

**UNIVERSIDAD NACIONAL AUTÓNOMA DE NICARAGUA**  
**UNAN-León**

Facultad de Ciencias y Tecnología  
Departamento de Computación



Monografía para optar al título de Ingeniería en Sistemas de Información:  
**“Sistema Contable Cafetalera San José (SCCSJ 2010)”**

**Autores:**

Br. Adonis Emir Barrera Zambrana  
Br. Abismael Jonathan Bonilla Núñez  
Br. Frankling Alberto Gallo Muñoz

**Tutor:**

Msc. Álvaro Altamirano Osorio

León, Nicaragua, Mayo 2010.



**A DIOS**

Por haberme permitido lograr culminar mis estudios y por todas las cosas maravillosas que me ha dado.

**A MIS PADRES**

Por el apoyo incondicional, los consejos de perseverancia y dedicación, por haberme forjado con las virtudes necesarias para el recorrido de esta travesía.

**A MI HERMANA Y DEMAS FAMILIARES**

Por influir de una u otra manera en el paso por mis estudios, por sus buenas intenciones y palabras de aliento.

**A MI TUTOR**

Por haberme brindado su apoyo en la elaboración de esta monografía y por los conocimientos transmitidos a lo largo de la carrera.

**A MIS AMIGOS**

Por haber compartido momentos de alegría y por ser buenos compañeros.

**Adonis E. Barrera**

**A DIOS**

Por haberme permitido lograr culminar mis estudios y por todas las cosas maravillosas que nos ha dado.

**A mis padres**

Por el apoyo incondicional, los consejos de perseverancia y dedicación, por haberme forjado con las virtudes necesarias para el recorrido de esta travesía.

**A nuestro tutor**

Por habernos brindado su apoyo en la elaboración de esta monografía y por los conocimientos transmitidos a lo largo de la carrera.

**Franklin A. Gallo**

**A DIOS**

Le doy infinitamente las gracias a Dios, porque él me ha permitido escalar un peldaño más; haber podido culminar mi carrera ha sido una bendición de él para conmigo.

**A mis padres**

Ellos han sido piezas claves en mi vida; la confianza que han depositado en mí y el amor que han tenido para conmigo ha sido un impulso pleno para poder ser lo que hoy soy, Dios les bendiga grandemente.

**A nuestro tutor**

Su paciencia y compromiso para con nosotros, se ha visto reflejado en nuestro trabajo. Le agradecemos grandemente, Dios le bendiga.

**Abismael J.Bonilla**



**INDICE**

|             | <b>Contenido</b>                                  | <b>Páginas</b> |
|-------------|---|----------------|
| <b>I.</b>   | Introducción                                      | 1              |
| <b>II.</b>  | Antecedentes                                      | 2              |
| <b>III.</b> | Justificación                                     | 3              |
| <b>IV.</b>  | Objetivos   | 4              |
| <b>V.</b>   | Marco Teórico                                     | 5              |
| <b>5.1</b>  | ¿Qué es un sistema?                               |                |
| <b>5.2</b>  | ¿Sistema informático?                             |                |
| <b>5.3</b>  | Aplicaciones del software                         | 7              |
| <b>5.4</b>  | ¿Qué es un usuario?                               | 9              |
| <b>5.5</b>  | ¿Qué es un análisis?                              | 10             |
| <b>5.6</b>  | ¿Qué es un análisis de sistema?                   |                |
| <b>5.7</b>  | ¿Qué es un diseño de sistema?                     | 20             |
| <b>5.8</b>  | ¿Qué es un diseño del software?                   | 21             |
| <b>5.9</b>  | ¿Qué es ingeniería del software?                  |                |
| <b>5.10</b> | ¿Qué es un diccionario de datos?                  | 22             |
| <b>5.11</b> | ¿Qué es un diseño arquitectónico?                 | 25             |
| <b>5.12</b> | ¿Qué es un diagrama de flujo de datos?            |                |
| <b>5.13</b> | ¿Qué es una especificación de requisito software? | 26             |

|             |  |    |
|-------------|--|----|
| <b>5.14</b> | ¿Qué es un caso de uso?                              | 27 |
| <b>5.15</b> | ¿Qué es un modelo entidad relación?                  | 29 |
| <b>5.16</b> | Contabilidad.  |    |
| <b>5.17</b> | Sistema Contable                                     |    |
| <b>5.18</b> | Activos  | 30 |
| <b>5.19</b> | Pasivo   | 37 |
| <b>5.20</b> | Balance General                                      | 38 |
| <b>5.21</b> | ¿Qué es la valorización?                             |    |
| <b>5.22</b> | ¿Qué es el patrimonio                                |    |
| <b>5.23</b> | Capital  | 39 |
| <b>5.24</b> | Utilidades acumuladas                                |    |
| <b>5.25</b> | ¿Qué es el estado de ganancia o pérdida?             |    |
| <b>5.26</b> | ¿Cómo se hace el estado de ganancia o pérdida?       | 40 |
| <b>5.27</b> | Que es el flujo de caja o de presupuesto de efectivo |    |
| <b>5.28</b> | Propósito del flujo de caja                          |    |
| <b>5.29</b> | Flujo de caja elementos básicos                      | 41 |
| <b>5.30</b> | Como se prepara el flujo de caja                     |    |
| <b>5.31</b> | Balanza comparativa                                  |    |
| <b>5.32</b> | Balance comparativo                                  |    |
| <b>5.33</b> | Aumentos del Capital contable                        |    |
| <b>5.34</b> | Ingreso  |    |
| <b>5.35</b> | Aportaciones adicionales de capital                  | 42 |

|      |   |    |
|------|---|----|
| 5.36 | Disminuciones del Capital contable                      |    |
| 5.37 | Egresos   |    |
| 5.38 | Reducciones parciales de capital                        |    |
| 5.39 | Determinación del Capital contable final                |    |
| 5.40 | Modificaciones del Activo y del Pasivo aumentan el C. C | 43 |
| 5.41 | Modificaciones del Activo y el Pasivo disminuye el C. C |    |
| 5.42 | Casos en que aumenta el capital contable                |    |
| 5.43 | Casos en que disminuye el capital contable              |    |
| 5.44 | Balance comparativo en forma de cuenta                  | 45 |
| 5.45 | Catálogo de Cuentas                                     |    |
| 5.46 | Formularios   | 46 |
| 5.47 | Estados financieros                                     |    |
| 5.48 | La utilidad de la información contable                  |    |
| 5.49 | Libros de contabilidad                                  |    |
| 5.50 | Bases de Datos  | 47 |
| 5.51 | SQL Server 2008   | 48 |
| 5.52 | Expression Blend:                                       | 49 |
| 5.53 | Project 2007  |    |
| 5.54 | SQL Server Management Studio                            |    |
| 5.55 | Visual Studio 2008                                      | 50 |
| 5.56 | Report Winzar   | 51 |

|              |  |    |
|--------------|--|----|
| <b>5.58</b>  | Report Viewer  |    |
| <b>5.59</b>  | Características de ReportViewer                                      | 52 |
| <b>VI.</b>   | Diseño Metodológico  | 53 |
| <b>VII.</b>  | Cronograma   | 55 |
| <b>VIII.</b> | Recursos y Necesidades   | 57 |
| <b>IX.</b>   | Diccionario de Datos   | 58 |
| <b>X.</b>    | Modelo Conceptual  | 60 |
| <b>XI.</b>   | Diagrama Entidad – Relación (ER)                                     | 61 |
| <b>XII.</b>  | Diseño Arquitectónico  | 62 |
| <b>XIII.</b> | Especificaciones de Requisitos del Software (ERS)                    | 63 |
| <b>13.1</b>  | Introducción   |    |
| <b>13.2</b>  | Descripción general  | 67 |
| <b>13.3</b>  | Requisitos específicos   | 69 |
| <b>13.4</b>  | Requisitos de funcionamiento<br>Restricciones de diseño<br>Atributos | 86 |
| <b>13.5</b>  | Otros requisitos   | 87 |
| <b>XIV.</b>  | Diagrama de Caso de Uso  | 88 |
| <b>XV.</b>   | Diagrama de clases   | 89 |
| <b>XVI.</b>  | Diagramas de Flujo de Datos (DFD)                                    | 90 |
| <b>15.1</b>  | Nivel 0  |    |
| <b>15.2</b>  | Nivel 1  | 91 |

---

|              |  |     |
|--------------|--|-----|
| <b>15.3</b>  | Nivel 2  | 92  |
| <b>XVII.</b> | Codificación   | 94  |
| <b>12.1</b>  | Formulario de acceso al sistema general  |     |
| <b>12.2</b>  | Formulario para la modificacion de datos de una cuenta administrativa existente.                               | 97  |
| <b>12.3</b>  | Formulario para la creacion de una cuenta administrativa.  | 100 |
| <b>12.4</b>  | Presentacion del nombre del sistema con efectos de animacion.  | 102 |
| <b>12.5</b>  | En este formulario master se concentra toda la funcionalidad del sistema SCCSJ.                                | 104 |
| <b>12.6</b>  | El formulario de comprobante se encarga de registrar los movimientos de la empresa y generar reportes.         | 106 |
| <b>12.7</b>  | Formulario para generar el reporte del comprobante de ajuste y perteneciente a la clase Reportes Comprobantes. | 113 |
| <b>12.8</b>  | Formulario para generar el reporte del comprobante de diario y perteneciente a la clase Comprobante Diario.    |     |
| <b>12.9</b>  | Formulario para generar el reporte del comprobante de Egreso y perteneciente a la clase Comprobante Egreso.    | 114 |
| <b>12.10</b> | Formulario para generar el reporte del comprobante de Ingreso y perteneciente a la clase Comprobante Ingreso.  |     |
| <b>12.11</b> | Formulario para el registro del catálogo de cuentas.   | 115 |
| <b>12.12</b> | Formulario para el reporte del Catalogo de cuentas.  | 118 |
| <b>12.13</b> | En este formulario se generan los reportes de los estados financieros de la empresa.                           |     |
| <b>12.14</b> | Vista del menu herramientas del formulario Master.   | 123 |

---

---

|               |  |     |
|---------------|--|-----|
| <b>12.15</b>  | Calculadora del sistema  |     |
| <b>12.16</b>  | Formulario Calendario perteneciente a la clase del mismo nombre y que le permitirá al usuario administrador navegar por diferentes fechas. | 124 |
| <b>12.17</b>  | Vista del menú ayuda del formulario máster.  |     |
| <b>12.18</b>  | Información sobre el SCCSJ   | 125 |
| <b>12.19</b>  | Formulario para crear y restaurar copias de seguridad (BackUp)   |     |
| <b>XVII.</b>  | Anexos   | 127 |
|               | Balance general  |     |
|               | Estado de pérdida o ganancia   |     |
|               | Balanza comparativa  |     |
| <b>XVIII.</b> | Bibliografía   | 131 |

---



## I. INTRODUCCIÓN

Cafetalera San José, S.A., como su nombre lo indica es una empresa que se dedica a la comercialización del café a nivel nacional, actualmente cuenta con un área de contabilidad que registra las operaciones diarias en un sistema contable basado en MS-DOS; Este sistema ya bastante antiguo presenta algunas deficiencias en su manejo, tales como: incompatibilidad con algunos tipos de impresoras, problemas de actualización del sistema, falta de información sobre su funcionamiento e interfaces tediosas y con pocos recursos en su ejecución.

Entre las deficiencias del sistema tenemos incompatibilidad con algunos tipos de impresoras como las modernas, debemos recordar que el sistema es antiguo y los avances tecnológicos crecen a pasos agigantados, en términos reales es un sistema obsoleto a pesar que funciona y da respuesta a la empresa. Los problemas de actualización tiene que ver cuando la empresa solicita alguna mejora al sistema a la persona que desarrolló el software, este le envía por correo o le llega a instalar esa nueva mejora que son en realidad correcciones al sistema, por ejemplo si falla la entrega de reporte. En resumidas cuentas no es muy estable. Por otra parte la falta de información en su funcionamiento esta el hecho de que no hay un apartado en todo el sistema que le indique o ayude al usuario a resolver detalles propios del sistema.

Debido a las dificultades del sistema, la empresa ha optado por la implementación de nuevas tecnologías, que aporten al fortalecimiento y eficiencia de la empresa.

## II. ANTECEDENTES

Cafetalera San José, S.A, nace en el año 1977 con un capital social de cien mil córdobas con acciones nominales de C\$ 1,000.00 córdobas cada una, los socios son seis los cuales comparte igual derechos como obligaciones, la Junta Directiva se reelige cada 4 años. San José, S.A es una empresa Agrícola que se constituyó con el objetivo específico de sembrar café, dejando abierta la posibilidad de dedicarse a cualquier otro campo dentro del sector agropecuario, por ejemplo; Dedicarse a la ganadería.

Cafetalera San José posee una finca rústica ubicada en la ciudad de Matagalpa municipio de La Dalia y consta de 160 Manzanas todas cultivables. La producción promedio de los últimos 3 años ha sido de 3,600 QQ de Café S-H- G (Oro de Exportación).

En los últimos 10 años ha sido administrada por el Lic. Carlos R. Gurdian Debayle quien es socio de la misma.

Hace aproximadamente 8 años la empresa adquirió un sistema contable que actualmente siguen usando, el cual posee algunas interfaces tediosas, nos referimos a que cuando se accede a una interfaz por medio de otra, ya no es posible regresar a esa interfaz por la cual accedimos, lo que obliga al usuario a cerrar la aplicación y volver a iniciar nuevamente; Otro de los inconvenientes es que no se puede usar el Mouse o ratón, solamente el teclado lo que resulta complicado para algunos de sus usuarios. Otro problema es que cuando finaliza un mes y se hace la generación de reportes ya no es posible regresar a meses anteriores para ver los estados financieros, esto quiere decir que no es visible la información de los meses anteriores en el sistema. No se dispone de una opción para el respaldo de la información y su interfaz es rudimentaria.

### III. JUSTIFICACIÓN

Para resolver estos problemas debemos crear un nuevo sistema que mejore en: Elegancia, sencillez en su manejo y métodos de tratamiento y accesibilidad de la información. Aquí no es posible la reingeniería debido a que hay que corregir casi todo y porque la empresa no es dueña del código sólo de la aplicación.

Como se ha dicho anteriormente, el sistema antiguo presenta muchas dificultades para sus usuarios por lo que con el diseño y creación de este nuevo sistema totalmente moderno, la empresa podrá realizar sus operaciones de una manera: Ágil, sin contratiempos, de forma segura y sin complicaciones. Con el uso de las nuevas herramientas de programación que existe en la actualidad trataremos que el sistema que desarrollaremos sea sencillo, pero a la vez potente y estable, con interfaces elegantes, con acceso a la información de forma segura y en el momento solicitado, donde no será necesarios esos engorrosos métodos de actualización, podrá usar todo tipo de impresora sin ningún problema y el usuario podrá estar seguro que su información está a salvo porque incorporaremos técnicas de respaldo de la información como el uso del backup.

La creación de este sistema contará con una base de datos donde se registrará el catálogo de cuentas y las operaciones de la empresa en el libro auxiliar; Además, generará estados financieros (Balance General, Estado de Perdida o Ganancia y Balanza Comparativa) e informes.

Este proyecto nos servirá para hacernos una autoevaluación de nuestros conocimientos adquiridos durante todos estos años de estudios, a la vez será un soporte que utilizaremos como plataforma para lanzarnos al campo laboral.

## IV. OBJETIVOS

### 1. Objetivo General:

- ❖ Diseñar y crear un sistema contable para la empresa: Cafetalera San José, S.A.

### 2. Objetivos Específicos:

- ❖ Implementar niveles de seguridad en el tratamiento de la información, como el uso de backup.
- ❖ Implementar el uso de interfaces que sean fáciles de usar y con una apariencia agradable.
- ❖ Generar los estados financieros de forma exacta tales como: Balance General, Balanza comparativa y Estado de Pérdida o Ganancia.

## V. MARCO TEÓRICO

Es necesario tener bien claro algunos términos que ampliaremos durante el desarrollo del proyecto para una mejor comprensión:

### 5.1 ¿Qué es un sistema?

Es un conjunto de componentes que interactúan entre sí, para lograr un objetivo común. Un sistema basado en computadora hace uso de varios elementos del sistema (Software, Hardware, Bases de Datos, Procedimientos, etc.).

