la Facultad de Ciencias Económicas y Empresariales de la UNAN – León, es necesario que cuenten con un correcto Sistema Contable y Administrativo, para poder cumplir a cabalidad sus objetivos y hacer valer su misión y su visión como empresa.

Un Sistema Contable y Administrativo sólido en una institución estatal, es aquel cuyas bases son forjadas tomando en cuenta las Normas Técnicas de Control Interno (NTCI) emitidas por las Contraloría General de la República, ya que estas definen el nivel mínimo de calidad o marco general requerido para el control interno del sector público y proveen las bases para que los sistemas de administración y las Unidades da Auditoría Interna (UAI) puedan ser evaluadas.

Es por esto que me apoyé en las NTCI para brindar un análisis de evaluación concreto del Sistema Contable y Administrativo actual, responsable de llevar un control de los ingresos generados por los Cursos por Encuentro de la Facultad de Ciencias Económicas y Empresariales, UNAN –León.

La siguiente tabla muestra cada una de las operaciones administrativas y contables que son llevadas por la Oficina de Administración y Contabilidad de la Facultad de Ciencias Económicas y Empresariales, UNAN – León, para controlar y administrar los ingresos generados y la calidad del servicio prestado a los estudiantes de dicha modalidad de estudio.

Se identifican cuales son los factores del Riesgo (R), Debilidades (D) y Fortalezas (F), de igual manera se indica cómo administrarlos a través de las actividades de

control o elementos de los Sistemas de Administración que presentados en el Apéndice de las NTCI[12].

Descripción	R	D	F	Sistema de Administración	Nombre de la Norma General
1. La dependencia administrativa lleva a cabo el registro y control de los ingresos que se perciben por parte de los estudiantes de los Cursos por Encuentros.			**X**	5. Administración Financiera.	5.1. Registro y depósito de los Ingresos. 5.4. Documentos de respaldo, clasificación y Autorización de desembolsos.
2. La dependencia mantiene un fono destinado para gastos menores el cuál es manejado por una sola persona, que es la encargada de su resguardo.			**X**	5. Administración Financiera	5.5 Fondos fijos de Caja Chica.

[12] Normas Técnicas de Control Interno.

3. Todos los movimientos financieros, tales como ingreso de dinero, arqueo de caja, etc. son manejados de la siguiente manera:			**X**	6. Contabilidad Integrada.	6.9. Rendición de Cuentas.
a. Se elaboran informes de control de flujo de efectivo. b. Cada dinero que entra a caja es registrado y se genera su respectivo recibo al estudiante de forma manual.		**X**			6.10. Presentación de Estados Financieros.

Descripción	R	D	F	Sistema de Administración	Nombre de la Norma General
c. La oficina se guarda un recibo el cual pasa a formar parte del registro de los ingresos del mes.			X		6.11. Propiedad de la Documentación.
d. Además se elaboran informes que controlan el estado financiero de los Cursos por Encuentros.			X		
4. La generación y procesamiento de la cartera de cobro de los estudiantes es generada de forma manual y eso retrasa en		X		9. Tecnología de la Información.	9.2. Ingreso de Datos. 9.4. Procesamiento.

gran medida el procesamiento de los datos.					

MARCO METODOLÓGICO

I. Descripción del Proyecto

Este proyecto consta de herramientas de alta calidad que permiten mayor rapidez en el procesamiento de los datos. La ingeniería del software se centra en el hecho, establecimiento y uso de los principios robustos de la ingeniería. Orientados a obtener software económicos y fiables para que funcionen de manera eficiente en máquinas reales.

II. Material y Método

Material: Para el poder describir el material utilizado, es necesario dividirlo en la parte Hardware y la parte Software:

Hardware: Se empleará el siguiente equipo para llevar a cabo la elaboración, diseño y programación del sistema:

- Una PC con las siguientes características:
    - Procesador AMD Athlon™ II P340 Dual - Core Processor 2.20 GHz
    - Memoria RAM 3.00 GB
    - Sistema Operativo: Windows XP Professional
    - Disco Duro: 280GB

Software: Las herramientas software que se empleará en el desarrollo de la aplicación son:

- PostgreSQL 8.1 (Gestor de Base de Datos)

- Microsoft Visual Basic 6.0
- Microsoft Office 2003

Método: Indican como construir técnicamente el software con la ayuda de diferentes tipos de modelos (estructurado, cascada, espiral entre otros) los cuales proporcionan una serie de pasos que se deben seguir durante el proceso de desarrollo.

Los modelos del ciclo de vida realizan actividades comunes tales como:

- Gestión del proyecto por medio de la descomposición del mismo en etapas.
- Uso de alguna metodología de trabajo en cada etapa.
- Verificación y validación de cada fase de desarrollo.

III. Descripción del ciclo de vida

El ciclo de vida exige un enfoque sistemático y secuencial acerca del desarrollo del software que indica el nivel del sistema y progresa a través del análisis, diseño, codificación, prueba y mantenimiento.

Los modelos de desarrollo abordan el estudio de las actividades del sistema en las siguientes etapas:

- Planificación.
- Análisis.
- Diseño del sistema.
- Construcción y elaboración del sistema.

El modelo de desarrollo de sistemas que será implementado en nuestro sistema es el **MODELO EN CASCADA**.

Modelo en Cascada

Descompone el proceso de desarrollo en diferentes fases, constituyendo la salida de cada una de ellas la entrada requerida por la siguiente. En este modelo se supone que todos los requisitos son conocidos y comprendidos perfectamente al iniciar el desarrollo del software.

Actividades del Ciclo de vida en Cascada

1. **Ingeniería del sistema:** estudio de sistemas para obtener una convicción del análisis y diseño. Abarca el cómo trabaja un sistema existente y preparar modificaciones al sistema para cambiar el comportamiento del mismo.
2. **Análisis:** Esta fase contempla la identificación de requisitos software y la definición de los alcances del sistema para el tiempo de desarrollo planificado. Se realizarán las siguientes actividades:
    a. Entrevistas con cada el administrador de la Facultad el cual es el responsable de llevar el control de la cartera.
    b. Entrevista con la contadora de la Facultad. Ella es responsable de llevar el control de los ingresos que se tienen por parte de los estudiantes.
    c. Entrevista con las cajeras que actualmente laboran y que son las que de una manera directa se ven relacionadas con los estudiantes al momento de que estos llegan a efectuar sus respectivos pagos arancelarios.
    d. Entrevistas con el Decano de la Facultad para que se brinden las últimas consideraciones generales que según su criterio, el sistema deba tener.

3. **Diseño:** En esta fase se contempla la instalación de todas las herramientas necesarias para el desarrollo del sistema, así como la elaboración de una vista previa en papel de todos los requerimientos del sistema tanto las entradas como las salidas.

   Se realizarán las siguientes actividades:

   a. Elaboración del diseño de bases de datos, tendrá una duración 4 semanas.
   b. Elaboración en papel de todas las entradas y salidas del sistema, tendrá una duración 2 semanas.
   c. Determinación de todas las restricciones que tendrá el sistema en cuanto a validaciones de datos, tendrá una duración de 2 semanas.

4. **Codificación:** Esta fase es la que tiene mayor duración debido a que en ella se ejecutará todo lo que se contempló en las actividades anteriores. Se realizarán las siguientes actividades:

   a. Codificación necesaria para el acceso a la base de datos.
   b. Codificación de la plantilla principal que tendrá todo la aplicación.
   c. Codificación de cada una de las interfaces de entrada y salida de datos con sus respectivas operaciones y restricciones.
   d. Elaboración del manual de usuario.

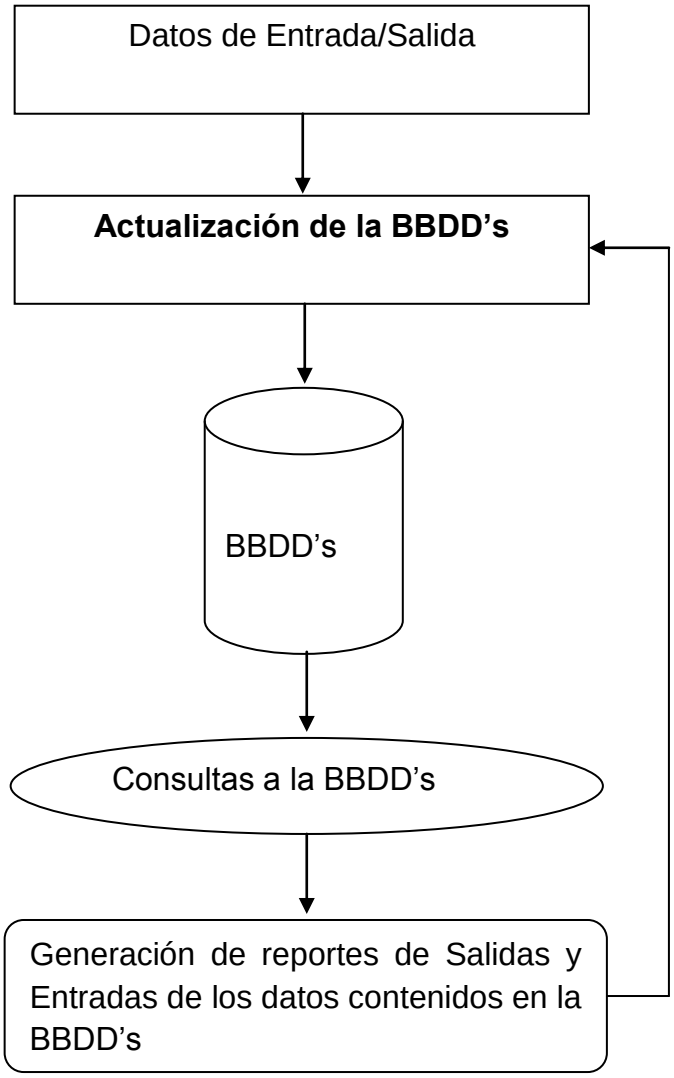
5. **Prueba:** El software debe ser probado para descubrir los defectos que puedan existir en la función, en la lógica y en la implementación. Se realizarán las siguientes actividades:

   a. Presentación del producto terminado ante las respectivas autoridades y demás personas autorizadas y que de alguna u otra manera tengan alguna relación con el sistema.
   b. Se recopilará toda la información de los errores que lance el sistema para su posterior corrección, esta se realizará simultáneamente con las otras actividades de esta fase.