### 5.2 ¿Sistema informático?

Un sistema es un conjunto de componentes que interaccionan entre sí para lograr un objetivo común. Aunque existe una gran variedad de sistemas, la mayoría de ellos pueden representarse a través de un modelo formado por cinco bloques básicos: Elementos de entrada, elementos de salida, sección de transformación, mecanismos de control y objetivos.

En la actualidad, la expresión sistema de información se utiliza de forma común y habitual en las organizaciones; sin embargo, existen tantas definiciones y matrices para ella como escuelas o autores del tema. Aun así, y basándonos en la definición dada de sistema, se podrá realizar una primera aproximación definiéndola como un conjunto de componentes que interaccionan entre sí para lograr un objetivo común: Satisfacer las necesidades de información de una organización.

Los autores Laudon y Laudon (2004) definen los sistemas de información como un conjunto de componentes interrelacionados que recolectan (o recuperan), procesan, almacenan y distribuyen información para apoyar la toma de decisiones y el control de una organización. Además de apoyar la toma de decisiones, la coordinación y el control, los sistemas de información también pueden ayudar a los gerentes y trabajadores a analizar problemas, a visualizar asuntos complejos y a crear productos nuevos.

La definición proporcionada por Laudon y Laudon (2004) refleja tres aspectos básicos de los sistemas de información:

Un sistema de información está formado por un conjunto de componentes. A diferencia de otras definiciones que analizaremos más adelante, los autores no especifican que componentes interactúan en el sistema de información. Esto es debido a querer englobar los distintos sistemas de información en una única definición. En contraposición a lo que la mayoría de personas creen, un sistema de información puede ser formal e informal. De forma similar, nos podemos encontrar con sistemas de información basados en ordenadores (o en la tecnología de la información), y sistemas de información que utilizan la tecnología del papel y el lápiz.

Los sistemas formales de información son aquellos que se apoyan en definiciones fijas y aceptadas de datos y procedimientos que operan en conformidad con reglas predefinidas, mientras que los sistemas informales de información se basan en reglas de comportamiento no establecidas. Las conservaciones de trabajo en la máquina de café, o una reunión durante la comida pueden considerarse sistemas informales.

El segundo aspecto que trata la definición de Laudon y Laudon son las actividades que realizan los sistemas de información, más concretamente el sistema de transformación de un sistema. Tal y como queda reflejado en la definición de Laudon y Laudon, así como en las posteriores definiciones que se presentaran a continuación, las actividades básicas de un sistema de información son la recolección (o recuperación), el procesado, el almacenamiento y la distribución de la información introducida. En función del nivel de complejidad del procesado, se obtendrán distintos sistemas de información. En apartados posteriores, se expondrán diversas clasificaciones de sistemas de información en función del tipo de procesado de datos.

El siguiente y último punto en la definición de Laudon y Laudon (2004) explicita la utilidad y las funciones de un sistema de información: (1) apoyar la toma de decisiones y (2) el control de una organización.

El determinismo de la información se refiere a la predecibilidad del orden y del tiempo de llegada de los datos. Un programa de análisis de ingeniería acepta datos que están en un orden predefinido, ejecuta el algoritmo(s) de análisis sin interrupción y produce los datos resultantes en un informe o formato gráfico. Se dice que tales aplicaciones son determinadas.

Un sistema operativo multiusuario, por otra parte, acepta entradas que tienen un contenido variado y que se producen en instantes arbitrarios, ejecuta algoritmos que pueden ser interrumpidos por condiciones externas y produce una salida que depende de una función del entorno y del tiempo. Las aplicaciones con estas características se dice que son indeterminadas.

Algunas veces es difícil establecer categorías genéricas para las aplicaciones del software que sean significativas. Conforme aumenta la complejidad del Software, es más difícil establecer compartimentos nítidamente separados. Las siguientes áreas del software indican la amplitud de las aplicaciones potenciales:

### **5.3 Aplicaciones del software**

El software puede aplicarse en cualquier situación en la que se haya definido previamente un conjunto específico de pasos procedimentales (es decir, un algoritmo) (excepciones notables a esta regla son el software de los sistemas expertos y de redes neuronales). El contenido y el determinismo de la información son factores importantes a considerar para determinar la naturaleza de una aplicación de software. El contenido se refiere al significado y a la forma de la información de entrada y salida. Por ejemplo, muchas aplicaciones bancarias usan unos datos de entrada muy estructurados (una base de datos) y producen «informes» con determinados formatos. El software que controla una máquina automática (por ejemplo: un control numérico) acepta elementos de datos discretos con una estructura limitada y produce órdenes concretas para la máquina en rápida sucesión.

#### **5.3.1 Software de sistemas**

El software de sistemas es un conjunto de programas que han sido escritos para servir a otros programas. Algunos programas de sistemas (por ejemplo: compiladores, editores y utilidades de gestión de archivos) procesan estructuras de información complejas pero determinadas. Otras aplicaciones de sistemas (por ejemplo: ciertos componentes del sistema operativo,

utilidades de manejo de periféricos, procesadores de telecomunicaciones) procesan datos en gran medida indeterminados. En cualquier caso, el área del software de sistemas se caracteriza por una fuerte interacción con el hardware de la computadora; una gran utilización por múltiples usuarios; una operación concurrente que requiere una planificación, una compartición de recursos y una sofisticada gestión de procesos; unas estructuras de datos complejas y múltiples interfaces externas.

### 5.3.2 Software de tiempo real

El software que coordina, analiza y controla sucesos del mundo real conforme ocurren, se denomina de tiempo real. Entre los elementos del software de tiempo real se incluyen: un componente de adquisición de datos que recolecta y da formato a la información recibida del entorno externo, un componente de análisis que transforma la información según lo requiera la aplicación, un componente de control/salida que responda al entorno externo, y un componente de monitorización que coordina todos los demás componentes, de forma que pueda mantenerse la repuesta en tiempo real (típicamente en el rango de un milisegundo a un segundo).

### 5.3.3 Software de gestión.

El proceso de la información comercial constituye la mayor de las áreas de aplicación del software. Los «sistemas» discretos (por ejemplo: nóminas, cuentas de haberes-débitos, inventarios, etc.) han evolucionado hacia el software de sistemas de información de gestión (SIG) que accede a una o más bases de datos que contienen información comercial.

Las aplicaciones en esta área reestructuran los datos existentes para facilitar las operaciones comerciales o gestionar la toma de decisiones. Además de las tareas convencionales de procesamientos de datos, las aplicaciones de software de gestión también realizan cálculo interactivo (por ejemplo: el procesamiento de transacciones en puntos de ventas).

### 5.3.4 Software de ingeniería y científico.

El software de ingeniería y científico está caracterizado por los algoritmos de «manejo de números». Las aplicaciones van desde la astronomía a la vulcanología, desde el análisis de la presión de los automotores a la dinámica orbital de las lanzaderas espaciales y desde la biología molecular a la fabricación automática. Sin embargo, las nuevas aplicaciones del área de ingeniería/ciencia se han alejado de los algoritmos convencionales numéricos. El diseño asistido por computadora (del inglés CAD), la simulación de sistemas y otras aplicaciones interactivas, han comenzado a coger características del software de tiempo real e incluso del software de sistemas.

### 5.3.5 Software empotrado.

Los productos inteligentes se han convertido en algo común en casi todos los mercados de consumo e industriales. El software empotrado reside en memoria de sólo lectura y se utiliza para controlar productos y sistemas de los mercados industriales y de consumo. El software empotrado puede ejecutar funciones muy limitadas y curiosas (por ejemplo: el control de las teclas de un horno de microondas) o suministrar una función significativa y con capacidad de control (por ejemplo: funciones digitales en un automóvil, tales como control de la gasolina, indicadores en el salpicadero, sistemas de frenado, etc.).

## 5.4 ¿Qué es un usuario?

Son personas o individuos de una organización, que interactúan entre sí para lograr un objetivo en común.

### 5.4.1 Tipos de usuario informático

En sentido general, un usuario es un conjunto de permisos y de recursos (o dispositivos) a los cuales se tiene acceso. Es decir, un usuario puede ser tanto una persona como una máquina, un programa, etc.

### 5.4.2 Usuario final

El usuario final de un producto informático (bien sea hardware o software), es la persona a la que va destinada dicho producto una vez que ha superado las fases de desarrollo correspondientes.

Normalmente, el software se desarrolla pensando en la comodidad del usuario final, y por esto se presta especial interés y esfuerzo en conseguir una interfaz de usuario lo más clara y sencilla posible.

Usuario final no es necesariamente sinónimo de cliente o comprador, una compañía puede ser un importante comprador de software, pero el utilizador final puede ser solamente un empleado o grupo de empleados dentro de la compañía, como una secretaria o un capturista. El concepto clave es la interacción directa con el programa, no la propiedad.

### 5.5 ¿Qué es un análisis?

Es el estudio de un problema antes de realizar una acción.

El análisis es una de las etapas del ciclo de vida de un sistema informático.

En esta etapa los analistas se encargan de analizar los requerimientos del sistema (a veces se incluye en esta etapa la recolección de requerimientos también).

Esta etapa centra su atención en la interacción de los usuarios con el sistema.

En el análisis estructurado, el resultado de la etapa de análisis es el Modelo Esencial.

### 5.6 ¿Qué es un análisis de sistema?

Es el proceso de clasificación e interpretación de hechos, diagnósticos de problema y empleo de la información para recomendar mejoras al sistema.

El Análisis de Sistemas trata básicamente de determinar los objetivos y límites del sistema objeto de análisis, caracterizar su estructura y funcionamiento, marcar las directrices que permitan alcanzar los objetivos propuestos y evaluar sus consecuencias. Dependiendo de los objetivos del análisis, podemos encontrarnos ante dos problemáticas distintas:

- ❖ Análisis de un sistema ya existente para comprender, mejorar, ajustar y/o predecir su comportamiento.
- ❖ Análisis como paso previo al diseño de un nuevo sistema-producto.

En cualquier caso, podemos agrupar más formalmente las tareas que constituyen el análisis en una serie de etapas que se suceden de forma iterativa hasta validar el proceso completo:

#### **5.6.1 Conceptualización.**

Consiste en obtener una visión de muy alto nivel del sistema, identificando sus elementos básicos y las relaciones de éstos entre si y con el entorno.

#### **5.6.2 Análisis funcional.**

Describe las acciones o transformaciones que tienen lugar en el sistema. Dichas acciones o transformaciones se especifican en forma de procesos que reciben unas entradas y producen unas salidas.

#### **5.6.3 Análisis de condiciones.**

Debe reflejar todas aquellas limitaciones impuestas al sistema que restringen el margen de las soluciones posibles. Estas se derivan a veces de los propios objetivos del sistema:

- ❖ Operativas, como son las restricciones físicas, ambientales, de mantenimiento, de personal, de seguridad, etc.
- ❖ De calidad, como fiabilidad, mantenibilidad, seguridad, convivencia, generalidad, etc.

Sin embargo, en otras ocasiones las constricciones vienen impuestas por limitaciones en los diferentes recursos utilizables:

- ❖ Económicos, reflejados en un presupuesto.
- ❖ Temporales, que suponen unos plazos a cumplir.
- ❖ Humanos.
- ❖ Metodológicos, que conllevan la utilización de técnicas determinadas.
- ❖ Materiales, como espacio, herramientas disponibles, etc.

#### **5.6.4 Construcción de modelos.**

Una de las formas más habituales y convenientes de analizar un sistema consiste en construir un prototipo (un modelo en definitiva) del mismo.

### 5.6.5 Validación del análisis.

A fin de comprobar que el análisis efectuado es correcto y evitar, en su caso, la posible propagación de errores a la fase de diseño, es imprescindible proceder a la validación del mismo. Para ello hay que comprobar los extremos siguientes:

- ❖ El análisis debe ser consistente y completo.
- ❖ Si el análisis se plantea como un paso previo para realizar un diseño, habrá que comprobar además que los objetivos propuestos son correctos y realizables.

Una ventaja fundamental que presenta la construcción de prototipos desde el punto de vista de la validación radica en que estos modelos, una vez construidos, pueden ser evaluados directamente por los usuarios o expertos en el dominio del sistema para validar sobre ellos el análisis.

### 5.6.6 Diseño estructurado.

Los métodos de diseño del software se obtienen del estudio de cada uno de los tres dominios del modelo de análisis. El dominio de los datos, el funcional y el de comportamiento sirven de directriz para la creación del diseño.

En el diseño estructurado orientado al flujo de datos, partimos de la representación del flujo de la información obtenida en la fase de análisis, donde la información puede representarse como un flujo continuo que sufre una serie de transformaciones conforme va de la entrada a la salida.

El diagrama de flujo de datos DFD (o de burbujas) se utiliza como herramienta gráfica para la descripción del flujo de la información.

En programación y diseño de algoritmos, el diseño estructurado persigue elaborar algoritmos que cumplan la propiedad de modularidad, para ello, dado un problema que se pretende resolver mediante la elaboración de un programa de ordenador, se busca dividir dicho programa en módulos siguiendo los principios de diseño de Descomposición por refinamientos sucesivos, creación de una Jerarquía modular y elaboración de módulos Independientes.

### 5.6.7 Etapas del diseño estructurado.

#### ❖ Descomposición

¿Por qué descomponer un problema en partes? Experimentalmente está comprobado que:

- ❖ Un problema complejo cuesta más de resolver que otro más sencillo.
- ❖ La complejidad de un problema global es mayor que el valor de las complejidades de cada una de sus partes por separado.

Según esto, merece la pena el esfuerzo de dividir un problema grande en subproblemas más pequeños. Si el objetivo es elaborar un programa para resolver dicho problema *grande*, cada subproblema (menos complejo) podrá ser resuelto por un módulo (subalgoritmo) relativamente fácil de implementar (más que el programa global **No** dividido).

Ahora la cuestión es *¿cómo realizar la descomposición?*; realizando un estudio descendente Top-Down que nos lleve desde la concepción del problema (programa o algoritmo) global hasta identificar sus partes (módulos). Esta técnica se repite aplicando una estrategia llamada de refinamiento sucesivo propuesta por el experto en Ciencias de la Computación **Niklaus Wirth**, que consiste precisamente en volver a aplicar el estudio descendente Top-Down a cada subproblema una y otra vez hasta obtener subproblemas suficientemente pequeños, que puedan ser resueltos por módulos que cumplan, en la medida de lo posible, las características deseables en un módulo en el ámbito de la programación. En palabras del propio **Niklaus Wirth**:

- ❖ En cada paso (del refinamiento), una o varias instrucciones del programa dado, se descomponen en instrucciones más detalladas. Esta descomposición sucesiva o refinamiento de especificaciones termina cuanto todas las instrucciones están expresadas en términos de la computadora usada o del lenguaje de programación...

- ❖ Conforme se refinan las tareas, también los datos pueden ser refinados, descompuestos o estructurados, siendo lo natural refinar las especificaciones del programa y de los datos en paralelo.
- ❖ Cada paso de refinamiento implica algunas decisiones de diseño. Es importante que el programador sea consciente de los criterios subyacentes (en las decisiones de diseño adoptadas) y de la existencia de soluciones alternativas...

#### 5.6.8 Problema del refinamiento sucesivo.

¿Cuándo parar el refinamiento?. Un refinamiento excesivo podría dar lugar a un número tan grande de módulos que haría poco práctica la descomposición. Se tendrán en cuenta estos criterios para dejar de descomponer:

- ❖ Cuando no haya tareas bien definidas.
- ❖ Cuando la interfaz de un módulo sea tan complicada como el propio módulo.

#### 5.6.9 Jerarquía de módulos.

Ésta es una consecuencia directa de la descomposición del problema mediante **refinamientos sucesivos**, el resultado será un conjunto de módulos estratificados en capas a modo de pirámide donde en la cima habrá un único módulo que representará al programa global y en los niveles inferiores aparecerán los módulos resultantes de las sucesivas divisiones.

Al final, debe obtenerse una estructura piramidal donde los módulos de los niveles superiores se encargan de las tareas de coordinación, lógica de la aplicación y manipulación de los módulos inferiores; estos otros deberán realizar tareas de cálculo, tratamiento y entrada/salida de información.