6. **Mantenimiento:** En esta fase se harán todas las correcciones de errores tanto de funcionamiento como de procedimiento que se plantearon en la fase de prueba.

Análisis

Conceptualmente el sistema se muestra a como lo indica la siguiente gráfica:



Especificación de Requisitos del Software

1. Introducción.

1.1 Propósito

El propósito del proyecto es definir un conjunto de especificaciones de requisitos software los cuales se determinaron mediante el análisis de las actividades que a diario se llevan a cabo en las oficinas Administrativas de la Facultad de Ciencias Económicas y Empresariales de la mano de los usuarios finales que utilizaran el software, y los cuales debe de cumplir el sistema “Control de Caja”, que podrá ser utilizado para el control de sus actividades de recepción y control de dinero.

Este documento está destinado al Doctor Carlos Zúniga, administrador de la Facultad de Ciencias Económicas, que deberá estudiarlo para su aprobación o desacuerdo antes de abordar la siguiente fase.

1.2 Alcance

El nombre con el cual se identificará nuestra aplicación será: *Sistema de Control de Caja*.

El producto software deberá llevar a cabo las siguientes funciones:

- Captura de los datos de la Facultad.
- Seguridad mediante el uso de Contraseña.
- Captura de la Fecha de Inicio y Fin del mes en Proceso.
- Captura de los datos de los Estudiantes.
- Captura de los datos de las Carreras.
- Captura de los datos de los Docentes.

- Captura de los datos de la serie de Recibos a utilizar.
- Captura de los datos correspondientes a los Conceptos de Pago.
- Generación automática de los números de recibos a partir de las serie de Recibos ingresadas.
- Captura de los datos correspondientes a los recibos.
- Captura automática de Recibos Anulados.
- Generación de Reportes.
    - Reporte de Caja Diario.
    - Reporte de Caja por Rango de Fechas.
    - Reporte de Estado de Cuentas de los Estudiantes.
    - Reporte de Estudiantes Morosos por año lectivo y carrera.
    - Reporte de Flujo de Efectivo.
    - Reporte de Recibos Anulados.
- Generación de las siguientes Búsquedas.
    - Búsqueda de Recibos Emitidos por Fecha de Emisión.
    - Búsqueda de Recibos Emitidos por Número de Recibo.
    - Búsqueda de Recibos Anulados por Fecha de Anulación.
    - Búsqueda de Recibos Anulados por Número de Recibo.
    - Búsqueda de un Estudiante por Nombre.
    - Búsqueda de un Docente por Nombre.
- Captura del pago de un concepto realizado por un estudiante y emisión del recibo de pago del Estudiante.
- Modificación de los datos de la Facultad.
- Modificación de la Contraseña.
- Modificación de los datos correspondientes a los Conceptos de Pago.
- Modificación de los datos de los Estudiantes.
- Modificación de los datos de los Docentes.
- Modificación de los datos de las Carreras.
- Modificación de la Fecha del Mes en Proceso.
- Establecer el Cierre de Mes en Proceso.
- Realizar copia de seguridad de la base de datos del sistema en unidades de almacenamiento.

1.3 Definiciones de Acrónimos y Abreviaturas.

- **Facultad:** Organización dentro de la Universidad que ofrece determinadas Carreras a los Estudiantes y aparte ofrece servicios administrativos a los mismos.
- **Carrera:** Servicio de enseñanza ofertado por un Facultad
- **Estudiante:** Cada una de las personas que se encuentran matriculadas en alguna de las carreras que ofrece la Facultad de Ciencias Económicas.
- **Docentes:** Cada una de las personas encargadas de impartir un componente curricular o clase dentro de una **Carrera**.

- **Mes en proceso:** Identificador del mes en el que se encuentra llevándose a cabo cada una de las actividades de recepción de pagos y demás.
- **Pago:** Desembolso de dinero realizado a cambio de un bien o servicio.
- **Serie de recibos:** Cada recibo viene identificado por una letra respectiva que denota el tipo de moneda con el cual se hace efectivo el pago. En este caso la seria A de un recibo denota que el pago se ha efectuado en moneda nacional y la serie B denota que el pago se ha llevado a cabo en moneda extranjera.
- **Recibos:** Documento contable que demuestra de manera física que se ha llevado a cabo el pago de algún concepto por parte del estudiante.
- **Conceptos de pago:** Se refiere al servicio por el cual se ha desembolsado cierta cantidad de dinero y que forma parte del detalle de cada uno de los recibo. Cada concepto tiene un precio de pago distinto en dependencia del mismo.
- **Recibos anulados:** Documento contable que demuestra de manera física que se ha llevado a cabo el pago de algún concepto por parte del estudiante y que por distintas razones ha sido anulado. Entre las posibles causas de anulación de un recibo:
    - Recibo manchado a la hora de ser elaborado.
    - Datos incorrectos con el cual fue llenado el recibo.
    - Recibo físicamente dañado.
- **Reporte de caja:** Documento contable elaborado por la o las cajeras con el objetivo de mostrar de manera resumida cada uno de los pagos que se efectuaron. El documento puede ser emitido diariamente, semanalmente, mensualmente.
- **Estado de cuentas:** Documento contable elaborado por el administrador o la contadora con el objetivo de mostrar el estado de los pagos de cada estudiante efectuados durante el mes o el año. Con la ayuda de este documento se demuestra si el estudiante se encuentra solvente o pendiente de algún pago.
- **Estudiantes morosos:** Estudiantes con algún tipo de pago pendiente. Estos estudiantes son registrados con el objetivo de poder controlarlos a la hora de realizar algún examen o trámite especial dentro de la facultad, ya que para esto se necesita que los estudiantes se encuentren solventes en sus pagos.
- **Becas:** Ayuda que se le brinda a los estudiantes para sufragar sus gastos de estudio. La beca puede ser de dos tipos:
    - ***Beca Completa:*** Ayuda monetaria correspondiente al valor de la mensualidad del Curso por Encuentros que de manera automática es acreditada cada mes al estado de cuenta del estudiante.
    - ***Media Beca:*** Ayuda monetaria correspondiente a la mitad del valor de la mensualidad del Curso por Encuentros que de manera automática es acreditada cada mes al estado de cuenta del estudiante. La otra mitad debe ser sufragada por el estudiante.
- **Flujo de Efectivo:** Cantidad de dinero que diariamente entra a caja producto de los pagos que los estudiantes hacen efectivo. El flujo de efectivo puede

ser en Córdobas o en Dólares y es sacado diariamente para ser usado durante el arqueo de la o las cajeras.

- **Arqueo:** Actividad que se realiza diariamente con el objetivo de comparar que las entradas físicas de dinero corresponde de igual forma a las registradas en cada uno de los recibos que fueron emitidos. En caso de no ser así se procede a analizar el porqué del faltante.

1.4 Referencias.

- Información suministrada por el administrador de la facultad el Doctor Carlos Zúniga.
- Información suministrada por las cajeras y contadora de la facultad.
- Entrevistas realizadas a diferentes estudiantes para poder conocer las necesidades de los mismos y las deficiencias que actualmente cuenta la unidad administrativa contable de esta facultad.
- Documentos extraídos de internet:
    - Manual de PostgreSQL extraído de www.postgresql.org/docs/8.1/interactive/index.html

1.5 Visión General.

Se realizará una descripción general del producto a desarrollar para pasar posteriormente a estudiar cada uno de los Requisitos Específicos.

2. Descripción General

2.1. Relaciones del producto.

La aplicación se desarrollará e implementará en un equipo con las siguientes características:

- 70 GB en Disco Duro.
- 4 GB RAM.
- Intel Pentium IV.
- Sistema Operativo: Windows XP.

2.2. Funciones del producto.

- Manejo y control de los datos de los estudiantes.

- Manejo y control de los datos de los docentes.
- Manejo y control de los datos de las carreras.
- Registro de los recibos emitidos.
- Registro de los recibos anulados.
- Registro y control de las series de recibos a ser utilizadas.
- Registro y control de los conceptos de pago.
- Configuración de la fecha de inicio y fin del mes en proceso.
- Cambio de clave del usuario.
- Generación de reportes:
    - Reporte de Caja Diario.
    - Reporte de Caja por Rango de Fechas.
    - Reporte de Estado de Cuentas de los Estudiantes.
    - Reporte de Estudiantes Morosos por año lectivo y carrera.
    - Reporte de Flujo de Efectivo.
    - Reporte de Recibos Anulados.