#### 5.6.10 Evaluando el diseño.

Para evaluar o determinar como de bueno es un diseño estructurado se utilizan los conceptos de **acoplamiento** y **cohesión**; éstos están muy relacionados entre sí, tanto que difícilmente se puede variar uno sin que eso afecte al otro. También cabe decir que estos conceptos no son medidas que se puedan cuantificar numéricamente, son más bien magnitudes cualitativas. También se tienen en consideración los conceptos de **fan-in** y **fan-out**.

### 5.6.11 Acoplamiento.

Se define como el **grado de interdependencia que hay entre los distintos módulos de un programa**; lo deseable es que esta interdependencia sea lo menor posible, es decir, un bajo acoplamiento. Los niveles de acoplamiento, ordenados de menor (más deseable) a mayor (menos deseable) son:

#### ❖ **Acoplamiento normal.**

Un módulo llama a otro de un nivel inferior y tan solo intercambian datos (parámetros de entrada/salida). Dentro de este tipo de acoplamiento podemos encontrarlos 3 subtipos, dependiendo de los datos que intercambien los módulos:

- ❖ **Acoplamiento de datos:** Los módulos se comunican mediante parámetros.
- ❖ **Acoplamiento de marca o por estampado:** Los módulos se pasan datos con estructura de registro. No es muy deseable si el módulo receptor sólo requiere parte de los datos que se le pasan.
- ❖ **Acoplamiento de control:** Los datos que se intercambian entre los módulos son controles. Debido a que en este subtipo un módulo controla la ejecución del otro, no es un buen acoplamiento, ya que impide que sean totalmente independientes.
- ❖ **Acoplamiento Común:** Dos módulos acceden a un mismo recurso común, típicamente memoria compartida, una variable global o un fichero. Una variante de este tipo de acoplamiento es el acoplamiento externo:
- ❖ **Acoplamiento externo:** Los módulos están ligados a componentes externos. Por ejemplo, dispositivos de E/S, protocolos de comunicaciones... etc.
- ❖ **Acoplamiento de contenido:** Ocurre cuando un módulo necesita acceder a una parte de otro módulo.

### 5.6.12 Cohesión.

Se define como la **medida de fuerza o relación funcional existente entre las sentencias o grupos de sentencias de un mismo módulo**. Un módulo **coherente** ejecutará una única tarea sencilla interactuando muy poco o nada

con el resto de módulos del programa. Se persigue que los módulos tengan una alta cohesión.

En el diseño estructurado podemos encontrarnos con los siguientes 7 tipos de cohesión (de la mejor o más deseable a la menos recomendable):

- ❖ **Cohesión funcional:** Los elementos del módulo están relacionados en el desarrollo de una única función.
- ❖ **Cohesión secuencial:** Un módulo realiza distintas tareas en secuencia, de forma que las entradas de cada tarea son las salidas de la tarea anterior. No es una mala cohesión si las tareas implicadas no son muy complejas y requieren pocas líneas de código.
- ❖ **Cohesión comunicacional:** El módulo realiza actividades paralelas usando los mismos datos de entrada y salida. Como en el caso anterior, tampoco se trata de un mal tipo de cohesión si las tareas son relativamente sencillas.
- ❖ **Cohesión procedimental:** El módulo tiene una serie de funciones relacionadas por un procedimiento efectuado por el código (a modo de librería). Es similar a la secuencial, pero puede incluir el paso de controles. Será deseable que las funciones estén relacionadas o realicen tareas dentro del mismo ámbito (p.e. la librería `string.h` de C contienen funciones para operar con cadenas de caracteres).
- ❖ **Cohesión temporal:** Los elementos del módulo están implicados en actividades relacionadas con el tiempo.
- ❖ **Cohesión lógica:** Las actividades que realiza el módulo tienen la misma categoría. Esto es, es como si se tuvieran partes independientes dentro del mismo módulo.
- ❖ **Cohesión casual o coincidente:** Los elementos del módulo contribuyen a las actividades relacionándose mutuamente de una manera poco significativa. Este tipo de cohesión viola el principio de independencia y de caja negra de los módulos.

#### 5.6.13 Fan-In y Fan-Out.

Además de los dos conceptos anteriores, se deben tener en cuenta el **grado de absorción (fan-in)** y la **diseminación del control (fan-out)** de los módulos para garantizar la calidad del diseño.

- ❖ **Fan-In:** También llamado **grado de absorción**. Es el número de subordinados inmediatos que tiene el módulo en cuestión. Es conveniente maximizar el fan-in durante el proceso de diseño, ya que cada instancia de fan-in múltiple indica que se ha evitado la duplicación de código.
- ❖ **Fan-Out:** También llamado **diseminación del control**. Es el número de subordinados inmediatos que tiene el módulo en cuestión. Conviene no tener un fan-out ni muy alto ni muy bajo, ya que eso es un posible indicador de un diseño pobre. Si no es posible evitarlo, es preferible un fan-out bajo antes que uno alto.

#### 5.6.14 **Diseño de datos.**

El impacto de la estructura de datos sobre la estructura del programa y la complejidad procedimental hace que el diseño de datos tenga una gran influencia en la calidad del software. Los datos bien diseñados pueden conducir a una mejor estructura de programa, a una modularidad efectiva y a una complejidad procedimental reducida.

#### 5.6.15 **El ciclo de vida del desarrollo del sistema.**

El SDLC (SDLC, Systems Development LifeCycle) es un enfoque por fases para el análisis y el diseño cuya premisa principal consiste en que los sistemas se desarrollan mejor utilizando un ciclo específico de actividades del analista y el usuario.

Los analistas no se ponen de acuerdo en la cantidad de fases que incluye el ciclo de vida del desarrollo de sistemas, pero en general alaban su enfoque organizado. A pesar de que cada fase se explica por separado, nunca se realiza como un paso aislado. Más bien, es posible que varias actividades ocurran de manera simultánea, y algunas de ellas podrían repetirse. Es más práctico considerar que el SDLC se realiza por fases (con actividades en pleno apogeo que se traslapan con otras hasta terminarse por completo) y no en pasos aislados.

#### 5.6.16 **Identificación del problema, oportunidades y objetivo.**

En esta primera fase del ciclo de vida del desarrollo de sistemas, el analista se ocupa de identificar problemas, oportunidades y objetivos. Esta etapa es crítica para el éxito del resto del proyecto, pues a nadie le agrada desperdiciar tiempo trabajando en un problema que no era el que se debía resolver.

La primera fase requiere que el analista observe objetivamente lo que sucede en un negocio. A continuación, en conjunto con otros miembros de la organización, el analista determina con precisión cuáles son los problemas. Con frecuencia los problemas son detectados por alguien más, y ésta es la razón de la llamada inicial al analista. Las oportunidades son situaciones que el analista considera susceptibles de mejorar utilizando sistemas de información computarizados. El aprovechamiento de las oportunidades podría permitir a la empresa obtener una ventaja competitiva o establecer un estándar para la industria.

La identificación de objetivos también es una parte importante de la primera fase. En primer lugar, el analista debe averiguar lo que la empresa trata de conseguir. A continuación, podrá determinar si algunas funciones de las aplicaciones de los sistemas de información pueden contribuir a que el negocio alcance sus objetivos aplicándolas a problemas u oportunidades específicos.

Los usuarios, los analistas y los administradores de sistemas que coordinan el proyecto son los involucrados en la primera fase. Las actividades de esta fase consisten en entrevistar a los encargados de coordinar a los usuarios, sintetizar el conocimiento obtenido, estimar el alcance del proyecto y documentar los resultados. El resultado de esta fase es un informe de viabilidad que incluye una definición del problema y un resumen de los objetivos. A continuación, la administración debe decidir si se sigue adelante con el proyecto propuesto. Si el grupo de usuarios no cuenta con fondos suficientes, si desea atacar problemas distintos, o si la solución a estos problemas no amerita un sistema de cómputo, se podría sugerir una solución diferente y el proyecto de sistemas se cancelaría.

### 5.6.17 Determinación de los requisitos de información.

Entre las herramientas que se utilizan para determinar los requerimientos de información de un negocio se encuentran métodos interactivos como las entrevistas, los muestreos, la investigación de datos impresos y la aplicación de cuestionarios; métodos que no interfieren con el usuario como la observación del comportamiento de los encargados de tomar las decisiones y sus entornos de oficina, al igual que métodos de amplio alcance como la elaboración de prototipos.

El desarrollo rápido de aplicaciones (RAD, Rapid Application Development) es un enfoque orientado a objetos para el desarrollo de sistemas que incluye un método de desarrollo (que abarca la generación de requerimientos de información) y herramientas de software.

En la fase de determinación de los requerimientos de información del SDLC, el analista se esfuerza por comprender la información que necesitan los usuarios para llevar a cabo sus actividades. Como puede ver, varios de los métodos para determinar los requerimientos de información implican interactuar directamente con los usuarios. Esta fase es útil para que el analista confirme la idea que tiene de la organización y sus objetivos. En ocasiones sólo realizan las dos primeras fases del ciclo de vida del desarrollo de sistemas. Esta clase de estudio podría tener un propósito distinto y por lo general la lleva a la práctica un especialista conocido como analista de información (IA, Information Analysis).

Los implicados en esta fase son el analista y los usuarios, por lo general trabajadores y gerentes del área de operaciones. El analista de sistemas necesita conocer los detalles de las funciones del sistema actual: el quién (la gente involucrada), el qué (la actividad del negocio), el dónde (el entorno donde se desarrollan las actividades), el cuándo (el momento oportuno) y el cómo (la manera en que se realizan los procedimientos actuales) del negocio que se estudia. A continuación el analista debe preguntar la razón por la cual se utiliza el sistema actual. Podría haber buenas razones para realizar los negocios con los métodos actuales, y es importante tomarlas en cuenta al diseñar un nuevo sistema.

Sin embargo, si la razón de ser de las operaciones actuales es que "siempre se han hecho de esta manera", quizá será necesario que el analista mejore los procedimientos. La reingeniería de procesos de negocios podría ser útil para

conceptualizar el negocio de una manera creativa. Al término de esta fase, el analista debe conocer el funcionamiento del negocio y poseer información muy completa acerca de la gente, los objetivos, los datos y los procedimientos implicados.

### 5.7 ¿Qué es un diseño de sistema?

Es el proceso de planificar, reemplazar o complementar un sistema organizacional existente.

El Diseño de Sistemas se ocupa de desarrollar las directrices propuestas durante el análisis en función de aquella configuración que tenga más posibilidades de satisfacer los objetivos planteados tanto desde el punto de vista funcional como del no funcional (lo que antes hemos denominado constricciones). El proceso de diseño de un sistema complejo se suele realizar de forma descendente:

- A. Diseño de alto nivel (o descomposición del sistema a diseñar en subsistemas menos complejos).
- B. Diseño e implementación de cada uno de los subsistemas:
  - ❖ Especificación consistente y completa del subsistema de acuerdo con los objetivos establecidos en el análisis.
  - ❖ Desarrollo según la especificación.
  - ❖ Prueba.
- C. Integración de todos los subsistemas.
- D. Validación del diseño.

Dentro del proceso de diseño de sistemas hay que tener en cuenta los efectos que pueda producir la introducción del nuevo sistema sobre el entorno en el que deba funcionar, adecuando los criterios de diseño a las características del mismo. En este contexto está adquiriendo una importancia creciente la adaptación de todo sistema-producto a las capacidades de las personas que van a utilizarlo, de forma que su operación sea sencilla, cómoda, efectiva y eficiente.

De estas cuestiones se ocupa una disciplina, la Ergonomía, que tiene por objeto la optimización de los entornos hombre-máquina. Si bien en un principio estaba centrada en los aspectos antropométricos de la relación hombre-máquina, en la actualidad ha pasado a intervenir con fuerza en todos

los procesos cognitivos (análisis, interpretación, decisión, comunicación y representación del conocimiento). Así, con respecto al diseño de herramientas *software*, la ergonomía tiene mucho que decir en cuestiones relacionadas con

la disposición de informaciones en pantalla, profundidad de menús, formato de iconos, nombres de comandos, control de cursores, tiempos de respuesta, manejo de errores, estructuras de datos, utilización de lenguaje natural, etc.

### 5.8 ¿Qué es un diseño del software?

Puede definirse como el proceso de aplicar distintas técnicas y principios con el propósito de definir un sistema. El diseño es el proceso en que se asienta la calidad del software y es la única forma mediante la cual podemos traducir con precisión los requisitos de cliente en un producto o sistema acabado.

### 5.9 ¿Qué es ingeniería del software?

El término ingeniería de software abarca al grupo de métodos, técnicas y herramientas que se utilizan en la producción del software, más allá de la actividad principal de programación.

El término "ingeniería" es una referencia directa a la ingeniería civil, una referencia al estudio de la construcción. En programación se aplica el mismo principio que en la construcción de un edificio: poner simplemente ladrillos y cemento no es suficiente. La construcción de un edificio consta de diversos pasos antes de comenzar con la fase de construcción, tales como el diseño arquitectónico, la albañilería, la fontanería, el diseño eléctrico, y durante este período se calculan los presupuestos y los plazos.

Por lo tanto, la ingeniería de software requiere la gestión de proyectos para que se pueda desarrollar una aplicación en el plazo previsto y con el presupuesto establecido que sea satisfactoria para el cliente (el concepto de calidad).

### 5.10 ¿Qué es un diccionario de datos?

El diccionario de datos es una aplicación especializada de los tipos de diccionarios usados como referencia en la vida cotidiana. El diccionario de datos es una obra de consulta con información acerca de los datos (es decir, metadatos), compilada por los analistas de sistemas para guiarse en el análisis y diseño. Como un documento, el diccionario de datos recopila y coordina términos de datos específicos, y confirma lo que cada término significa para las diferentes personas en la organización.

Una razón importante para mantener un diccionario de datos es guardar datos ordenados. Esto significa que los datos deben, ser consistentes. Si usted guarda datos acerca del sexo de un hombre como "M" en un registro, "Masculino" en un segundo registro y como el número "1" en un tercer registro, los datos no son consistentes. Un diccionario de datos ayudará en este aspecto.

#### 5.10.1 Necesidad de entender el diccionario de datos.

Muchos sistemas de administración de base de datos están equipados con un diccionario de datos automatizado. Estos diccionarios pueden ser complejos o sencillos. Algunos diccionarios de datos computarizados catalogan automáticamente los elementos de datos cuando se hace la programación; otros simplemente proporcionan una plantilla para motivar a la persona que llene el diccionario a que lo haga de una manera uniforme para cada entrada.

A pesar de la existencia de los diccionarios de datos automatizados, entender qué datos conforman un diccionario de datos, las convenciones usadas en estos últimos y cómo se desarrolla un diccionario de datos, son problemas que el analista de sistemas debe tener siempre presentes durante el esfuerzo de sistemas. Entender el proceso de compilar un diccionario de datos puede ayudar al analista de sistemas a visualizar el sistema y su funcionamiento. Las

Próximas secciones permiten al analista de sistemas ver la lógica detrás de lo que existe tanto en los diccionarios automatizados como en los manuales.

Además de proporcionar documentación y eliminar la redundancia, el diccionario de datos se podría usar para:

- ❖ Validar la integridad y exactitud del diagrama de flujo de datos.
- ❖ Proporcionar un punto de partida para desarrollar pantallas e informes.
- ❖ Determinar el contenido de los datos almacenados en archivos.
- ❖ Desarrollar la lógica para los procesos del diagrama de flujo de datos.

#### 5.10.2 Definición de los flujos de datos.

Por lo general, los flujos de datos son los primeros elementos que se definen. Las entradas y salidas del sistema se determinan mediante las entrevistas y la observación de los usuarios, y el análisis de documentos y de otros sistemas existentes. La información capturada para cada flujo de datos se podría resumir usando un formulario que contenga la siguiente información:

- a) ID, un número de identificación opcional. A veces éste se codifica usando un esquema para identificar el sistema y la aplicación del sistema.
- b) Un solo nombre descriptivo para este flujo de datos. Este nombre es el texto que debe aparecer en el diagrama y se debe referenciar en todas las descripciones que usen el flujo de datos.
- c) Una descripción general del flujo de datos.
- d) La fuente del flujo de datos. Ésta podría ser una entidad externa, un proceso o un flujo de datos proveniente de un almacén de datos.
- e) El destino del flujo de datos (los mismos elementos que se describieron en la fuente).
- f) Algo que indique si el flujo de datos es un registro que está entrando o saliendo de un archivo o un registro que contiene un informe, formulario o pantalla. Si el flujo de datos contiene datos que se usan entre los procesos, se designa como interno.