2.3. Características del usuario.

El usuario final de aplicación debe tener conocimientos básicos en el manejo de sistemas de información computarizados.

2.4. Restricciones generales.

Las cajeras de la facultad de Ciencias Económicas y Empresariales que serán los usuarios finales de la aplicación deberán de tener como requisitos mínimos:

- PC con: 40 GB de Disco Duro, 256 MB de memoria RAM, Procesadores con velocidad de 1.6 GHz.
- Sistema Operativo Windows.
- Red de Área Local.

2.5. Seguridad de la aplicación

La aplicación contará con el siguiente nivel de seguridad:

- Autentificación de usuario vía aplicación. Cada uno de los usuarios autorizados a utilizar el sistema tendrán su usuario y contraseña de acceso. Así de esta manera nos aseguraremos que la información

sea únicamente manipulado por el usuario autorizado a hacerlo. Además el sistema permitirá trabajar con distintas cuentas lo que permitirá llevar un control de quien accede al sistema y que información manipula.

3. Requisitos Específicos.

3.1. Requisitos Funcionales.

3.1.1. Manejo y control de los datos de los estudiantes.

3.1.1.1. Especificación

3.1.1.1.1. Introducción

Esta función permitirá al usuario registrar en el sistema los datos correspondientes a los estudiantes de la modalidad Curso por Encuentros.

3.1.1.1.2. Entradas

Por pantalla:

a. Carnet: Este campo representa el número de carnet que cada estudiante posee y que lo identifica de manera única en la universidad.
b. Nombres: Este campo representa el o los nombres del estudiante.
c. Apellidos: Este campo representa el o los apellidos del estudiante.
d. Cédula: Documento de identificación que el estudiante como persona civil posee. Cabe señalar que este campo es opcional.
e. Dirección: Especifica la dirección de domicilio de un estudiante. Esto se hace con el objetivo de que se conozca el lugar donde reside el estudiante para facilitar su localización en caso de ser necesario.
f. Municipio: Municipio de residencia del estudiante.
g. Teléfono: Número de teléfono fijo o celular del estudiante. Este campo es opcional.
h. Grupo: Grupo de estudio del estudiante. Identificado generalmente con una letra del abecedario.
i. Activo: Determina si el estudiante se encuentra actualmente estudiando dentro de la facultad o se ha dado de baja al efectuar un retiro de matrícula.
j. Beca: Determina si el estudiante es becado completamente.
k. Media Beca: Determina si el estudiante posee únicamente media beca.

De manera automática:

a. Código de la Carrera: Cada carrera posee un código que la identifica del resto. El sistema almacenará automáticamente el código de la carrera a partir del nombre de la carrera que se seleccione.
b. Código del estudiante: Se genera automáticamente y no es visible para el usuario.

3.1.1.1.3. Proceso

Se mostrará al usuario una interfaz donde, por medio de una rejilla, podrá navegar en el registro completo de los estudiantes. Podrá realizar una búsqueda del estudiante que desea visualizar. Al dar clic en el botón de **Insertar** se visualizará una nueva ventana donde se pedirá el ingreso de los datos correspondientes a: Carnet, Nombres, Apellidos, Dirección, Municipio, Teléfono, Grupo, Activo, Beca, Media Beca.

De esta manera se podrá agregar un registro nuevo del estudiante lo que permitirá en todo momento mantener actualizada la base de datos del sistema.

Para guardar la información en la base de datos, el usuario deberá de pulsar el botón de **Guardar**.

3.1.1.1.4. Salidas

Se agregará un nuevo registro correspondiente para ese estudiante dentro de la entidad *Estudiantes*.

3.1.2. Manejo y control de los datos de los docentes

3.1.2.1. Especificación

3.1.2.1.1. Introducción

Esta función permitirá al usuario registrar en el sistema los datos correspondientes a los docentes de la modalidad Curso por Encuentros.

3.1.2.1.2. Entradas

Por pantalla:

a. INSS: Este campo representa el número de seguro social del docente.
b. Nombres: Este campo representa el o los nombres del docente.
c. Apellidos: Este campo representa el o los apellidos del docente.
d. Cédula: Documento de identificación que el docente como persona civil posee. Cabe señalar que este campo es opcional.
e. Dirección: Especifica la dirección de domicilio de un docente. Esto se hace con el objetivo de que se conozca el lugar donde reside el docente para facilitar su localización en caso de ser necesario.
f. Municipio: Municipio de residencia del estudiante.
g. Teléfono: Número de teléfono fijo o celular del estudiante. Este campo es opcional.
h. Titulo Académico: Ofrece las opciones de Licenciado, Master, Doctor, Alumno Ayudante, Otros. Este dato es obligatorio.

3.1.2.1.3. Proceso

Se mostrará al usuario una interfaz donde, por medio de una rejilla, podrá navegar en el registro completo de los docentes. Podrá realizar una búsqueda del docente que desea visualizar. Al dar clic en el botón de **Nuevo** se visualizará una nueva ventana donde se pedirá el ingreso de los datos correspondientes a: Cédula, Nombres, Apellidos, Dirección, Teléfono, Título Académico.

De esta manera se podrá agregar un registro nuevo del docente lo que permitirá en todo momento mantener actualizada la base de datos del sistema.

Para guardar la información en la base de datos, el usuario deberá de pulsar el botón de **Aceptar**.

3.1.2.1.4. Salidas

Se agregará un nuevo registro correspondiente para ese docente dentro de la entidad *Docentes*.

3.1.3. Manejo y control de los datos de la carrera.

3.1.3.1. Especificación

3.1.3.1.1. Introducción

Esta función permitirá al usuario registrar en el sistema los datos correspondientes a las carreras de la modalidad Curso por Encuentros.

3.1.3.1.2. Entradas

Por pantalla:

a. Carrera: Este campo representa el nombre de la carrera.

De manera automática se genera:

b. Cod. Carrera: Dato numérico que identifica de manera única los registros de cada una de las carreras.

3.1.3.1.3. Proceso

Se mostrará al usuario una interfaz donde, por medio de una rejilla, podrá navegar en el registro completo de las carreras. Al dar clic en el botón de **Nuevo** se visualizará una nueva ventana donde se pedirá el ingreso de los datos correspondientes a las carreras de los Cursos por Encuentros.

De esta manera se podrá agregar un registro nuevo de una carrera, lo que permitirá en todo momento mantener actualizada la base de datos del sistema.

Para guardar la información en la base de datos, el usuario deberá de pulsar el botón de **Aceptar**.

3.1.3.1.4. Salidas

Se agregará un nuevo registro correspondiente para esa carrera dentro de la entidad *Carreras*.

3.1.4. Registro de los recibos emitidos.

3.1.4.1. Especificación

3.1.4.1.1. Introducción

Esta función permitirá al usuario registrar en el sistema los datos correspondientes a los recibos emitidos por la cajera al momento de que un estudiante se hace presente para efectuar sus respectivos pagos arancelarios.

3.1.4.1.2. Entradas

Por pantalla:

a. Serie: Se selecciona la serie del recibo que se desea utilizar.
b. Fecha: Se selecciona la fecha para cual se desea generar el recibo. Por defecto es la fecha actual.
c. Monto en córdobas: La cantidad en córdobas que se el estudiante entrega para realizar el pago.
d. Monto en dólares: La cantidad en dólares que el estudiante entrega para realizar el pago.
e. Detalles del Concepto: Listado de los conceptos que serán pagados con el recibo.
f. Becado: Se selecciona si el estudiante posee una beca completa.
g. Media Beca: Se selecciona si el estudiante posee media beca.
h. Por cuenta de: Se escribe el nombre de la persona que esta haciendo efectivo el pago del arancel.

De manera automática:

i. No. Recibo: Generado automáticamente a partir de la serie seleccionada. Este lleva un consecutivo en base a todos los

números de recibo y sus correspondientes series registrados en el sistema.

j. Nombres, Apellidos, Carnet, Cédula: Se selecciona a partir de una lista existente de los estudiantes registrados en el sistema.
k. En letras: Se genera automáticamente en base al monto registrado en cada uno de los conceptos. Esto evita errores de escritura por parte del usuario.
l. En concepto de: Se selecciona a partir de los datos de los conceptos registrados en el sistema.
m. Importe en Córdobas: Es el monto total en córdobas a pagarse en el recibo. Se calcula automáticamente.
n. Importe en Dólares: Es el monto total en dólares a pagarse en el recibo. Se calcula automáticamente.
o. Cambio Dólar: Muestra la tasa de cambio del dólar. Se genera automáticamente a partir de la tasa de cambio registrada en el sistema.

3.1.4.1.3. Proceso

Se mostrará al usuario una interfaz desde la cual se pedirán cada uno de los datos por pantalla especificados anteriormente, el usuario podrá seleccionar a partir de un botón que mostrará una rejilla, el estudiante para el cual se le está realizando el pago y de igual manera, a través de otra rejilla se podrá seleccionar el concepto de pago del recibo. De esta manera se facilitará y reducirá el margen de error del usuario al ingresar los datos. Desde la misma interfaz en caso de ser necesario se podrá ingresar un nuevo concepto a través de la pulsación de un botón que será incluido dentro de esta misma interfaz. Luego que cada uno de los datos se han registrados, el usuario podrá pulsar un botón de Agregar que por medio del cual se podrá agregar un concepto al recibo, si es necesario agregar otro concepto, el usuario podrá hacerlo siguiendo el mismo procedimiento. De esta manera, el recibo podrá contar más de un concepto en él. Así mismo, habrá un botón de Eliminar, este botón podrá eliminar el concepto seleccionado en caso de existir algún error. Luego el sistema generará el Importe en Córdobas o el Importe en Dólares en base a la suma de cada uno de los conceptos registrados para el recibo.