- g) El nombre de la estructura de datos que describe los elementos encontrados en este flujo de datos. Para un flujo de datos simple, podrían ser uno o varios elementos.
- h) El volumen por unidad de tiempo. Los datos podrían ser registros por día o cualquier otra unidad de tiempo.
- i) Un área para comentarios adicionales y anotaciones sobre el flujo de datos.

### 5.10.3 Descripción de las estructuras de datos.

Normalmente las estructuras de datos se describen usando una notación algebraica. Este método permite al analista producir una vista de los elementos que constituyen la estructura de datos junto con información referente a dichos elementos. Por ejemplo, el analista indicará si hay muchos elementos iguales en la estructura de datos (un grupo de repetición), o si dos elementos podrían excluirse mutuamente. La notación algebraica usa los siguientes símbolos:

- ❖ Un signo de igual (=) significa "está compuesto de".
- ❖ Un signo de suma (+) significa "y".
- ❖ Las llaves { } indican elementos repetitivos, también llamados grupos de repetición o tablas. En el grupo podría haber un elemento de repetición o varios de ellos. El grupo de repetición podría tener condiciones, tal como un número fijo de repeticiones o límites superiores e inferiores para el número de repeticiones.
- ❖ Los corchetes [ ] representan una situación de uno u otro. Se podría representar un elemento u otro, pero no ambos. Los elementos listados entre los corchetes son mutuamente excluyentes.
- ❖ Los paréntesis ( ) representan un elemento opcional. Los elementos opcionales se podrían dejar en blanco en la entrada de las pantallas y podrían contener espacios o ceros para campos numéricos en las estructuras de archivos.

### 5.11 ¿Qué es un diseño arquitectónico?

El diseño arquitectónico es un proceso creativo en el que se intenta establecer una organización del sistema que satisfaga los requerimientos funcionales y no funcionales del propio sistema. Debido a que es un proceso creativo, las actividades dentro del proceso difieren radicalmente dependiendo del tipo de sistema a desarrollar, el conocimiento y la experiencia del arquitecto del sistema, y los requerimientos específicos del mismo.

El objetivo principal del diseño arquitectónico es desarrollar una estructura de programa modular y representar las relaciones de control entre los módulos. Mezcla la estructura de programas y la estructura de datos y define las relaciones que facilitan el flujo de los datos a lo largo del programa.

El diseño orientado al flujo de datos es compatible con un amplio rango de áreas de aplicación. Es particularmente útil cuando se procesa secuencialmente la información y no existe ninguna estructura jerárquica formal. De hecho, todo el software puede representarse como un diagrama de flujo de datos. Ejemplo: aplicaciones con microprocesadores, procedimientos de análisis numérico, procesos de control, etc.

### 5.12 ¿Qué es un diagrama de flujo de datos?

Un diagrama de flujos de datos (DFD), es una técnica grafica que describe el flujo de información y las transformaciones que se aplican a los datos, conforme se mueven de la entrada a la salida.

A medida que la información se mueve a través del software, es modificada por una serie de transformaciones.

El 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. El DFD es también conocido como grafo de flujo de datos o como diagrama de burbujas.

Se puede usar el diagrama de flujo de datos para representar un sistema o un software a cualquier nivel de abstracción. De hecho, los DFDs pueden ser divididos en niveles que representen un mayor flujo de información y un mayor detalle funcional. Por consiguiente, el DFD proporciona un mecanismo para el modelado funcional, así como el modelado del flujo de información.

Un DFD de nivel 0, también denominado modelo fundamental del sistema o modelo de contexto, representa al elemento de software completo como una sola burbuja con datos de entrada y de salida representados por flechas de entrada y de salida, respectivamente. Al dividir el DFD de nivel 0 para mostrar más detalles, aparecen representados procesos (burbujas) y caminos de flujo de información adicionales. Por ejemplo, un DFD de nivel 1 puede contener cinco o seis burbujas con flechas interconectadas. Cada uno de los procesos representados en el nivel 1 es una subfunción del sistema general en el modelo del contexto.

La notación básica que se usa para desarrollar un DFD no es en sí misma suficiente para describir los requisitos del software. Por ejemplo, una flecha de un DFD representa un objeto de datos que entra o sale de un proceso. Un almacén de datos representa alguna colección organizada de datos. Pero ¿cuál es el contenido de los datos implicados en las flechas o en el almacén? Si la flecha (o el almacén) representan una colección de objetos, cuáles son? Para responder a estas preguntas, aplicamos otro componente de la notación básica del análisis estructurado +1 diccionario de datos-.

### 5.13 ¿Qué es una especificación de requisito software?

La Especificación de Requisitos Software (ERS) es una descripción completa del comportamiento del sistema que se va a desarrollar. Incluye un conjunto de casos de uso que describe todas las interacciones que tendrán los usuarios con el software. Los casos de uso también son conocidos como requisitos funcionales.

### 5.14 ¿Qué es un caso de uso?

Los Casos de Uso no son parte del diseño (cómo), sino parte del análisis (qué). De forma que al ser parte del análisis nos ayudan a describir qué es lo que el sistema debe hacer. Los Casos de Uso son qué hace el sistema desde el punto de vista del usuario. Es decir, describen un uso del sistema y cómo este interactúa con el usuario.

Delimitar a partir del uso:

- ❖ Por personas, si el sistema es interactivo.
- ❖ Por máquinas, si el sistema controla procesos.
- ❖ Por otros programas, si el sistema coordina y controla aplicaciones.

Elementos:

- ❖ Actores.
- ❖ Caso de uso (Especificación del caso de uso).
- ❖ Relaciones (Diagrama de casos de uso).

Un caso de uso representa una unidad funcional coherente de un sistema, subsistema o clase.

En un caso de uso uno o más actores interactúan con el sistema que realiza algunas acciones.

Descripción de varias secuencias de acciones, con variantes, que un sistema realiza para obtener un resultado observable útil para algún actor

#### 5.14.1 Actor

Representan un conjunto coherente de papeles que los usuarios de una entidad (sistema, subsistema, clase) pueden desempeñar al interactuar con la misma.

Cualquier cosa que se comunica (interacciona) con el sistema y que es externo a él.

Representan papeles (ROLES) que interpretan personas, periféricos u otros sistemas cuando el sistema está en uso.

Un actor podría desempeñar distintos papeles dependiendo del caso de uso en que participe.

No necesariamente coinciden con USUARIOS. Un usuario puede interpretar distintos roles. Cada uno de ellos será un actor.

#### 5.14.2 Tipos de actores

**Primarios:** interaccionan con el sistema para explotar su funcionalidad; trabajan directa y frecuentemente con el software.

**Secundarios:** soporte del sistema para que los primarios puedan trabajar.

**Iniciadores:** no utilizan directamente el sistema pero desencadenan el trabajo de otro actor. (No aparecen en UML pero sí los consideran otros autores)

#### 5.14.3 Diagramas de casos de uso

Los diagramas de casos de uso muestran las relaciones entre los casos de uso de un sistema y sus actores.

Los diagramas de casos de uso dan son sólo una visión general del modelo de casos de uso

Un caso de uso se representa mediante una elipse que contiene el nombre del caso

Puede representarse mediante un monigote o mediante un rectángulo en que se indique <<actor>>.

#### 5.14.4 Relaciones

Entre casos de uso pueden darse relaciones:

- ✓ extend
- ✓ includes
- ✓ de generalización

Inclusión (<<include>>): el caso de uso inicial incluye el comportamiento del caso de uso final (subcasos).

Una relación A <<include>> a B significa que una instancia de A también incorporaría el comportamiento especificado en B.

Se añadiría en el lugar indicado en A.

Extensión (<<extend>>): el caso de uso final se puede extender con el comportamiento del caso de uso inicial en un punto concreto del primero.

si A extiende B, significa que una instancia del caso de uso B podría incorporar el comportamiento especificado en A (si se cumplen las condiciones especificadas en la extensión).

### 5.15 ¿Qué es un modelo entidad relación?

Es una herramienta de modelado para bases de datos, mediante el cual se pretende 'visualizar' los objetos que pertenecen a la Base de Datos como entidades (se corresponde al concepto de clase, cada tupla representaría un objeto, de la Programación Orientada a Objetos) las cuales tienen unos atributos y se vinculan mediante *relaciones*.

### 5.16 Contabilidad.

Es la disciplina que sirve para proporcionar información útil en la toma de decisiones, el producto final de la contabilidad son todos los Estados Contables o Estados Financieros que son los que resumen la situación económica y financiera de la empresas.

### 5.17 Sistema Contable

Es una estructura organizada mediante la cual se recogen las informaciones de una empresa como resultado de sus operaciones, valiéndose de recursos como

formularios, reportes, libros etc. y que presentados a la gerencia le permitirán a la misma tomar decisiones financieras.

Los sistemas de información contable están definidos como la combinación del personal, registros y procedimientos que se usan en una organización, con el propósito de cubrir con las necesidades rutinarias de información financiera de manera que permita el cabal cumplimiento de los objetivos de dicha organización.

### 5.18 Activos

Activos es el conjunto de bienes tangibles o intangibles que posee una empresa. Se considera activo a aquellos bienes que tienen una alta probabilidad de generar un beneficio económico a futuro y se pueda gozar de los beneficios económicos que el bien otorga. Eso no significa que sea necesaria la propiedad ni la tenencia. Los activos son un recurso o bien económico propiedad de un negocio, con el cuál se obtienen beneficios. Los activos de un negocio varían de acuerdo con la naturaleza de la empresa.

Una empresa pequeña puede tener un solo vehículo o una modesta oficina. Mientras que una gran tienda, un departamento o una fábrica, puede tener edificios, maquinarias y equipos sofisticados, terrenos, mobiliarios, cuentas por cobrar, etc.

Un ente tiene un activo cuando debido a un hecho ya ocurrido, controla los beneficios económicos que produce un bien (material o inmaterial, con valor de uso o de cambio para el ente).

Un bien tiene valor de cambio cuando existe la posibilidad de:

- ✓ Canjearlo por efectivo o por otro activo.
- ✓ Utilizarlo para cancelar una obligación.
- ✓ Distribuirlo entre los propietarios del ente.

Un bien tiene valor de uso cuando el ente puede emplearlo en alguna actividad productora de ingresos.

### 5.18.1 Clasificación de los activos

Los activos se clasifican en el balance general atendiendo a su liquidez, es decir, a la facilidad con la que puedan convertirse en efectivo, quedando de esta manera agrupados en dos categorías: activos *circulantes* y *activos no circulantes*.

### 5.18.2 Activos circulantes

Son aquellos activos de los cuales se espera que se conviertan en efectivo, se vendan o se consuman en el transcurso del ciclo normal de operaciones en la empresa. El ciclo normal de operaciones es el lapso dentro del cual la compañía utiliza efectivo para adquirir bienes, estos se venden a los clientes y el negocio recupera el efectivo; para la mayoría de las empresas, este ciclo es de unos cuantos meses, pero para otras puede abarcar más de un año.

Dentro de este grupo de activos se incluyen:

- ❖ **Efectivo:** Está constituido por las monedas en curso o sus equivalentes que están disponibles para la operación, tales como: billetes y monedas en caja, depósitos bancarios en cuentas de cheques, giros bancarios, telegráficos o postales a favor de la empresa, monedas extranjeras y metales preciosos amonedados.
- ❖ **Inversiones temporales:** las empresas deben mantener en la cuenta de cheques un mínimo de efectivo suficiente para cubrir los gastos normales de operación y cualquier imprevisto que se presente y el resto, llamado *excedente*, se invierte en instrumentos que le generen mayores rendimientos que lo que ofrece la cuenta de cheques, que es casi nulo. Si se quiere ser eficiente en la administración de los recursos no se debe tener el dinero ocioso. Por lo tanto, las inversiones temporales representan lo que la empresa tiene invertido en algún instrumento que le permite obtener un mayor rendimiento en el corto plazo, como pueden ser las inversiones en Cetes, acciones o bonos.
- ❖ **Cuentas por cobrar:** representan un derecho a favor de la empresa que se originó por la venta de productos o la prestación de servicios a crédito, el otorgamiento de préstamos o cualquier otro concepto análogo. El rubro de cuentas por cobrar incluye cuentas o documentos por cobrar a favor de la

empresa, los cuáles dependiendo de su origen, se pueden clasificar en dos grupos:

**A cargo de clientes:** documentos o cuentas por cobrar que se originaron por la prestación de un servicio o la venta de un producto a los clientes, lo cual representa la actividad normal de la empresa.

**A cargo de otros deudores:** documentos o cuentas por cobrar originados por actividades distintas de aquéllas para las cuales fue constituida la empresa, como por ejemplo préstamos a empleados o accionistas y ventas de activos fijos, impuestos por cobrar, entre otros.

Las empresas deben ser cautelosas a la hora de otorgar crédito a sus clientes; si bien es cierto que para poder incrementar el volumen de ventas es necesario ofrecer a sus clientes plazos para el pago, si no se estudia bien el historial crediticio de los mismos, puede traer como consecuencia un alto número de cuentas incobrables, lo que se reflejaría en problemas de liquidez.

Dentro de las cuentas por cobrar a cargo de otros deudores están los *impuestos por cobrar*, que representan los impuestos a favor de la empresa, como pueden ser saldos a favor de IVA (impuesto al valor agregado), de ISR (impuesto sobre la renta) o cualquier otro impuesto que se tenga a favor. En caso de que una empresa tenga al mismo tiempo impuestos por cobrar y por pagar, se presenta en el balance el neto de los dos, ya sea como activo o pasivo según sea el caso.

❖ **Inventarios:** Todos los bienes de la empresa que están destinados a la venta o a la producción para su posterior venta, la decisión del nivel de inventarios que se desea mantener en existencia repercute directamente en la liquidez de la empresa. No es una decisión sencilla de tomar; por una parte, debe mantener un nivel de inventarios suficiente para satisfacer la demanda de sus clientes, pero por otra, tener niveles excesivos de inventarios no habla de una buena administración de los recursos de la empresa, pues ese dinero podría estar generando rendimientos en una inversión temporal. El inventario ocupa espacio en las bodegas y puede volverse obsoleto, por lo que las empresas deben programar su producción o su compra de tal manera que mantengan los niveles de inventario en el mínimo posible.

- ❖ **Pagos anticipados:** Representan una erogación efectuada por servicios que se van a recibir en el futuro o por bienes que se van a consumir en el negocio y cuyo propósito no es venderlos ni utilizarlos en el proceso productivo. Algunos ejemplos de estos activos son:
- ✓ Rentas pagadas por anticipado.
  - ✓ Mantenimiento pagado por anticipado.
  - ✓ Primas de seguros y fianzas liquidadas por períodos que aún no se cumplen.
  - ✓ Materiales de oficina.

En muchas ocasiones, por la naturaleza del servicio que se espera recibir -por ejemplo un servicio de consultoría- se debe dar un anticipo o pagar el valor total del mismo antes de empezar a recibir el servicio; en otros casos, pagar por anticipado un servicio genera un ahorro para la empresa. Tomemos como ejemplo una póliza de seguro; es más barata si se cubre en un solo pago, en lugar de estar pagándola mensualmente, además de que cualquier demora en el pago de la póliza ocasionaría que ésta no fuera de utilidad en caso de un siniestro. En el caso de los materiales de oficina, existe un ahorro al comprarlos por mayoreo en lugar de estar adquiriendo lo que se necesita cada mes, siempre y cuando no se llegue a niveles excesivos de materiales, porque entonces habría un costo de oportunidad ya que ese dinero podría estar generando intereses en otro tipo de inversión.

Los pagos anticipados son considerados por las compañías como una inversión (activo), debido a que le generan beneficios por más de un periodo, y se va considerando una parte proporcional de su costo como un gasto en el estado de resultados, de lo contrario, no se lograría un enfrentamiento adecuado de ingresos y gastos.