Para guardar la información en la base de datos, el usuario deberá de pulsar el botón de **Registrar**. De igual manera podrá cancelar la operación al pulsar el botón **Cancelar.**

3.1.4.1.4. Salidas

Se agregará un nuevo registro correspondiente para ese recibo dentro de la entidad *Tabla Recibos* y *Detalle Transacción*.

3.1.5. Registro de los recibos anulados.

3.1.5.1. Especificación

3.1.5.1.1. Introducción

Esta función permitirá al usuario registrar en el sistema los datos correspondientes a los recibos anulados por algún error al momento de emitirlos.

3.1.5.1.2. Entradas

Por pantalla:

a. No. Recibo: Se digita el número del recibo que se desea anular.
b. Serie: Se selecciona la serie del recibo que se desea utilizar.
c. Anular recibo en blanco: Esta opción permite eliminar un recibo que aun no ha sido generado.

3.1.5.1.3. Proceso

Se mostrará al usuario primeramente una ventana solicitando la contraseña de autorización para poder anular el recibo. Una vez ingresada la contraseña correcta de autorización, se mostrará una interfaz desde la cual se pedirán cada uno de los datos por pantalla especificados anteriormente.

Para guardar la información en la base de datos, el usuario deberá de pulsar el botón de **Anular.**

3.1.5.1.4. Salidas

Se agregará un nuevo registro correspondiente para ese recibo anulado dentro de la entidad *Recibos Anulados y Detalle Recibo Anulado*.

3.1.6. Registro y control de las series de recibos a ser utilizadas.

3.1.6.1. Especificación

3.1.6.1.1. Introducción

Esta función permitirá al usuario registrar en el sistema los datos correspondientes a las series y los números de recibo de cada talonario. Esto permitirá generar de manera automática un consecutivo para cada recibo que se vaya emitiendo.

3.1.6.1.2. Entradas

Por pantalla:

a. Código Serie: Este campo representa la letra de la serie del recibo. Esta puede ser A, B, C o D. Para evitar algún error se muestra una lista ya definida.
b. Serie inicia en: Si no existe una numeración registrada anteriormente para la serie seleccionada, el sistema solicita el número con el que inicia el talonario. Si previamente ya se había registrado una numeración para la serie seleccionada, el sistema aumenta en uno el último número de recibo para dicha serie y pasa a ser el dato de este campo.
c. Usar a partir de este momento: Indica que la serie que se está ingresando será usada inmediatamente. Si no está seleccionado, el sistema registra la serie pero no la toma en cuenta al momento de realizar una transacción.

De manera automática se genera:

d. Serie finaliza en: Numeración final del talonario de la serie especificada en Código Serie.

3.1.6.1.3. Proceso

Se mostrará al usuario una interfaz donde, por medio de una rejilla, podrá navegar en el registro completo de las series registradas en el sistema y que aún se encuentran en uso. Al dar clic en el botón de **Especificar** el usuario podrá agregar un registro nuevo de una serie de recibos, lo que permitirá en todo momento mantener actualizada la serie de recibos utilizados para generar un recibo.

3.1.6.1.4. Salidas

Se agregará un nuevo registro correspondiente para esa serie de recibo dentro de la entidad *Tabla Serie Recibos*.

3.1.7. Registro y control de los conceptos de pago.

3.1.7.1. Especificación

3.1.7.1.1. Introducción

Esta función permitirá al usuario registrar en el sistema los datos correspondientes los conceptos de pago de los recibos. Esto permitirá generar de manera automática cada concepto a la hora de generar un recibo por una transacción.

3.1.7.1.2. Entradas

Por pantalla:

a. Cod. Concepto: Este campo recibe la información concerniente al código del concepto.
b. Concepto: Nombre del concepto de pago o descripción del mismo.
c. Monto del Concepto: Valor en dinero del concepto.

3.1.7.1.3. Proceso

A la hora de registrar una transacción de un recibo, se contará con un botón por medio del cual se le visualizará al usuario una ventana donde se le pedirá cada uno de los datos de entrada por pantalla especificados anteriormente. De esta manera se podrá mantener actualizada la base de datos de los registros de conceptos que se manejan en el sistema.

Para guardar la información en la base de datos, el usuario deberá pulsar el botón de **Ingresar**.

3.1.7.1.4. Salidas

Se agregará un nuevo registro correspondiente para ese concepto de pago dentro de la entidad *Conceptos*.

3.1.8. Configuración de la fecha de inicio y fin del mes en proceso.

3.1.8.1. Especificación

3.1.8.1.1. Introducción

Esta función permitirá al usuario especificar en el sistema la fecha de inicio y fin del mes que se está registrando cada una de las transacciones. Esto permitirá mantener cada una de las transacciones realizadas por los estudiantes ordenados según el mes en que fueron llevadas a cabo.

3.1.8.1.2. Entradas

Por pantalla:

a. Fecha de inicio: Este campo recibe la información concerniente a la fecha de inicio que corresponde al periodo de trabajo.
b. Fecha de fin: Este campo recibe la información concerniente a la fecha de fin que corresponde al periodo de trabajo.

3.1.8.1.3. Proceso

Se muestra primeramente una ventana en la que se solicita la contraseña de autorización para poder estipular el periodo de trabajo del sistema. Luego de haber ingresado la contraseña adecuada, el sistema muestra una ventana donde se especifican cada uno de los datos de entrada por pantalla mencionados anteriormente.

Para guardar la información en la base de datos, el usuario deberá pulsar el botón de **Aceptar**.

3.1.8.1.4. Salidas

Se agregará un nuevo registro correspondiente al periodo de trabajo dentro de la entidad *Periodo Trabajo*.

3.1.9. Cambio de clave del usuario.

3.1.9.1. Especificación

3.1.9.1.1. Introducción

Esta función permitirá al usuario especificar en el sistema la nueva clave de acceso al mismo por parte del usuario.

3.1.9.1.2. Entradas

Por pantalla:

c. Nueva Contraseña: Este campo recibe la información concerniente a la contraseña que el usuario desea establecer como su nueva contraseña de acceso al sistema.
d. Contraseña: El usuario escribe nuevamente la nueva contraseña como medida de seguridad para corroborar que la ha escrito correctamente.

3.1.9.1.3. Proceso

Primeramente se mostrará una ventana en la cual se pedirá la introducción por pantalla de la contraseña de acceso actual del sistema. Una vez que la contraseña ha sido ingresada, se mostrará una ventana en la cual se pedirán cada uno de los datos por pantalla mostrados anteriormente.

Para guardar la información en la base de datos, el usuario deberá pulsar el botón de **Aceptar**.

3.1.9.1.4. Salidas

Se agregará un nuevo registro correspondiente para ese concepto de pago dentro de la entidad *Contrasenya*.

3.1.10. Reporte de caja diario.

3.1.10.1. Especificación

3.1.10.1.1. Introducción

Esta función permitirá al usuario visualizar un reporte de los ingresos obtenidos por caja en el día en base a lo que se haya registrado en la entidad *Tabla Recibos* y *Detalle Recibos*.

3.1.10.1.2. Entradas

Por pantalla:

No se genera ninguna entrada por pantalla.

3.1.10.1.3. Proceso

El usuario dará clic en la opción *Reporte de Caja Diario* del menú *Imprimir*.

3.1.10.1.4. Salidas

Se mostrará un informe al usuario que contenga los montos de cada uno de los conceptos pagados por los estudiantes ese día, tanto en córdobas como en dólares.

3.1.11. Reporte de caja en un periodo de tiempo.

3.1.11.1. Especificación

3.1.11.1.1. Introducción

Esta función permitirá al usuario visualizar un reporte de los ingresos obtenidos por caja en un periodo de tiempo en base a lo que se haya registrado en la entidad *Tabla Recibos* y *Detalle Recibos*.

3.1.11.1.2. Entradas

Por pantalla:

a. Fecha de Inicio: Fecha a partir de la cual iniciará a generarse el reporte.
b. Fecha de Fin: Fecha final en la que se desea cortar el reporte.

3.1.11.1.3. Proceso

El usuario dará clic en la opción *Reporte de Caja en un Rango de Fecha* del menú *Imprimir*. Se mostrará por pantalla una interfaz que recoja los datos concernientes a: Fecha de Inicio y Fecha de Fin respectivamente.

Para mostrar el formulario, el usuario deberá de dar clic en **Aceptar**.

3.1.11.1.4. Salidas

Se mostrará un informe al usuario que contenga el monto de cada uno de los conceptos pagados por los estudiantes dentro de ese rango de fechas, tanto en córdobas como en dólares.

3.1.12. Reporte de Estado de Cuentas de los Estudiantes.

3.1.12.1. Especificación

3.1.12.1.1. Introducción

Esta función permitirá al usuario visualizar un reporte que muestre el estado de cuenta de un estudiante en particular.