### 5.18.3 Activos no circulantes

Son aquellos activos que no se convertirán a efectivo en el ciclo normal de operaciones. Este tipo de activos, se subdivide a su vez en tres grupos:

- ❖ **Otros activos no circulantes:** es una categoría utilizada por las empresas para agrupar aquellos activos que no empatan con ninguna otra categoría, por ejemplo los documentos o cuentas por cobrar a largo plazo y las inversiones en activos fijos que no se están utilizando actualmente, como un terreno para futuras expansiones, o las inversiones en acciones de una subsidiaria.

- ❖ **Inmuebles, maquinaria y equipo:** a estos activos también se les conoce comúnmente con el nombre de *activo fijo*, debido a que es una inversión permanente, es decir que permanece constante a lo largo del tiempo, por lo que ambos términos son utilizados indistintamente.

Los activos fijos son bienes tangibles que tienen por objeto:

1. El uso o usufructo de los mismos para beneficio de la entidad.
2. La producción de artículos para su venta o para el uso de la propia entidad.
3. La prestación de servicios a la entidad, a los clientes o al público en general.

Las características básicas que deben reunir los activos para considerarse activos fijos son:

- ❖ Que tengan naturaleza corpórea, es decir que sean bienes tangibles.
- ❖ Que sean utilizados dentro de la operación de la empresa, ya sea en la producción o suministro de bienes o servicios, para rentar a otros o para propósitos administrativos.
- ❖ Que no se adquieran con el propósito de venderlos.
- ❖ Que sean empleados sobre una base continua.

Como ejemplos de este tipo de activos tenemos:

- ✓ Terrenos
- ✓ Edificios
- ✓ Equipo de oficina
- ✓ Equipo de cómputo
- ✓ Equipo de transporte
- ✓ Maquinaria y equipo de fábrica
- ✓ Herramientas

La decisión de las empresas de cuánto invertir en activos fijos es una de las más importantes que se toman en los negocios; antes de invertir en maquinaria y equipo deben estudiar bien sus pronósticos de ventas en el largo plazo, porque difícilmente pueden dar marcha atrás a esas decisiones sin comprometer una gran cantidad de recursos.

Por ejemplo, si se invierte en un hotel de cinco pisos, y sólo se estuvieran rentando dos, estarían tres pisos desperdiciados, lo que

representaría recursos que no estarían generando ningún beneficio para la empresa, además de ocasionar gastos por mantenimiento y limpieza, entre otros; una vez hecha la inversión, la empresa no podría dar marcha atrás ni podría mantener la operación del hotel con esos niveles de ocupación por mucho tiempo, lo que ocasionaría que tarde o temprano tuviera que cerrar.

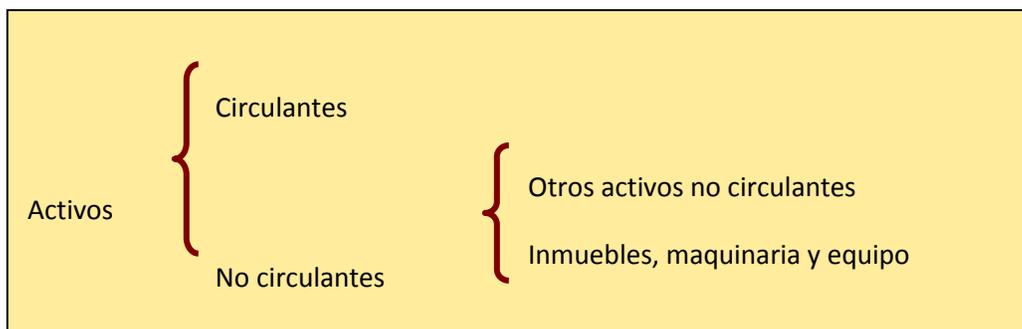
Asimismo en caso de requerir financiamiento para la compra de un activo fijo, las compañías deben recurrir a financiamientos a largo plazo, debido a que el activo no proporciona beneficios de inmediato, sino en el mediano y largo plazo.

- ❖ **Activos intangibles:** Son aquellos activos no circulantes que no poseen una sustancia física, es decir, que no se pueden tocar; sin embargo son utilizados para la producción o abastecimiento de bienes, prestación de servicios o para propósitos administrativos, y generarán beneficios económicos futuros que serán controlados por la empresa.

Algunos ejemplos de este tipo de activos son:

- ❖ **Patentes:** representan derechos exclusivos para la producción y venta de bienes que poseen una o más características singulares. Es el caso de los inventos que se patentan para que nadie los copie, por ejemplo la fórmula secreta del Coronel Sanders de KFC.
- ❖ **Marcas registradas:** es un nombre, palabra, frase, término o símbolo utilizado para identificar una empresa y sus productos o servicios y diferenciarlos de cualquier otro que intente apropiársela o imitarla. Por ejemplo, la frase de Sabritas "A que no puedes comer sólo una". Una marca registrada de servicios es lo mismo que una marca registrada de productos, con la excepción de que identifica y distingue la fuente de un servicio en vez de un producto. Normalmente, una marca de un producto aparece en el mismo o en su envoltura con el símbolo ®, mientras que una marca registrada de un servicio aparece en sus promociones.

- ❖ **Licencias y permisos:** son derechos y privilegios que otorgan el gobierno o los particulares para realizar la venta de algún producto o la prestación de un servicio, por ejemplo, el permiso de explotación de una mina.
- ❖ **Franquicias:** es un contrato que hace una empresa con otra, para utilizar su marca comercial y su *saber hacer* por un tiempo y en un territorio determinado, mediante el pago de una remuneración económica.
- ❖ **Derechos de autor:** las leyes de los derechos de autor garantizan que su dueño tiene el derecho exclusivo para reproducir una obra (libro, película, canción, etc.) así protegida, así como para preparar obras derivadas de la misma, distribuir copias, exhibirla públicamente o autorizar a otros para llevar a cabo cualquiera de estas actividades.
- ❖ **Crédito mercantil:** cuando una empresa es adquirida por otra se le asignan valores de mercado a los activos de la empresa que está siendo vendida, a ese importe se le restan los valores de mercado de sus pasivos y cualquier cantidad pagada de más se registra por la empresa compradora como crédito mercantil.



## 5.19 Pasivo

Está formado por el conjunto de recursos financieros obtenidos por la empresa para el desarrollo de sus funciones y por las estimaciones de gastos futuros.

### 5.19.1 Clasificación del pasivo

Los pasivos se clasifican de acuerdo con su fecha de vencimiento o liquidación en dos grupos: **pasivo a corto plazo** y **pasivo a largo plazo**.

### 5.19.2 Pasivo a corto plazo

Es aquel cuya liquidación se producirá antes de un año, o dentro del ciclo de operaciones a corto plazo como pueden ser:

- ✓ Préstamos bancarios a corto plazo
- ✓ Deudas con proveedores
- ✓ Impuestos por pagar
- ✓ Dividendos por pagar
- ✓ Anticipos de clientes
- ✓ Deudas con acreedores diversos y gastos acumulados no pagados
- ✓ Provisiones para contingencias
- ✓ Porción a corto plazo de los pasivos a largo plazo, es decir, la parte de los pasivos a largo plazo que tendrá que pagarse en el ciclo normal de operaciones.

### 5.19.3 Pasivo a largo plazo

El pasivo a largo plazo está integrado por todas aquellas deudas y obligaciones que la empresa tendrá que cubrir en un plazo mayor a un año o del ciclo de operaciones a corto plazo tales como:

- ✓ Préstamos bancarios a largo plazo
- ✓ Hipotecas por pagar
- ✓ Obligaciones por pagar
- ✓ Otros pasivos a largo plazo

## 5.20 Balance General

Es un resumen de todo lo que tiene la empresa, de lo que debe, de lo que le deben y de lo que realmente le pertenece a su propietario, a una fecha determinada.

Al elaborar el balance general el empresario obtiene la información valiosa sobre su negocio, como el estado de sus deudas, lo que debe cobrar o la disponibilidad de dinero en el momento o en un futuro próximo.

El Balance General está formado por tres cuentas:

❖ **La cuenta de los activos**

Es el conjunto de Bienes, derechos y valores que la empresa tiene en su poder en el día de su balance.

❖ **La cuenta de los pasivos**

Es el conjunto de deudas y obligaciones que la empresa tiene en la fecha del balance.

❖ **La cuenta del patrimonio**

Es todo lo que pertenece al dueño de la empresa en el momento del balance; es decir, lo que le quedaría si le pagaran y si pagara todas las deudas ese día.

### 5.20 ¿Qué es la valorización?

La valorización es la diferencia entre el valor de un bien depreciado y su valor comercial.

La depreciación acumulada no es una cuenta del activo ni del pasivo, es una disminución de los activos fijos.

### 5.21 ¿Qué es el patrimonio?

Es el valor de los bienes que realmente pertenecen a la empresa. El patrimonio es lo que le quedaría a la empresa si cumpliera todas las obligaciones en la fecha del balance.

El patrimonio tiene cuatro cuentas:

### 5.22 Capital

Es la suma del dinero y del valor de los bienes con los cuales se fundo la empresa; o sea la parte del patrimonio que apporto el empresario.

### 5.23 Utilidades acumuladas

Son las utilidades obtenidas durante el tiempo que la empresa lleva funcionando y que han sido reinvertidas por ella.

### 5.24 ¿Qué es el estado de ganancia o pérdida?

- ❖ Es una operación que me informa los resultados de la actividad de la empresa en un periodo determinado.
  
- ❖ El estado de ganancias o pérdidas esta constituido por las cuentas de resultados:
  - ✓ Ventas.
  - ✓ Compras.
  - ✓ Costos y Gastos.
  - ✓ Depreciación.

El estado de ganancias o pérdidas consiste en averiguar:

- ❖ Cuanto vendió la empresa en el período.
- ❖ Cuanto costaron los productos que vendió.
- ❖ Cuanto costo la administración del negocio.
- ❖ Cuanto le quedo de ganancia.

La utilidad bruta es lo que deja el negocio para pagar la administración y para generar la ganancia.

La utilidad liquida es el dinero que queda libre después de pagar todos los gastos.

Cuando la utilidad bruta es mayor que los gastos de administración, la empresa ha dado ganancias.

En cambio, Cuando los gastos de administración suman más que la utilidad bruta, la empresa ha dado pérdida.

### 5.25 ¿Cómo se hace el estado de ganancia o pérdida?

#### 1. **Determino el periodo**

Quiere decir que lo enmarco entre dos fechas precisas.

#### 2. **Reúno la información**

- ✓ ¿Cuánto vendió la empresa en el periodo?
- ✓ ¿Cuánto costaron los productos vendidos en el periodo?
- ✓ ¿Cuánto sumaron los gastos de administración en el periodo?

En las ventas de periodo sumo las ventas a crédito y las ventas de contado.

#### 3. **Realizo la operación y lo registro el formato**

### 5.26 Que es el flujo de caja o de presupuesto de efectivo.

- ✓ Es una Herramienta Fundamental para Administrar la Liquidez
- ✓ Permite la Planificación Financiera
- ✓ Facilita el Control y Coordinación del Efectivo
- ✓ Permite establecer la Capacidad de Pago
- ✓ Es un mecanismo para disminuir el riesgo
- ✓ Es un pronóstico de los probables ingresos y egresos en efectivo en un período.

### 5.27 Propósito del flujo de caja

- ✓ Presentar la información correspondiente a las entradas y salidas de efectivo de una empresa, durante un período determinado de tiempo.
- ✓ Evaluar la habilidad de la empresa para generar futuros flujos de efectivo.
- ✓ Prever las necesidades de efectivo y la manera de cubrirlas adecuada y oportunamente.
- ✓ Permitir la planeación de lo que la empresa puede hacer con los excesos temporales de efectivo.

### 5.28 Flujo de caja elementos básicos

- ✓ El Saldo Inicial de Caja
- ✓ Las entradas de efectivo
- ✓ Las salidas de efectivo
- ✓ El Flujo Neto del Período
- ✓ El Saldo Final de Caja

### 5.29 Como se prepara el flujo de caja

- ✓ Definición de los factores determinantes de ingresos.
- ✓ Definición de los factores determinantes de egreso
- ✓ Establecer sus parámetros de comportamiento
- ✓ Cálculo de los ingresos y egresos Operacionales y financieros

### 5.30 Balanza comparativa

Para hacer una estimación del aumento o de la disminución del capital contable, así como de las modificaciones que han sufrido el activo por las operaciones practicadas, es necesario establecer una comparación entre el balance existente al principio del ejercicio y el actual. El balance existente al principio del ejercicio se conoce con el nombre de balance inicial y el actual como balance final. El documento en el cual se comparan el balance inicial y el balance final se llama balance comparativo.

### 5.31 Balance comparativo

Es un documento contable que muestra los aumentos o disminuciones que han tenido el Activo, Pasivo y el Capital contable por las operaciones efectuadas durante el ejercicio. Los aumentos y disminuciones del Activo y del Pasivo ya fueron tratados ampliamente; por tanto, únicamente se van a estudiar los aumentos y disminuciones del capital contable.

### 5.32 Aumentos del Capital contable

El Capital contable aumenta cuando la empresa percibe ingresos o cuando recibe aportaciones adicionales de capital.

### 5.33 Ingreso

Son las utilidades que generan los bienes o servicios prestados por la empresa.

### **5.34 Aportaciones adicionales de capital**

Son las entregas en efectivo o en especie que hace el inversionista para incrementar su capital propio, o las que hacen los socios para aumentar la inversión original de capital social.

Los aumentos del Capital contable que producen tanto ingresos como las aportaciones adicionales de capital aumentan el Activo o disminuyen el Pasivo.

### **5.35 Disminuciones del Capital contable**

El capital contable disminuye cuando la empresa causa egresos o cuando acepta reducciones parciales de capital.

### **5.36 Egresos**

Son los gastos que originan los bienes o servicios consumidos por la empresa.

### **5.37 Reducciones parciales de capital**

Son los retiros en efectivo o en especie que hace el inversionista para reducir su capital propio, o los que hacen los socios para disminuir la inversión original del capital social.

Las disminuciones del capital contable que originan tanto los egresos como las reducciones parciales del capital disminuyen el Activo o aumentan el Pasivo.

### **5.38 Determinación del Capital contable final**

Para determinar el Capital contable final se suman al Capital contable inicial de los ingresos y las aportaciones adicionales de capital, y se restan de el los egresos y las reducciones parciales de capital.

Precisamente, el balance comparativo es el documento contable que muestra los aumentos o disminuciones que han tenido el Activo, Pasivo y Capital contable por las operaciones efectuadas durante el ejercicio.

### **5.39 Modificaciones del Activo y del Pasivo que aumentan el Capital contable**

El capital contable aumenta en los siguientes casos:

- a) Cuando aumenta el activo
- b) Cuando disminuye el pasivo

### **6 Modificaciones del Activo y el Pasivo que disminuye el Capital contable**

El capital contable disminuye en los siguientes casos:

- a) Cuando disminuye el activo
- b) Cuando aumenta el pasivo

#### **5.40 Casos en que aumenta el capital contable**

- a) Cuando aumenta el activo solamente
- b) Cuando disminuye pasivo solamente
- c) Cuando aumenta el activo y disminuye el pasivo
- d) Cuando aumenta el activo en mayor proporción que el pasivo
- e) Cuando disminuye el pasivo en mayor proporción que el activo

#### **5.41 Casos en que disminuye el capital contable**

- a) Cuando disminuye el activo solamente
- b) Cuando aumenta el pasivo solamente
- c) Cuando disminuye el activo aumenta el pasivo
- d) Cuando disminuye el activo en mayor proporción que el pasivo
- e) Cuando aumenta el pasivo en mayor proporción que el activo

Para que el balance comparativo se pueda interpretar con más facilidad es necesario que tanto el activo como el pasivo aparezcan debidamente clasificados.

- ❖ **Encabezado** -- Nombre del negocio, Indicación de que se trata de un balance comparativo, Fecha de los ejercicios que se compara.
- ❖ **Cuerpo** -- Valor detallado de cada uno de los bienes que formen el activo tanto del balance inicial como del final, Valor detallado de cada una de las obligaciones que formen el pasivo tanto del balance inicial como del final, Importe del capital inicial así como del final.
- ❖ **Firmas** -- Del contador que lo hizo y autorizo, Del propietario.