3.1.12.1.2. Entradas

Por pantalla:

a. Carnet: Número de Carnet del estudiante que lo identifica de manera única.
b. Año: Año calendario para el cual se desea visualizar el estado de cuenta del estudiante.

3.1.12.1.3. Proceso

El usuario dará clic en la opción *Reporte de Estado de Cuenta* del menú *Imprimir*. Se mostrará por pantalla una interfaz que recoja los datos concernientes a: Carnet y Año respectivamente.

Para mostrar el formulario, el usuario deberá de dar clic en **Aceptar**.

3.1.12.1.4. Salidas

Se mostrará un informe al usuario que muestre el estado de cuenta del estudiante con todos los meses que ha pagado y/o debe en el año especificado por el usuario.

3.1.13. Reporte de Estudiantes Morosos por año lectivo y carrera.

3.1.13.1. Especificación

3.1.13.1.1. Introducción

Esta función permitirá al usuario visualizar un reporte que muestre un listado de los estudiantes que se encuentran pendientes de pago de su mensualidad en un determinado año y carrera.

3.1.13.1.2. Entradas

Por pantalla:

a. Año Lectivo: Año lectivo de los estudiantes que se desean visualizar como morosos.
b. Carrera: Carrera que desean ver si existen estudiantes morosos.

3.1.13.1.3. Proceso

El usuario dará clic en la opción *Reporte de Estudiantes Morosos* del menú *Imprimir*. Se mostrará por pantalla una interfaz que recoja los datos concernientes a: Año Lectivo y Carrera respectivamente.

Para mostrar el formulario, el usuario deberá de dar clic en **Aceptar**.

3.1.13.1.4. Salidas

Se mostrará un informe al usuario que contenga el listado de cada uno de los estudiantes que se encuentren morosos en sus mensualidades.

3.1.14. Reporte de Flujo de Efectivo.

3.1.14.1. Especificación

3.1.14.1.1. Introducción

Esta función permitirá al usuario visualizar un reporte que muestre el total de ingresos en córdobas y el total de ingresos en dólares del día. Una especie de hoja de resumen de ingresos que ayude al momento de realizar el arqueo de caja.

3.1.14.1.2. Entradas

Por pantalla:

a. Fecha: La fecha que se desea ver el reporte de flujo de efectivo, es decir, el dinero que se percibió en dicha fecha.

3.1.14.1.3. Proceso

El usuario dará clic en la opción *Reporte de Flujo de Efectivo* del menú *Imprimir*. Se mostrará por pantalla una interfaz que recoja los datos concernientes a: Fecha.

Para mostrar el formulario, el usuario deberá de dar clic en **Aceptar**.

3.1.14.1.4. Salidas

Se mostrará un informe al usuario que contenga la cantidad total de dinero que se percibió para cada una de las series de recibo que se han emitido.

3.1.15. Reporte de Recibos Anulados.

3.1.15.1. Especificación

3.1.15.1.1. Introducción

Esta función permitirá al usuario visualizar un reporte que muestre los recibos anulados en un rango de fechas definidas por el usuario.

3.1.15.1.2. Entradas

Por pantalla:

a. Fecha de inicio: La fecha de inicio a partir de la cual se desea comenzar a listar los recibos anulados.

b. Fecha de fin: Corresponde a la fecha en la que el reporte deja de mostrar recibos anulados.

3.1.15.1.3. Proceso

El usuario dará clic en la opción *Reporte de Recibos Anulados* del menú *Imprimir*. Se mostrará por pantalla una interfaz que recoja los datos concernientes a: Fecha.

Para mostrar el formulario, el usuario deberá de dar clic en **Aceptar**.

3.1.15.1.4. Salidas

Se mostrará un informe al usuario que contenga un listado de los recibos anulados, junto con las razones por las cuales se anularon, en un rango de fechas definidas por el usuario.

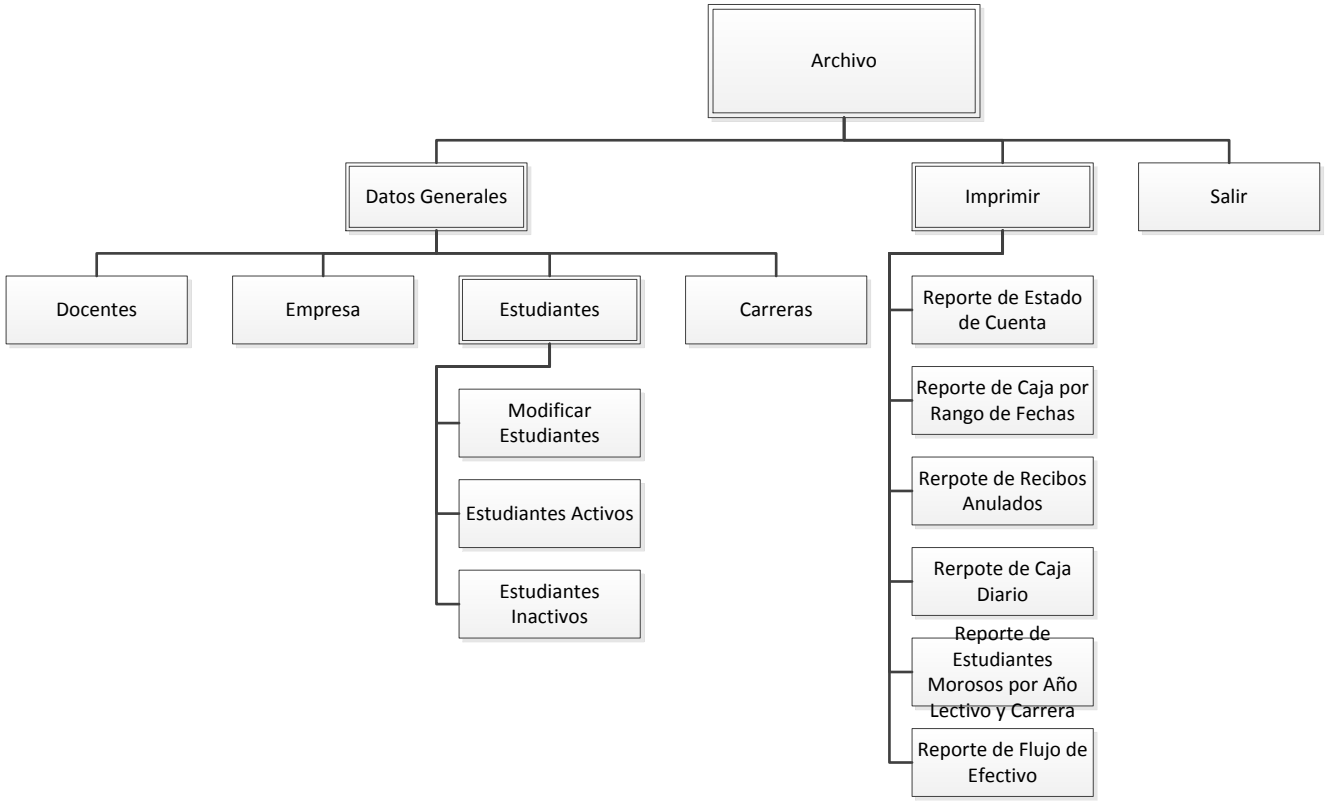
4. Características del usuario.

Los usuarios finales de la aplicación serán personas con conocimientos básicos sobre el manejo del programa y además deben de estar autorizadas para utilizar el software.

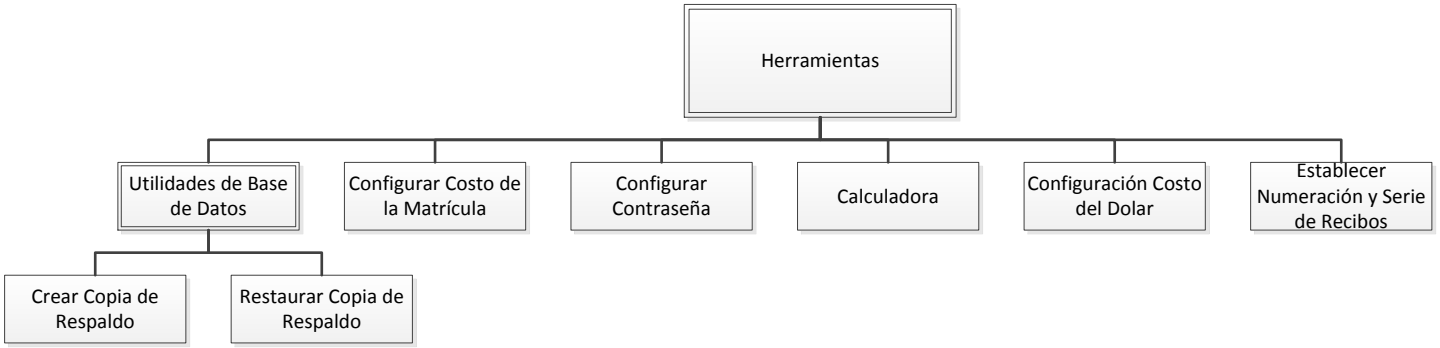
5. Diseño.

5.1. Diseño Arquitectónico.

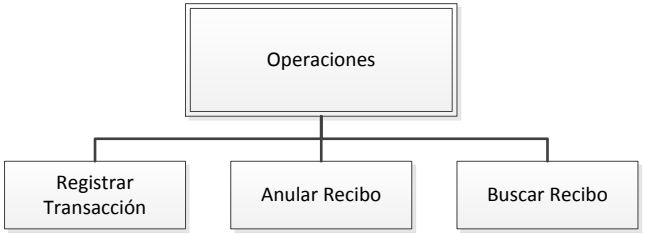
- Menú Archivo



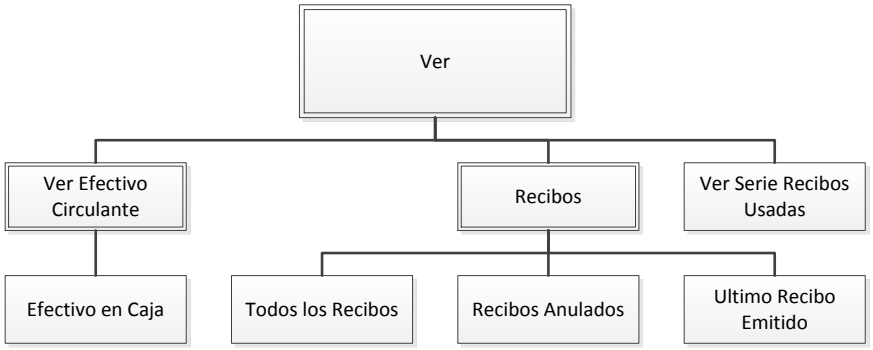
- Menú Herramientas



- Menú Operaciones



- Menú Ver


Ver
Ver Efectivo Circulante
Recibos
Ver Serie Recibos Usadas
Efectivo en Caja
Todos los Recibos
Recibos Anulados
Ultimo Recibo Emitido

5.2. Diseño de Datos

- Tabla Estudiante

carnet	varchar	10
nombres	varchar	30
apellidos	varchar	30
cedula	varchar	14
fecha_naci	date	0
sexo	varchar	12
cod_pais	varchar	3
cod_muni	varchar	4
telefono	varchar	8
email	varchar	30
codcarr	varchar	3
direccion	varchar	70
observacion	varchar	70
fechains	date	0
codusuario	varchar	3
anyo_est	varchar	0
situacion_aca	varchar	15
tipo_estu	varchar	12
tipo_mat	varchar	20
plan_est	varchar	0
anyo_lec	varchar	4

- Tabla Carrera

codcarr	varchar	3
nombre_carrera	varchar	30

- Tabla Docentes

coddocente	varchar	0
cedula	varchar	14
nombres	varchar	30
apellidos	varchar	30
telefono	varchar	7
direccion	varchar	70
tituloacademico	varchar	10

- Tabla Recibos

carnet	varchar	10
codmesproceso	varchar	3
codtransaccion	varchar	0
norecibo	varchar	0
serie	char	1
recibide	varchar	50
debe	float8	53
fecha	date	0

- Tabla Detalle_ Transacción

codtransaccion	varchar	0
codconcepto	varchar	10
norecibo	varchar	0
tipo_transaccion	varchar	3
valor_transaccion	float8	53
enletras	varchar	280

- Tabla Conceptos

codconcepto	varchar	10
concepto	varchar	0
costoconcepto	float8	53

- Tabla Abonos

carnet	varchar	10
codtransaccion	varchar	10
fecha	date	0
montoabono	float8	53
saldopendiente	float8	53
conceptoabono	varchar	0

- Tabla Serie _Recibos

codserie	char	1
reciboini	numeric	0
recibofin	numeric	0
fecha_uso	date	0
en_uso	char	1

- Tabla Saldos _Anulados

carnet	varchar	10
fecha_saldo	date	0
norecibo	varchar	0
debe	float8	53
haber	float8	53

- Tabla Saldos _Estudiantiles

carnet	varchar	10
fecha_saldo	date	0
norecibo	varchar	0
debe	float8	53
haber	float8	53

- Tabla Recibos _Anulados

carnet	varchar	10
codtransaccion	varchar	0
codmesproceso	varchar	0
norecibo	varchar	0
serie	char	1
recibi_de	varchar	50
debe	float8	53
fecha	date	0

- Tabla Detalle _Recibos _Anulados

codtransaccion	varchar	0
norecibo	varchar	0
tipo_transaccion	varchar	3
codconcepto	varchar	10
valor_transaccion	float8	53
enletras	varchar	280

Arquitectura SQL de la Base de Datos.

Se procedió al diseño de la estructura de la Base de Datos la cual debería de ser la adecuada, acorde a las necesidades de tratamiento de la información planteadas con anterioridad.

El esquema de las tablas propuesto es el siguiente:

```
CREATE TABLE mes En Proceso (
        Cod mes proceso character varying(3),
        Fecha inicio date NOT NULL,
        Fecha fin date NOT NULL,
        CONSTRAINT mesEnProcesopk PRIMARY KEY (codmesproceso)
);

CREATE TABLE carreras (
    codcarr character varying(3) NOT NULL,
    nombre_carrera character varying(30) NOT NULL,
    CONSTRAINT carreraspk PRIMARY KEY (codcarr)
);

CREATE TABLE costomatricula (
    costomatricula double precision DEFAULT 0
);
```

```
CREATE TABLE docentes (
    coddocente character varying NOT NULL,
    cedula character varying(14),
    nombres character varying(30) NOT NULL,
    apellidos character varying(30) NOT NULL,
    telefono character varying(7),
    direccion character varying(70),
    tituloacademico character varying(10)
);

CREATE TABLE efectivo_circulante (
    efect_sa double precision DEFAULT 0,
    efect_sc double precision DEFAULT 0,
    efect_sb double precision DEFAULT 0,
    efect_sd double precision DEFAULT 0
);
```

```
CREATE TABLE estudiante (
    carnet character varying(10) NOT NULL,
    nombres character varying(30) NOT NULL,
    apellidos character varying(30) NOT NULL,
    cedula character varying(14) NOT NULL,
    fecha_naci date NOT NULL,
    sexo character varying(12) NOT NULL,
    cod_pais character varying(3),
    telefono character varying(8),
    email character varying(30),
    codcarr character varying(3),
    direccion character varying(70),
    observacion character varying(70),
    fechains date,
    codusuario character varying(3),
    anyo_est character varying NOT NULL,
    situacion_aca character varying(15) NOT NULL,
    tipo_estu character varying(12) NOT NULL,
    tipo_mat character varying(20),
    plan_est character varying,
    anyo_lec character varying(4) NOT NULL,
```

```
    CONSTRAINT estudiantepk PRIMARY KEY (carnet),
    CONSTRAINT carrerasestudfk FOREIGN KEY (codcarr)
      REFERENCES carreras (codcarr) MATCH SIMPLE
      ON UPDATE CASCADE ON DELETE CASCADE
);

CREATE TABLE conceptos (
    codconcepto character varying(10) NOT NULL,
    concepto character varying NOT NULL,
    costoconcepto double precision DEFAULT 0,
    CONSTRAINT conceptospk PRIMARY KEY (codconcepto)
);

CREATE TABLE recibos (
    carnet character varying(10) NOT NULL,
    codmesproceso character varying(3),
    codtransaccion character varying NOT NULL,
    norecibo character varying NOT NULL,
    serie character(1) NOT NULL,
    recibide character varying(50) NOT NULL,
    debe double precision DEFAULT 0,
```

```
    fecha date NOT NULL,
        CONSTRAINT tb_recibopk PRIMARY KEY (codtransaccion),
        CONSTRAINT mesrecibofk FOREIGN KEY (codmesproceso)
      REFERENCES mesEnProceso (codmesproceso) MATCH SIMPLE
      ON UPDATE CASCADE ON DELETE CASCADE,
      CONSTRAINT carnetrecibofk FOREIGN KEY (carnet)
      REFERENCES estudiante (carnet) MATCH SIMPLE
      ON UPDATE CASCADE ON DELETE CASCADE
);

CREATE TABLE detalle_transaccion (
    codtransaccion character varying NOT NULL,
    codconcepto character varying(10),
    norecibo character varying NOT NULL,
    tipo_transaccion character varying(3) NOT NULL,
    valor_transaccion double precision DEFAULT 0,
    enletras character varying(280),
        CONSTRAINT tb_detalletransfk FOREIGN KEY (codtransaccion)
      REFERENCES recibos (codtransaccion) MATCH SIMPLE
      ON UPDATE CASCADE ON DELETE CASCADE,
        CONSTRAINT detalletransconcepfk FOREIGN KEY (codconcepto)
```

```
    REFERENCES conceptos (codconcepto) MATCH SIMPLE
    ON UPDATE CASCADE ON DELETE CASCADE
);

CREATE TABLE recibos_anul_solos (
    norecibo character varying NOT NULL,
    serie character(1) NOT NULL
);

CREATE TABLE recibos_anulados (
    carnet character varying(10) NOT NULL,
    codtransaccion character varying NOT NULL,
    codmesproceso character varying NOT NULL,
    norecibo character varying NOT NULL,
    serie character(1) NOT NULL,
    recibi_de character varying(50) NOT NULL,
    debe double precision DEFAULT 0,
    fecha date NOT NULL,
    CONSTRAINT recibosanulpk PRIMARY KEY (codtransaccion)
);
```

```
CREATE TABLE detalle_recibos_anulados (
    codtransaccion character varying NOT NULL,
    norecibo character varying NOT NULL,
    tipo_transaccion character varying(3) NOT NULL,
    codconcepto character varying(10) NOT NULL,
    valor_transaccion double precision DEFAULT 0,
    enletras character varying(280),
    CONSTRAINT detrecibanulfk FOREIGN KEY (codtransaccion)
      REFERENCES recibos_anulados (codtransaccion) MATCH SIMPLE
      ON UPDATE CASCADE ON DELETE CASCADE
);

CREATE TABLE saldos_anulados (
    carnet character varying(10) NOT NULL,
    fecha_saldo date NOT NULL,
    norecibo character varying NOT NULL,
    debe double precision DEFAULT 0,
    haber double precision DEFAULT 0
);
```

```
CREATE TABLE saldos_estudiantiles (
    carnet character varying(10) NOT NULL,
    fecha_saldo date NOT NULL,
    norecibo character varying NOT NULL,
    debe double precision DEFAULT 0,
    haber double precision DEFAULT 0,
        CONSTRAINT saldosestudfk FOREIGN KEY (carnet)
    REFERENCES estudiante (carnet) MATCH SIMPLE
    ON UPDATE CASCADE ON DELETE CASCADE

);

CREATE TABLE serie_recibos (
    codserie character(1) NOT NULL,
    reciboini numeric NOT NULL,
    recibofin numeric NOT NULL,
    fecha_uso date NOT NULL,
    en_uso character(1) NOT NULL
);
```

```
CREATE TABLE tempdetalletrans (
    codtransaccion character varying NOT NULL,
    NoRecibo character varying NOT NULL,
    tipo_transaccion character varying(3) NOT NULL,
    codconcepto character varying(10),
    valor_transaccion double precision DEFAULT 0,
    EnLetras character varying(280)
);

CREATE TABLE abonos (
    carnet character varying(10) NOT NULL,
    codtransaccion character varying(10) NOT NULL,
    fecha date NOT NULL,
    montoabono double precision DEFAULT 0 NOT NULL,
    saldopendiente double precision DEFAULT 0,
    conceptoabono character varying
);

CREATE TABLE cambiodolar (
    tipo_cambio double precision DEFAULT 0,
    fecha_cambio date NOT NULL);
```