**Al igual que el balance general, el balance comparativo también se puede presentar de dos formas:**

- ✓ Con forma de reporte
- ✓ Con forma de cuenta

#### 5.41.1 Balance comparativo con forma de reporte:

Esta forma consiste en anotar clasificadamente el activo y el pasivo de los balances que se comparan en una sola página de tal manera que a la suma del activo se le pueda restar verticalmente la suma del pasivo para determinar el capital contable de ambos ejercicios.

#### 5.41.2 Empleo de columnas:

Se utilizan 4 columnas

En la primera: se anota la cantidad de cada una de las cuentas que integran el balance final.

En la segunda: se anota la cantidad de cada una de las cuentas que integran el balance inicial

En la tercera: se anota la diferencia de cada cuenta

En la última: se anotan los signos de las cuentas si hubo un aumento o disminución (+, -).

### 5.41.3 Comprobación del balance:

El balance se dice que es correcto cuando la suma de las primeras dos columnas es igual; y cuando la suma de las últimas columnas también es igual.

### 5.42 Balance comparativo en forma de cuenta

Se emplean dos páginas; en la izquierda se anota clasificadamente el activo de los balances que se comparan, y en la derecha, el pasivo y el capital contable de ambos ejercicios, también debidamente clasificados y con sus respectivos signos al igual que el balance en forma de reporte.

### 5.43 Catálogo de Cuentas

Contiene todas las cuentas que se estima serán necesarias al momento de instalar un sistema de contabilidad. Debe contener la suficiente flexibilidad para ir incorporando las cuentas que en el futuro deberán agregarse al sistema.

Como su nombre lo indica, el catálogo es una lista de todas las cuentas que pueden necesitarse en vista de las actividades a que la empresa se dedique. Por supuesto, habrá cuentas que se usen en todas las empresas y otras que sean características de cierto tipo de negocios.

Es el índice o instructivo, donde, ordenada y sistemáticamente, se detallarán todas las cuentas aplicables en la contabilidad de una negociación o empresa, proporcionando los nombres y, en su caso los números de las cuentas”.

Objetivos.

- ❖ Permitir que distintos empleados puedan mantener registros coherentes con la implementación de un catálogo de cuenta similar.
- ❖ Facilitar el trabajo contable sobre todo cuando se trata de consolidar cifras financieras.
- ❖ Facilitar y satisfacer la necesidad de registro diario de las operaciones de una empresa o entidad.

#### 5.44 Formularios

Son todas las formas impresas con la finalidad de recaudar información en las diferentes áreas de una empresa. Constituyen un elemento que siempre y cuando esté autorizado (firmado) podrá servir como comprobante para garantizar una operación.

#### 5.45 Estados financieros

Los estados financieros son los documentos que debe preparar la empresa al terminar el ejercicio contable, con el fin de conocer la situación financiera y los resultados económicos obtenidos en las actividades de su empresa a lo largo de un período

#### 5.46 La utilidad de la información contable

La información presentada en los estados financieros interesa a:  
La administración, para la toma de decisiones, después de conocer el rendimiento, crecimiento y desarrollo de la empresa durante un periodo determinado.

Los propietarios para conocer el progreso financiero del negocio y la rentabilidad de sus aportes.

Los acreedores, para conocer la liquidez de la empresa y la garantía de cumplimiento de sus obligaciones.

El estado, para determinar si el pago de los impuestos y contribuciones está correctamente liquidado.

#### ❖ Clasificación

Los estados financieros básicos son:

- ✓ El balance general
- ✓ El estado de resultados
- ✓ El estado de cambios en el patrimonio
- ✓ El estado de cambios en la situación financiera

- ✓ El estado de flujos de efectivo
- ❖ Responsabilidad

Los estados financieros deben llevar las firmas de los responsables de su elaboración, revisión y elaboración.

#### 5.47 Libros de contabilidad

Son aquellos en los que se anotan de manera definitiva las informaciones que aportan los formularios. Los libros donde se asientan las operaciones de la empresa con el fin de cumplir las obligaciones que impone la ley a este respecto y lograr la información o los datos necesarios para conocer su situación y resultados mediante balances y estados demostrativos de ganancias y pérdidas.

#### 5.48 Bases de Datos

Es una colección de datos clasificados y estructurados con un asunto o finalidad. Los datos en una base de datos están clasificados en campos u registros.

Una **base de datos** o **banco de datos** (en ocasiones abreviada BB.DD.) es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En este sentido, una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta. En la actualidad, y debido al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos están en formato digital (electrónico), que ofrece un amplio rango de soluciones al problema de almacenar datos.

Existen programas denominados sistemas gestores de bases de datos, abreviado SGBD, que permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada. Las propiedades de estos SGBD, así como su utilización y administración, se estudian dentro del ámbito de la informática.

Las aplicaciones más usuales son para la gestión de empresas e instituciones públicas. También son ampliamente utilizadas en entornos científicos con el objeto de almacenar la información experimental.

#### 5.48.1 Bases de datos relacionales

Éste es el modelo utilizado en la actualidad para modelar problemas reales y administrar datos dinámicamente. Tras ser postulados sus fundamentos en 1970 por Edgar Frank Codd, de los laboratorios IBM en San José (California), no tardó en consolidarse como un nuevo paradigma en los modelos de base de datos. Su idea fundamental es el uso de "relaciones". Estas relaciones podrían considerarse en forma lógica como conjuntos de datos llamados "tuplas". Pese a que ésta es la teoría de las bases de datos relacionales creadas por Codd, la mayoría de las veces se conceptualiza de una manera más fácil de imaginar. Esto es pensando en cada relación como si fuese una tabla que está compuesta por *registros* (las filas de una tabla), que representarían las tuplas, y *campos* (las columnas de una tabla).

En este modelo, el lugar y la forma en que se almacenen los datos no tienen relevancia (a diferencia de otros modelos como el jerárquico y el de red). Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar para un usuario esporádico de la base de datos. La información puede ser recuperada o almacenada mediante "consultas" que ofrecen una amplia flexibilidad y poder para administrar la información.

El lenguaje más habitual para construir las consultas a bases de datos relacionales es SQL, *Structured Query Language* o *Lenguaje Estructurado de Consultas*, un estándar implementado por los principales motores o sistemas de gestión de bases de datos relacionales.

#### 5.49 SQL Server 2008

Es un sistema de gestión de bases de datos relacionales (SGBD) basado en el lenguaje Transact-SQL, y específicamente en Sybase IQ, capaz de poner a disposición de muchos usuarios grandes cantidades de datos de manera simultánea.

### **5.50 Expression Blend:**

Es la herramienta de diseño profesional para crear experiencias de usuario atractivas y conectadas a la Web para Windows, abriendo la puerta a la creación de diseños de interfaces de usuario mediante una amplia gama de tipos de medios, trabajando en un entorno de diseño en tiempo real.

### **5.51 Project 2007**

Es un programa de la suite Microsoft Office usado para la gestión de proyectos.

Microsoft Project (o MSP) es un software de administración de proyectos diseñado, desarrollado y comercializado por Microsoft para asistir a administradores de proyectos en el desarrollo de planes, asignación de recursos a tareas, dar seguimiento al progreso, administrar presupuesto y analizar cargas de trabajo.

### **5.52 SQL Server Management Studio**

SQL Server Management Studio es un entorno integrado para obtener acceso a todos los componentes de SQL Server, configurarlos, administrarlos y desarrollarlos. SQL Server Management Studio combina un amplio grupo de herramientas gráficas con una serie de editores de script para ofrecer acceso a SQL Server a programadores y administradores de todos los niveles de especialización.

SQL Server Management Studio combina las características del Administrador corporativo, el Analizador de consultas y Analysis Manager, herramientas incluidas en versiones anteriores de SQL Server, en un único entorno. Además, SQL Server Management Studio funciona con todos los componentes de SQL Server, como Reporting Services, Integration Services y SQL Server Compact 3.5 SP1. Los programadores obtienen una experiencia familiar y los administradores de bases de datos una única herramienta completa que combina herramientas gráficas fáciles de usar con funciones de script enriquecidos.

### 5.53 Visual Studio 2008

- ✓ Visual Studio 2008 nos provee una serie de herramientas para desarrollo, así como características de debugging, funcionalidad en base de datos y características innovadoras para la creación de aplicaciones en una variedad de plataformas.
- ✓ Visual Studio 2008 incluye realces como un diseñador visual para desarrollo rápido con el .NET Framework 3.5, esto nos ayuda mucho a los que desarrollan en web por que se incluyen las características de Microsoft Expression Web, que la verdad en mi punto de vista esta excelente, Visual Studio 2008 provee a desarrolladores con todas las herramientas y el framework el poder crear aplicaciones web con el soporte de AJAX.
- ✓ Visual Studio 2008 nos provee un nuevo lenguaje de consultas integrado para el manejo de la información, el cual se llama Microsoft Language Integrated Query (LINQ), que es lo nos hará la vida más fácil para programadores individuales para poder construir soluciones que analicen y actúen sobre la información.
- ✓ Visual Studio 2008 también nos provee a desarrolladores la habilidad de poder escoger entre múltiples versiones del Framework con el mismo entorno de desarrollo, así nosotros podemos desarrollar en la versión que queramos ya sea en .NET Framework 2.0, 3.0 o 3.5, entiendo así que soporta un gran variedad de proyectos en la versión X en el mismo entorno de desarrollo.
- ✓ Visual Studio 2008 ofrece a desarrolladores nuevas herramientas para la fácil creación de aplicaciones conectadas en las ultimas plataformas incluyendo web, Windows Vista, Office 2007, SQL Server 2008 y Windows Server 2008. Para la web, tenemos ASP.NET, AJAX y otras tecnologías como Silverlight, WPF, etc. que nos dará la posibilidad de crear aplicaciones con rica interfaz de usuario, para poder dar una experiencia de usuario sin precedentes.

Visual Studio 2008 nos entrega las ofertas para poder tener la colaboración en equipos de desarrollo mediante la nueva visión que tiene Microsoft, incluyendo herramientas que nos ayudan a integrar manejadores de base de datos (DBA's) y diseñadores gráficos en este proceso del desarrollo de software.

### **5.53.1 Usando la versión de Microsoft .NET Framework 3.5**

Se han incluido librerías con este nuevo Framework de Trabajo, recordemos que en la versión 3.0, teníamos 4 tecnologías clave, las de Windows Presentation Foundation, Windows Workflow, Windows Cardspace, Windows Communication Foundation, ahora tenemos las mismas tecnologías solo que se añaden más como LINQ, ASP.NET 3.5, Librerías y funcionamientos para el CRL, y más.

### **5.54 Report Wizard**

El Report Wizard es un componente multinivel para generar informes profesionales en profundidad con un sólo clic desde virtualmente cualquier fuente de datos. Esta versión soporta muchas funcionalidades incluyendo informes dinámicos rápidos, gráficos y diagramas sencillos, informes de datos multinivel, ordenación de múltiples columnas dinámicas, filtrado de datos, generación de informes de múltiples tablas, sumarios de múltiples campos, formateado de datos de los campos, creación de alias para los nombres de campos y mucho más...

El Report Wizard es la solución más productiva para crear visualmente aplicaciones para informes basados en bases de datos y sin necesidad de programar. El soporte para virtualmente todas las bases de datos relacionales y otras bases de datos más antiguas es lo que distingue a Report Wizard.

### **5.55 Report Viewer**

Los controles ReportViewer se utilizan para alojar los informes creados en Microsoft Visual Studio 2008. Hay dos versiones del control: el control de servidor Web ReportViewer para páginas ASP.NET y el control de Windows Forms ReportViewer que puede utilizarse en aplicaciones de Windows Forms. Visual Studio determinará qué versión debe utilizarse según el tipo de proyecto que use para incrustar el control. Si incrusta el control ReportViewer

en un proyecto de ASP.NET, por ejemplo, obtendrá el control de servidor Web automáticamente.

### 5.56 Características de ReportViewer

Las dos versiones del control ReportViewer admiten las siguientes características:

- ❖ Áreas de vista para mostrar un informe, una barra de herramientas y un mapa del documento. La barra de herramientas es configurable y proporciona características en tiempo de ejecución para admitir la exploración en un informe de varias páginas y la funcionalidad de zoom, búsqueda, impresión y exportación. Se mostrará un mapa del documento si agrega uno explícitamente al informe.
- ❖ Propiedades que le permitirán configurar el modo de procesamiento, las áreas de vista y la barra de herramientas.
- ❖ Compatibilidad con los modos de procesamiento local y remoto para controlar dónde y cómo se procesan los informes. El modo de procesamiento local recupera y combina un conjunto de datos existente en un diseño de informe y representa el informe mediante la funcionalidad de procesamiento interna al control. El procesamiento remoto representa un informe publicado en un servidor de informes de Microsoft SQL Server Reporting Services.
- ❖ Interfaces de programación que permiten personalizar, configurar e interactuar con el control mediante código, así como cambiar los orígenes de datos que utiliza ReportViewer en tiempo de ejecución.

## VI. DISEÑO METODOLÓGICO

Para el desarrollo de nuestro proyecto hemos tomado a bien, utilizar el Modelo de ciclo de vida del software o como muchos lo conocemos *Modelo de Cascada*. El mismo consiste en los siguientes pasos:

- ❖ **Análisis:** En esta etapa, lo que se pretende lograr es dar un profundo estudio a los requisitos que se deben y que se necesitan emplear en el sistema.
- ❖ **Diseño:** En esta parte tomaremos en cuenta la forma detallada del medio visual del usuario final, es decir; de qué forma estará creada la interfaz. La cual nos permitirá implementar o poner en práctica nuestra creatividad.
- ❖ **Codificación:** Es un proceso en donde se edita en un lenguaje el cual es entendible para la maquina. El mismo tendrá como objetivo la realización de interfaces útiles para el manejo del sistema.
- ❖ **Prueba:** Consiste en que los datos procesados sean los esperados. Esto significa, que en este apartado se harán simulaciones con datos verdaderos, los cuales nos permitan decidir si el sistema está correctamente.
- ❖ **Mantenimiento:** Consiste en realizar tareas con el objetivo de que nuestro sistema sea fiable, esto es; que el mismo pueda darnos la certeza y la plena convicción de que los resultados obtenidos sean lo correctos.

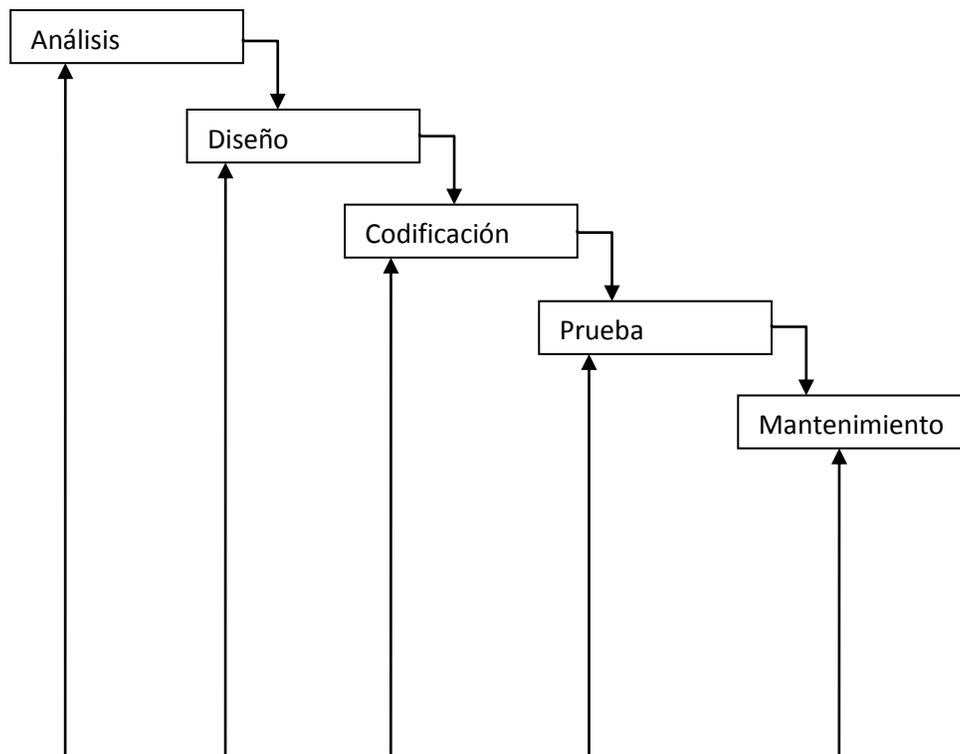
### 1. Recopilación de la información.

Para el desarrollo de este proyecto, realizaremos entrevistas, las cuales serán dirigidas al usuario final. Esto es con el objetivo de saber las debilidades que tiene el sistema con el que cuentan y así no implementar las mismas en nuestro sistema.

Realizaremos consultas a libros de Contabilidad, visitas a Sitios web, consultas a personas capacitadas en esa área y el uso de la ayuda que nos brinda cada uno de los programas que vayamos a utilizar.

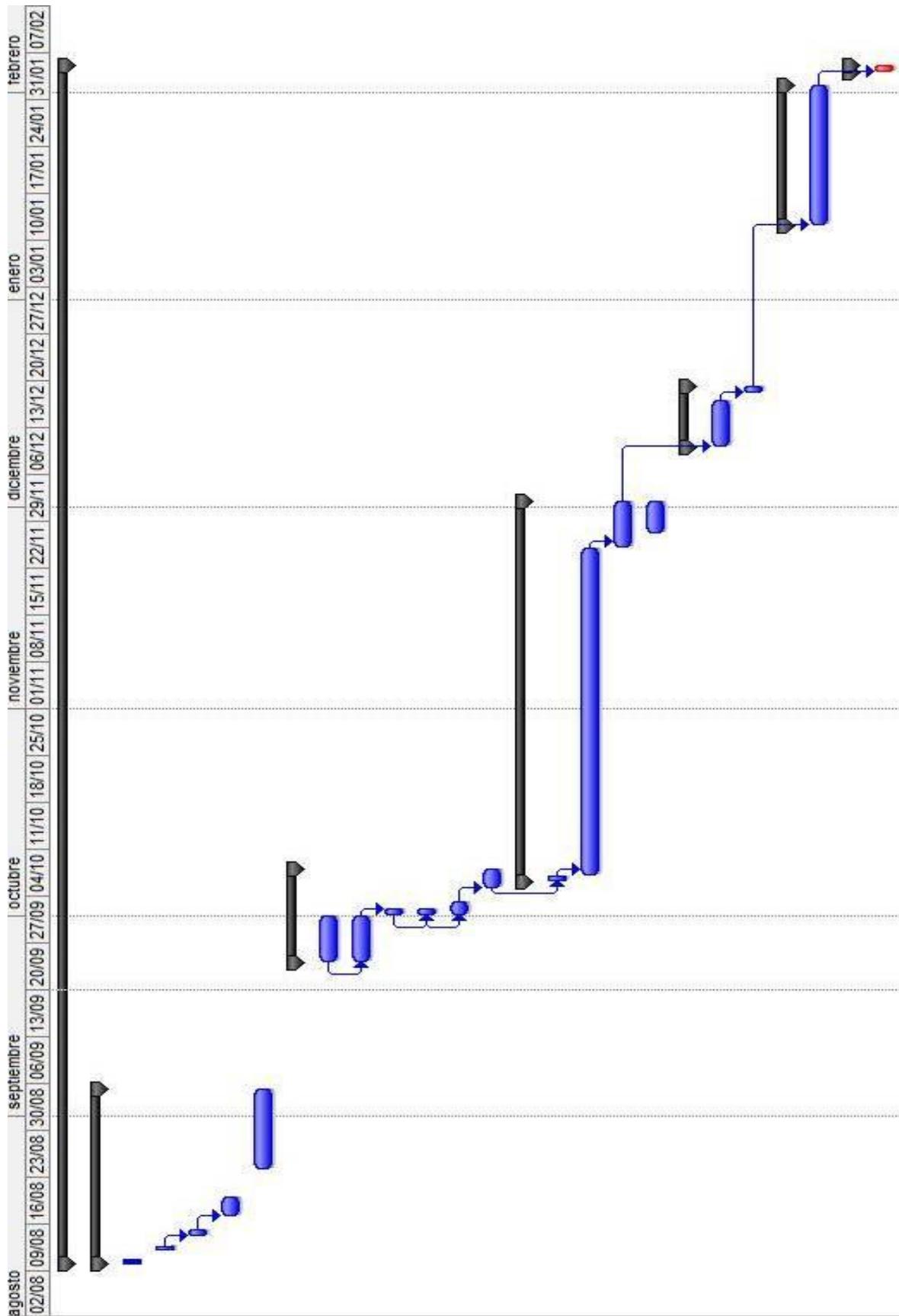
## 2. Modelo en cascada

En el siguiente gráfico, se muestra las fases o etapas en las que dividimos nuestro proyecto. El mismo, servirá para llevar un patrón a seguir. Cabe destacar, que este modelo, nos permitirá desplazarnos a etapas anteriores para realizar algunas mejoras que queramos implementar durante el desarrollo de nuestro sistema.

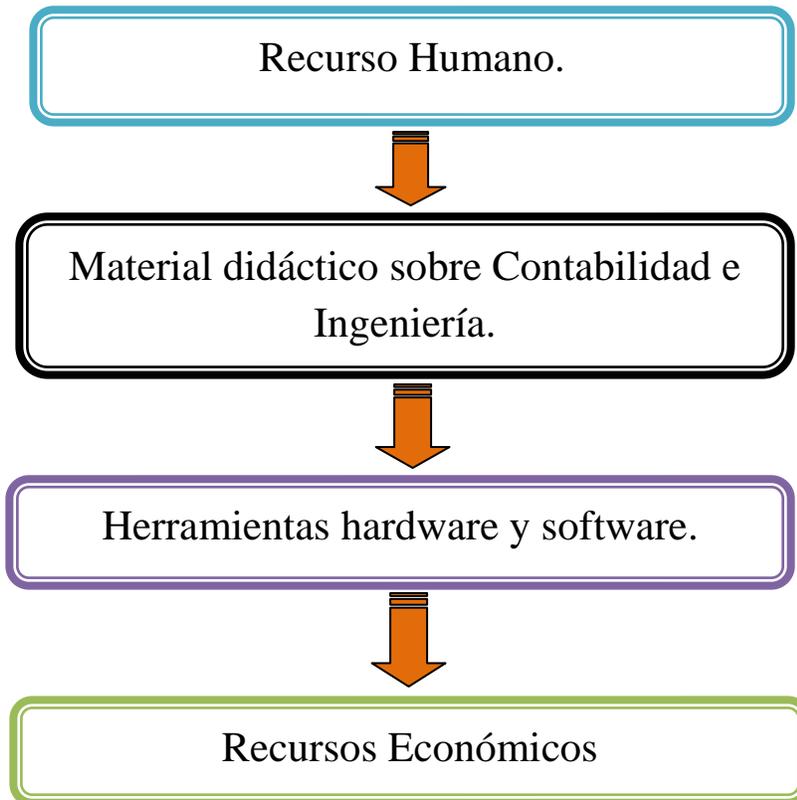


## VII. CRONOGRAMA

| Tareas para el Sistema Contable (SCCSJ)<br>Fecha inicio: 10/08/2009 Fecha de finalización:<br>04/02/2010 |  |
|--|--|
| Fase de inicio   |  |
| Reunión con el administrador de la empresa   |  |
| Establecer plan de reuniones   |  |
| Identificación del problema  |  |
| Análisis de la información   |  |
| Tratamiento de la información  |  |
| Fase de diseño   |  |
| Diseño de ERS  |  |
| Diseño de los DFD  |  |
| Diseño de la base de datos   |  |
| Diseño de interfaces   |  |
| Diseño del manual de usuario   |  |
| Documentar la fase de diseño   |  |
| Fase de desarrollo   |  |
| Creación de la base de datos   |  |
| Codificación de módulos  |  |
| Crear apariencias de interfaces  |  |
| Crear manual de usuario  |  |
| Fase de implementación   |  |
| Verificación de módulos  |  |
| Empaquetado del sistema  |  |
| Fase de prueba   |  |
| Validación del sistema   |  |
| Fase de mantenimiento  |  |
| Capacitación a usuarios finales  |  |



## VIII. RECURSOS Y NECESIDADES DEL PROYECTO



- ❖ **Recursos humanos:** Equipo de tesis encargado del desarrollo del sistema y colaboradores en la parte de contabilidad y tutorial.
- ❖ **Material didáctico:** Referente a documentos bibliográficos sobre contabilidad, metodologías de investigación, elaboración de proyectos y programación. Además, de la ayuda en línea con la codificación.
- ❖ **Herramientas hardware y software:** Equipos y sistemas para la elaboración del proyecto.
- ❖ **Recursos económicos:** Los costos del proyecto son asumidos por los integrantes.

## IX. DICCIONARIO DE DATOS

**Cuenta de usuario** = Nombre de usuario + Contraseña

**Nombre de usuario** = \* Cualquier cadena menor o igual a 15 caracteres \*

**Contraseña** = \* Cualquier cadena menor o igual a 8 caracteres \*

**Número de cuenta de catálogo** = Código de cuenta + Código de Sub-Cuenta + Código de Sub Sub-Cuenta + Código de Sub Sub Sub-Cuenta

**Código de cuenta** = \* Cualquier cadena de 3 dígitos \*

**Código de Sub-Cuenta** = \* Cualquier cadena de 2 dígitos \*

**Código de Sub Sub-Cuenta** = \* Cualquier cadena de 2 dígitos \*

**Código de Sub Sub Sub-Cuenta** = \* Cualquier cadena de 5 dígitos \*

**Fecha de periodo fiscal** = Fecha de inicio + Fecha de fin

**Fecha de cierre mensual** = Fecha de inicio + Fecha fin

**Fecha de inicio** = Fecha

**Fecha de fin** = Fecha

**Fecha** = Día + Mes + Año

**Día** = [01|02|...|31]

**Mes** = [01|02|...|12]

**Año** = \* Cadena de 4 dígitos \*

**Número de referencia** = [1|2|3|...|1000000]

**Catalogo de cuenta** = Nombre de la cuenta + Numero de cuenta de catalogo + Nivel de la cuenta

**Nivel de la cuenta** = [1|2|3|4|5]

**Nombre de la cuenta** = \* Cualquier cadena \*

**Ingreso** = Concepto + Número de comprobante + Fecha + Número de referencia + Número de cuenta de catalogo + Debe + Haber

**Egreso** = Concepto + Número de comprobante + Fecha + Número de referencia + Número de cuenta de catalogo + Debe + Haber

**Ajuste** = Concepto + Número de comprobante + Fecha + Número de referencia + Número de cuenta de catalogo + Debe + Haber

**Diario** = Concepto + Número de comprobante + Fecha + Número de referencia + Número de cuenta de catalogo + Debe + Haber

**Concepto** = \* Cualquier cadena de caracteres \*

**Número de comprobante** = \* Cualquier cadena de digito (s) \*

**Debe** = \* Cualquier cadena de dígitos \*

**Haber** = \* Cualquier cadena de dígitos \*

## X. MODELO CONCEPTUAL

**Base de Datos:** BBDD\_SCCSJ

**Tablas:**

Comprobantes

| Atributos     | Tipo de dato | Valor | Referencia  |
|---------------|--------------|-------|-------------|
| NumReferencia | Varchar      | 15    | Primary key |
| NumCta        | Varchar      | 15    | Not Null    |
| Dia           | Int          | 10    | Not Null    |
| Mes           | Int          | 10    | Not Null    |
| Anyo          | Int          | 10    | Not Null    |
| Concepto      | Varchar      | 100   | Not Null    |
| Debe          | Float        | 10    | Not Null    |
| Haber         | Float        | 10    | Not Null    |
| Tipo          | Varchar      | 15    | Not Null    |

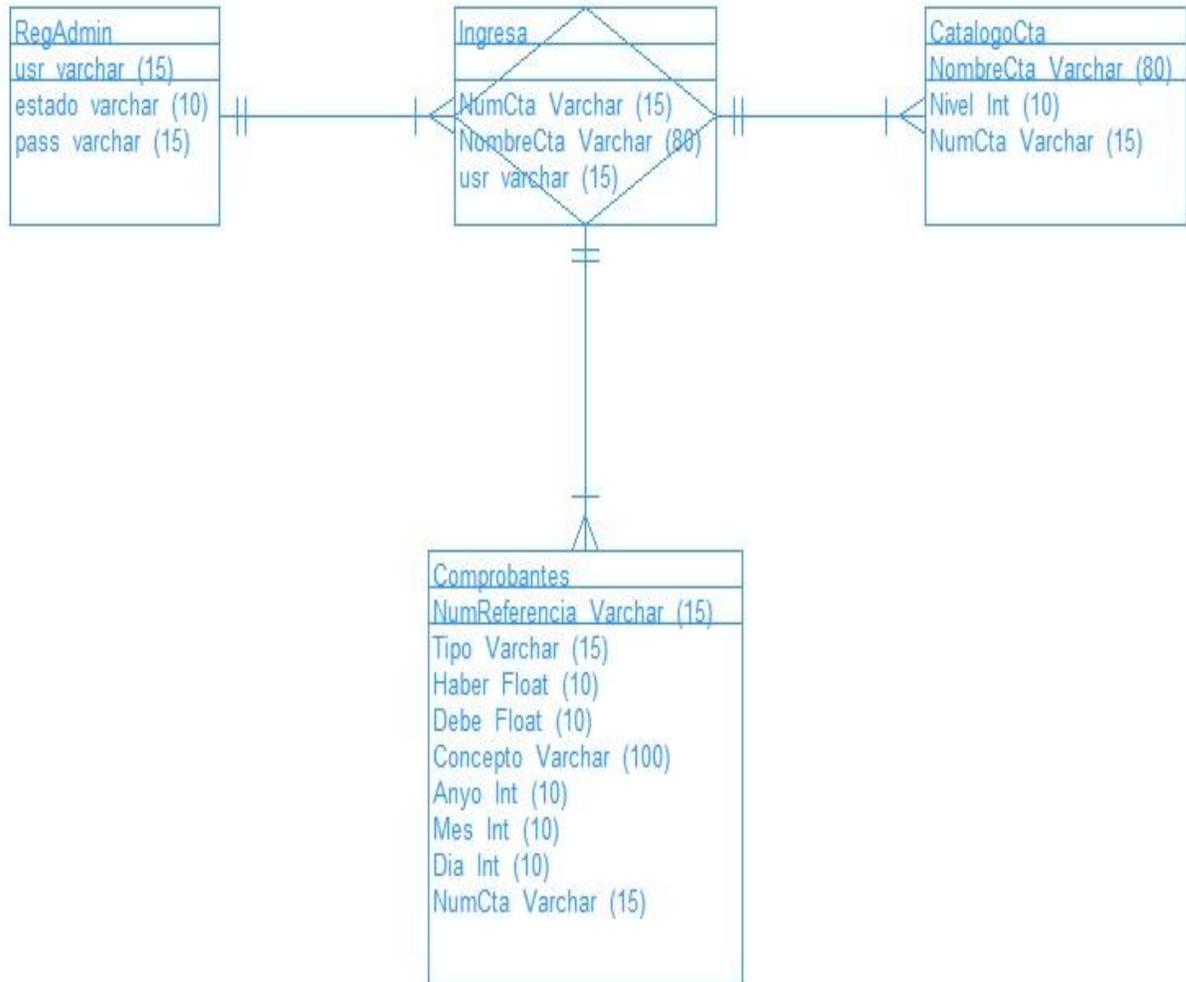
CatalogoCta

| Atributos | Tipo de dato | Valor | Referencia  |
|-----------|--------------|-------|-------------|
| NombreCta | Varchar      | 80    | Primary key |
| NumCta    | Varchar      | 15    | Not Null    |
| Nivel     | Int          | 10    | Not Null    |