CONCLUSIONES

Una vez terminada las fases de organización, clasificación y análisis de la información del proceso de control Administrativo y Contable de los Cursos por Encuentros y después de haber presentado los resultados de nuestro estudio, se considera oportuno expresar algunas conclusiones a las cuales se llegó a lo largo del desarrollo de la investigación.

- Dentro del área administrativa de los Cursos por Encuentro de la Facultad de Ciencias Económicas y Empresariales, existen muy pocas debilidades referentes al nivel operativo y contable. La mayoría de los procesos de realizados en esta área cumplen con las Normas Técnicas de Control Interno y esto le permite optimizar el proceso administrativo.

- La única debilidad que se encontró fue que todo el proceso de emisión de recibos, generación de documentos, estados de cuentas y manejo de la cartera son realizados de manera manual. Esto conlleva a pérdidas valiosas de tiempo y ocasiona inconformidad por parte de los estudiantes que demandan un mejor servicio.

- La cartera de la empresa exige un manejo de la información automatizado. El área administrativa carece de ese manejo por lo que el mantenimiento de la cartera es lento y no permite reflejar el estado correcto y confiable de las cuentas por cobrar que tiene la Facultad. Esto quiere decir que existen estudiantes con un gran tiempo de retraso en sus pagos, pero la poca eficiencia en el manejo de la información imposibilita el control de los mismos.

- Se encontraron inconformidades en la forma de generación de algunos documentos debido a la incomodidad y pérdida de tiempo al prepararlos.

Esto debido a la misma razones, una lenta y deplorable forma de manejo de la información.

- No existen mecanismo de respaldo de la información. Todos los documentos y recibos emitidos se encuentran físicamente archivados pero no existe ninguna otra forma de respaldo automatizado. Esto conlleva a grandes volúmenes de información que a la hora de ser necesitada dificulta su tratamiento.

RECOMENDACIONES

En base a las conclusiones planteadas anteriormente, se emiten una serie de recomendaciones que deben de llevarse a cabo si se desea aplicar una forma de manejo adecuada al proceso administrativo y contable de los Cursos por Encuentro de la Facultad de Ciencias Económicas y Empresariales. Dichas recomendaciones se citan a continuación:

1. Es necesario que el Departamento Administrativo de la Facultad de Ciencias Económicas y Empresariales acople el nuevo sistema generado a su antigua forma de trabajo, debido a que es imposible medir el grado de pérdidas y ganancias que se están ocasionando debido a un ineficiente tratamiento de la información, ni mucho menos pueden reflejar una correcta gestión administrativa con esta forma operativa.

Para llegar a cumplir lo dicho con anterioridad se debe de:

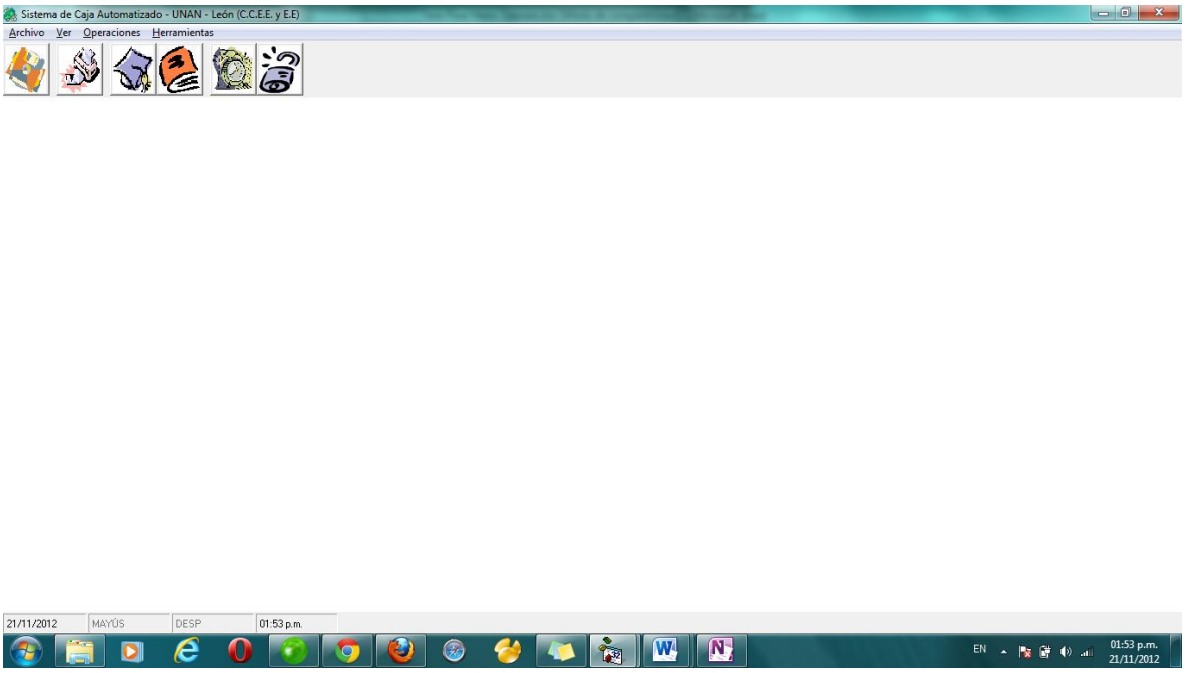
a. Se recomienda la emisión de políticas que ayuden a recuperar el monto de la cartera que aun se encuentra perdido debido a aquellos con mensualidades pendientes.

b. Se recomienda iniciar el uso del Sistema de Caja Automatizado partiendo de un nuevo año. Esto porque llevaría tiempo adecuarlo a la forma de trabajo que se ha venido llevando. Esto acarrearía retrasos e inconvenientes que podrían ser atribuidos a un mal uso del sistema.

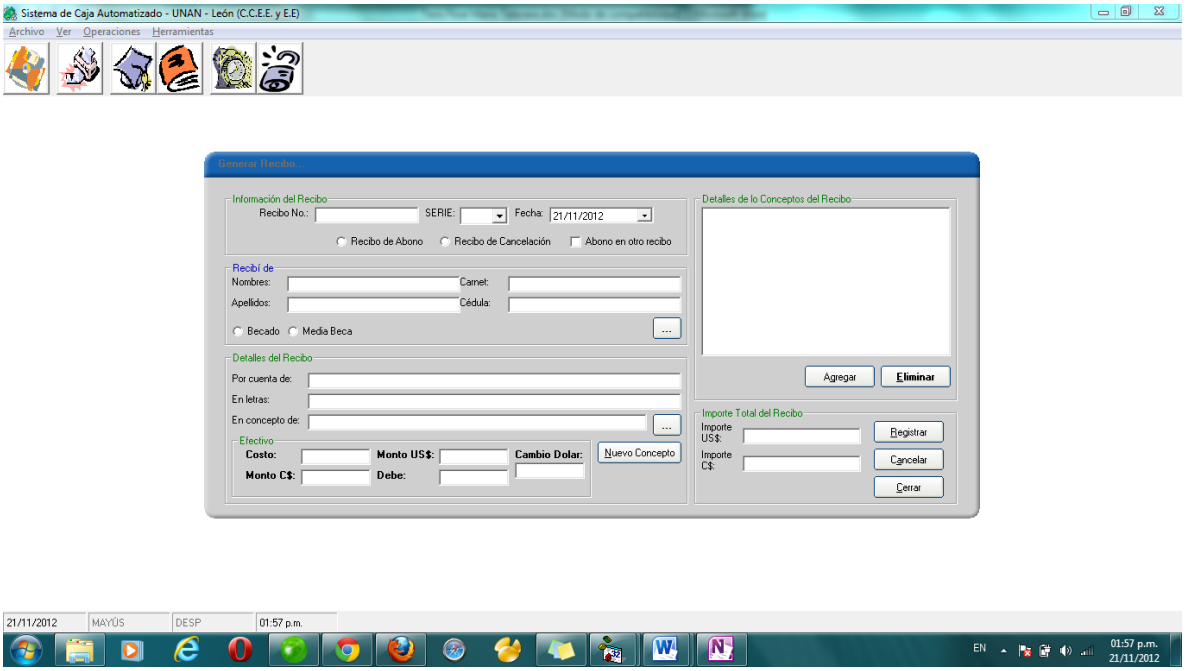
c. Se recomienda la capacitación de los posibles usuarios para que tengan conocimientos suficientes en el manejo de la nueva herramienta informática. Esto agilizará el proceso de aprendizaje y uso de la misma.

Anexos

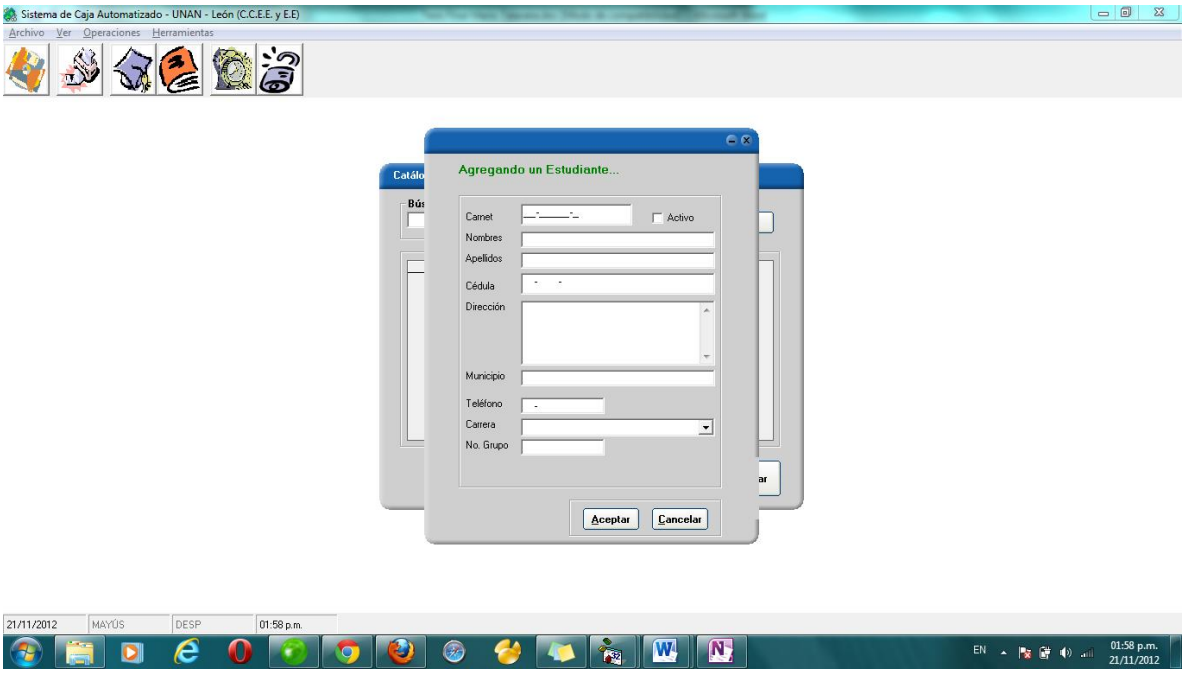
- Interfaz de Inicio.



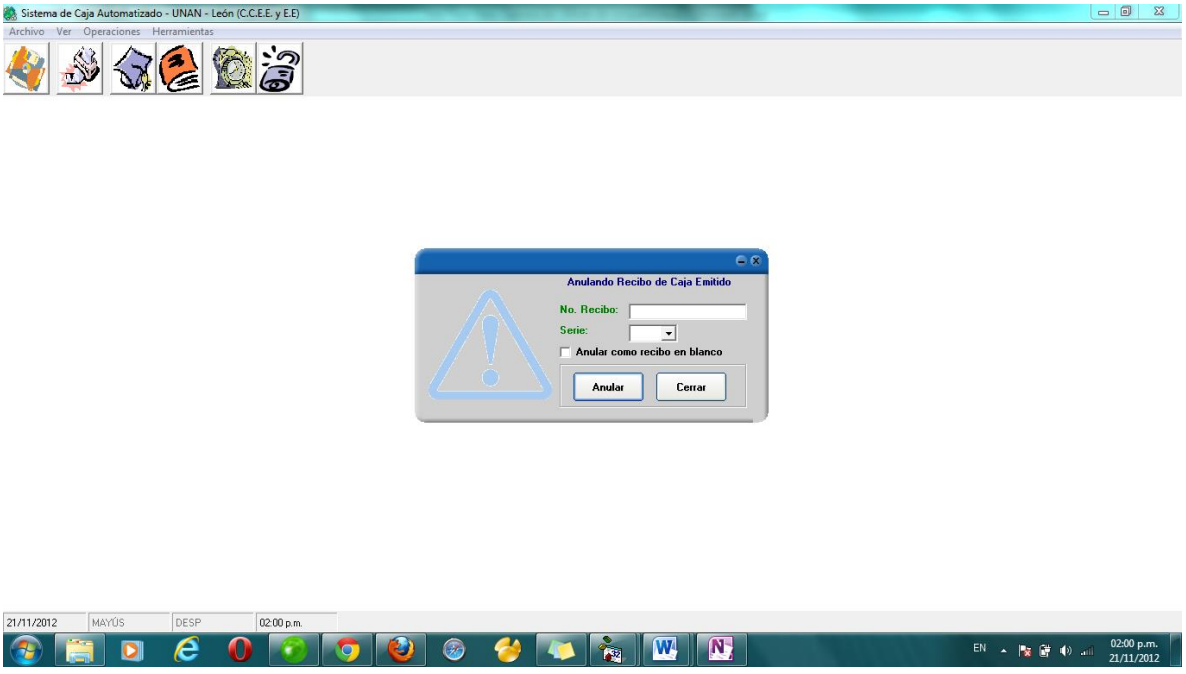
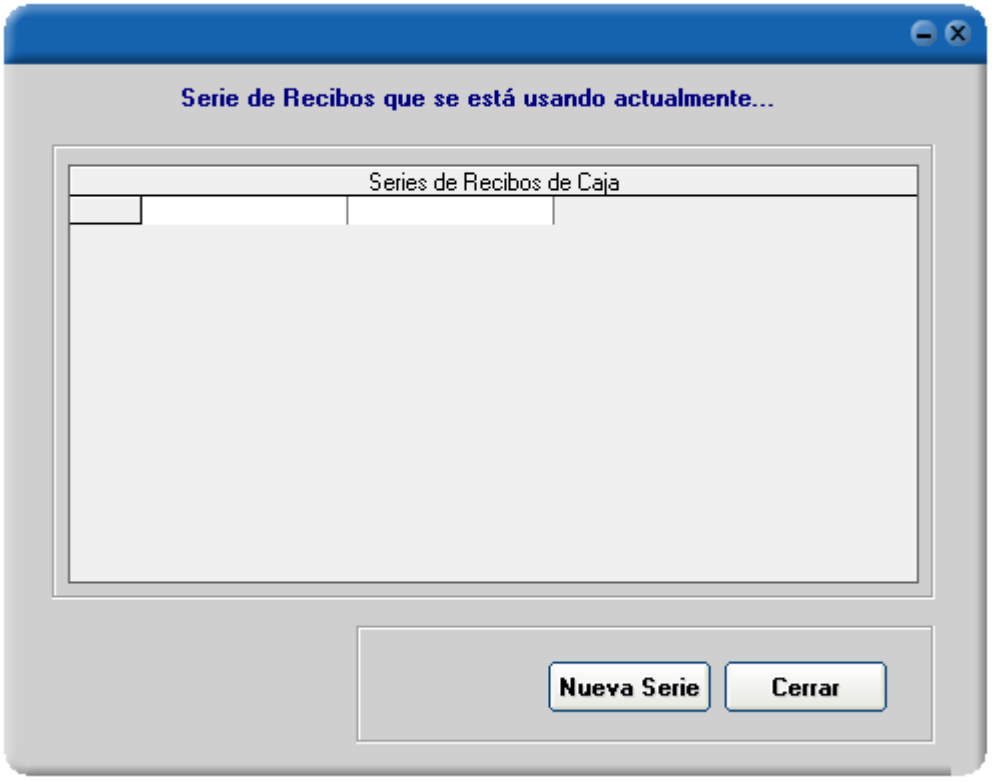
- Interfaz para registrar transacción.



- Interfaz para agregar un nuevo estudiante.



- Interfaz para anular un recibo.



- Interfaz para gestionar las series de recibos.