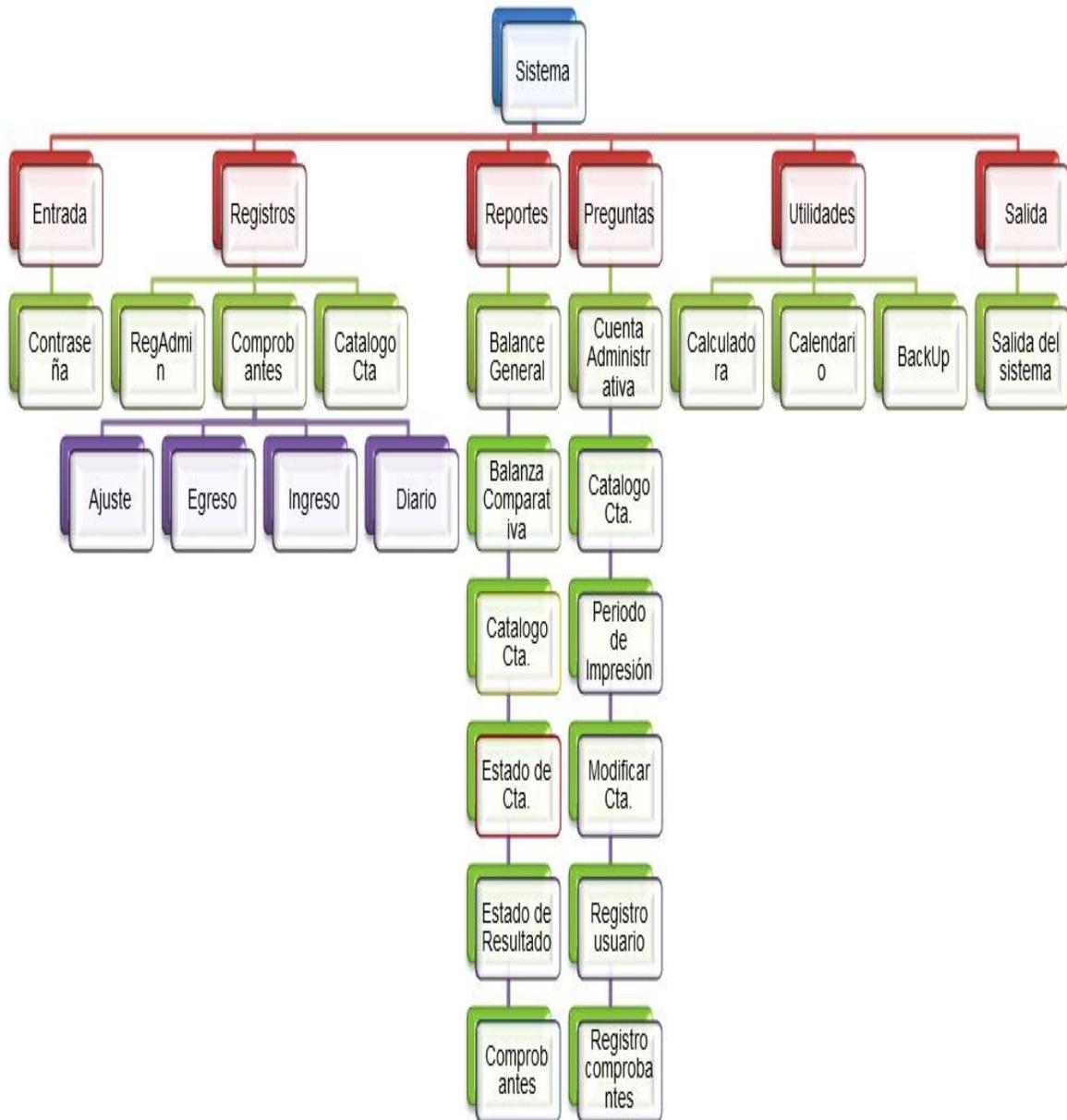
RegAdmin

| Atributos | Tipo de dato | Valor | Referencia  |
|-----------|--------------|-------|-------------|
| usr       | Varchar      | 15    | Primary key |
| pass      | Varchar      | 15    | Not Null    |
| estado    | Varchar      | 10    | Not Null    |

## XI. DIAGRAMA ENTIDAD RELACION



## XII. DISEÑO ARQUITECTONICO



## XIII. ESPECIFICACION DE REQUISITOS DEL SOFTWARE (ERS)

### 1. Introducción

#### 1.1. Propósito

Mostrar los requisitos los cuales tiene que cumplir la aplicación “Elaboración de un sistema contable para la empresa: Cafetalera San José, S.A”.

#### 1.2. Alcance

Las tareas que el sistema podrá realizar son:

❖ Imprimir los 4 tipos de comprobantes que la empresa realiza:

1. Ingresos.
2. Egresos.
3. Diarios.
4. Ajustes.

❖ Realizar copia de seguridad de los datos de la empresa.

❖ Ingresar y modificar contraseña del sistema.

❖ Consultas de cuentas.

❖ Creación de las subcuentas de catalogo.

❖ Creación de las sub\_subcuentas de catalogo.

❖ Creación de las sub\_sub\_subcuentas de catalogo.

❖ Cierre de Mes.

❖ Generación de Reportes:

1. Balance general.
2. Estado de pérdida o ganancia.
3. Balanza comparativa.
4. Catálogo de cuentas.
5. Comprobantes (Egreso, Ingreso, Ajuste y Diario)

### 1.3. Definiciones, acrónimos y abreviaturas

- ✓ **Cuenta:** Es la denominación que se le asigna a personas, bienes, valores, etc., con la finalidad de identificarlos y representarlos. Lo llamaremos **CUENTA**.
- ✓ **Balance general:** Es un estado financiero que se registra a una fecha determinada la situación financiera y económica de la empresa. Es el producto final que se obtiene como consecuencia del registro de los comprobantes de pago en los libros contables y de allí la emisión de los estados contables. Lo llamaremos **BALANCE\_G**.
- ✓ **Informe de Diario:** Es un libro contable donde se recogen, día a día, los hechos económicos de una empresa; es decir en él se registran todas las transacciones realizadas por una empresa. Lo llamaremos **INFORME\_D**
- ✓ **Libro mayor:** Es el libro más importante en cualquier contabilidad, en el que se registran cuentas individuales o colectivas de los bienes materiales, derechos y créditos que integran el activo; las deudas y obligaciones que forman el pasivo; el capital y superávit de los gastos y productos; las ganancias y pérdidas y en general todas las operaciones de una empresa. En dicho libro se registran todas las transacciones anotadas en el diario principal o en los diarios especiales cuando los haya, ya sea particularmente o bien en totales. Lo llamaremos **LIBRO\_M**.
- ✓ **Debe:** Nombre que se da al lado izquierdo de una cuenta del libro mayor. Nombre que se da a la columna de cifras en la que se anotan los cargos. Lo llamaremos **Debe**.
- ✓ **Haber:** Nombre que se da a la columna en la que se anotan las partidas de abono de una cuenta. Lado contrario al "debe" de una cuenta. Columna en la que se anotan los ingresos de un negocio, de una empresa o de una persona. Lo llamaremos **HABER**

- 
- ✓ **Catálogo de cuentas:** Es una lista o numeración pormenorizada y clasificada de los conceptos que integran el Activo, Pasivo y Capital Contable, así como los ingresos y egresos de una entidad económica. La elaboración de este catálogo puede ser de forma Numérica. Consiste en fijar un número progresivo a cada cuenta, debiendo crearse grupos de cuentas de acuerdo con la estructura de los estados financieros. Lo llamaremos **CATLOGO\_C**.
  
  - ✓ **Comprobante de Diario:** Es un documento que debe elaborarse previamente al registro de cualquier operación y en el cual se indica el número, fecha, origen, descripción y cuantía de la operación, así como las cuentas afectadas con el asiento. A cada comprobante se le anexan los documentos y soportes que lo justifiquen. Lo llamaremos **COMPROBANTE\_D**
  
  - ✓ **Contabilidad:** Es la disciplina que sirve para proporcionar información útil en la toma de decisiones. Ha venido siendo definida como Ciencia del Patrimonio, y ello porque su objeto es el estudio y la representación del patrimonio empresarial, tanto desde un punto de vista estático como dinámico. Registra de manera sistemática y cronológica los estados financieros de una empresa. Lo llamaremos **CONTABILIDAD**.
  
  - ✓ **Documento por pagar:** Son título de créditos a cargo del negocio que fueron girados por distintas razones como letras de cambio, pagare, etc. Lo llamaremos **DOCUMENTO\_P**.
  
  - ✓ **Código:** Texto escrito en un lenguaje específico y que puede ser leído por un programador. Debe traducirse a lenguaje máquina para que pueda ser ejecutado por la computadora o para que pueda ser ejecutado por un intérprete. Este proceso se denomina compilación. **CODIGO**.

- 
- ✓
  - ✓ **Cliente:** Un cliente es alguien que compra o alquila algo a un individuo u organización. Lo llamaremos **CLIENTE**.
  
  - ✓ **Empresa:** Es la unidad económico-social, con fines de lucro, en la que el capital, el trabajo y la dirección se coordinan para realizar una. Lo llamaremos **EMPRESA**.
  
  - ✓ **Proveedores:** Persona o sociedad que vende la materia prima utilizada para producir los artículos que se fabriquen. Lo llamaremos **PROVEEDORES**.
  
  - ✓ **Pasivo:** Se refiere sólo a los recursos ajenos (deudas y obligaciones de la empresa respecto a terceros). En sentido amplio, **el Pasivo** engloba los recursos ajenos (Pasivo en sentido estricto) y los recursos propios (Neto: aportación del empresario o de los accionistas y beneficios no distribuidos que quedan en la empresa en forma de reservas). Lo llamaremos **PASIVOS**.
  
  - ✓ **Activo:** Es el conjunto de bienes tangibles o intangibles que posee una empresa. Se considera activo a aquellos bienes que tienen una alta probabilidad de generar un beneficio económico a futuro y se pueda gozar de los beneficios económicos que el bien otorga. Lo llamaremos **ACTIVOS**.
  
  - ✓ **Capital:** El significado del capital, desde el punto de vista financiero, también denominado capital financiero es toda suma de dinero, que no ha sido consumido por su propietario, sino que ha sido ahorrada y trasladada a un mercado financiero con el fin de obtener una renta posteriormente. Lo llamaremos **CAPITAL**.

- ✓ **Equipos de Oficina o Mobiliario:** Llámese así, a todos y/o cada uno de las maquinarias que contribuyen de una manera u otra en el seguimiento del trabajo realizado en un negocio. Lo llamaremos **EQUIPOS\_OM**.

#### 1.4. Referencia

Sistema para el manejo contable de la empresa Cafetalera San José, S.A. Con el mismo se mantendrá un mejor reporte de las actividades contables de la empresa. La elaboración de este sistema es gracias al apoyo que el personal de economía de la empresa Cafetalera San José, S.A nos brindaron.

#### 1.5. Visión general

Se hará una descripción muy general del sistema que pretendemos desarrollar para luego ver detalladamente o estudiar cada uno de los requisitos específicos los cuales se tendrán que cumplir.

## 2. Descripción general

### 2.1. Relaciones del producto

La aplicación estará relacionada con módulos, es decir; será una aplicación la cual se compone o la cual tendrá varios módulos.

### 2.2. Funciones del producto

El sistema debe contener todas las tareas contables que se realizan en toda empresa, estas son:

- Cuando el usuario ejecute el sistema por primera vez, deberá registrar su **CATALOGO DE CUENTAS**.
- Cuando inicie el sistema por primera vez el usuario deberá registrarse para poder **ACCEDER AL SISTEMA**

- Al iniciar el sistema el usuario podrá **INGRESAR O MODIFICAR CONTRASEÑA**, especificando su contraseña actual y estableciendo su nueva contraseña.
- Cuando el usuario solicite el ingreso de un comprobante deberá especificar el tipo de comprobante **TIPO DE COMPROBANTES** (Ingresos, Egresos, Ajustes, Diarios) con sus correspondientes especificaciones presentadas en pantalla.
- Cuando el usuario solicite la **BUSQUEDA DE COMPROBANTES** deberá ingresar un número de referencia.
- Cuando el usuario solicite mostrar el saldo de una cuenta, este deberá seleccionar el número de cuenta.
- Cuando el usuario solicite generar informes este deberá ingresar información referente al período que cubre el informe.
- Para **GENERAR REPORTES DEL SISTEMA**, el usuario deberá seleccionar el tipo de reporte que desea.
  
- Cuando el usuario solicite el reporte de **BALANCE GENERAL**, el sistema lo mostrara y se podrá imprimir en papel si el usuario así lo desee.
- Cuando el usuario solicite el reporte de **ESTADO DE PERDIDA O GANANCIA**, el sistema lo mostrara y se podrá imprimir en papel si el usuario así lo desee.
- Cuando el usuario solicite el reporte de **BALANZA COMPARATIVA**, el sistema lo mostrara y se podrá imprimir en papel si el usuario así lo desee.
- Cuando el usuario solicite el reporte de **INFORME DE DIARIO**, el sistema lo mostrara y se podrá imprimir en papel si el usuario así lo desee.
- Cuando el usuario solicite el reporte de **CATALOGO DE CUENTAS**, el sistema lo mostrara y se podrá imprimir en papel si el usuario así lo desee.
  
- Cuando el usuario solicite el reporte de **INGRESOS**, el sistema lo mostrara y se podrá imprimir en papel si el usuario así lo desee.
- Cuando el usuario solicite el reporte de **EGRESOS**, el sistema lo mostrara y se podrá imprimir en papel si el usuario así lo desee.
- Cuando el usuario solicite el reporte de **AJUSTE**, el sistema lo mostrara y se podrá imprimir en papel si el usuario así lo desee.

### 2.3. Características del usuario

El usuario final de este sistema, será una persona que debe tener conocimientos informáticos, de base de datos, y lo mas importante es que tenga conocimientos avanzados en contabilidad, tener consigo todos los conocimientos antes mencionados, le servirá para tener un mayor uso del sistema.

### 2.4. Restricciones generales

- ✓ La plataforma en la que se trabajara es Windows XP o superior.
- ✓ La capacidad del Disco Duro será de 10 GB o más.
- ✓ Dispositivo de lectura (CD ROM).
- ✓ Lenguaje de programación; Microsoft Visual Studio 2008, el lenguaje de programación empleado C# 2008.
- ✓ Gestor de base de datos, SQL-Server 2008.

### 2.5. Suposiciones y dependencia

Durante las entrevistas iniciales, se hizo énfasis, en que cuando el cliente el cual es el encargado del área de contabilidad de la empresa Cafetalera San José, S.A. Desea modificar el sistema, se abocara a los creadores del mismo. No se incluye muerte del software.

## 3. Requisitos específicos

### 3.1 Requisitos funcionales

#### 3.1.1. Catálogo de Cuentas

##### 3.1.1.1 Especificación

##### 3.1.1.1.1 Introducción

Este proceso permitirá la realización de catálogos de cuentas, esto se hace con el objetivo de facilitar la elaboración de estados financieros. Estas cuentas serán almacenadas en la Base de datos.

### 3.1.1.1.2. Entrada

Dato proporcionado por el sistema:

- ✓ Fecha

Dato proporcionado por el usuario:

- ✓ Número de cuenta
- ✓ Nombre de cuenta
- ✓ Nivel

### 3.1.1.1.3. Proceso

Se mostrara en pantalla una interfaz la cual permitirá introducir datos, para la creación de catálogos de cuentas. El dato proporcionado por el sistema es:

- **Fecha:** Este dato es muy importante a la hora de querer consultar algún catalogo de cuenta hecho antes.
- **Número de cuenta:** El número de cuenta permite llevar un orden de todas las cuentas.
- **Nombre de cuenta:** Permite saber a que cuenta se está refiriendo.
- **Nivel:** Es la categoría que se tiene en el catalogo de cuenta.

### 3.1.1.1.4. Salida

Con todos los datos antes mencionados será suficiente para crear un catalogo de cuenta y la cual se podrá almacenar en la base de datos BBDD\_SCCSJ.

## 3.1.1.2. Interfaces externas

### 3.1.1.2.1. Interfaces de usuario

La captura de datos se hará de forma interactiva por pantalla.

### 3.1.1.2.2. Interfaces hardware

Se podrá utilizar cualquier ordenador capaz de soportar la instalación del sistema.

### 3.1.1.2.3. Interfaces software

El proceso interactúa con la base de datos de ficheros maestros.

### 3.1.1.2.4. Interfaces de comunicaciones

No existe una interfaz de comunicación en este sistema.

### **3.1.2. Acceder al sistema**

#### **3.1.2.1 Especificación**

##### **3.1.2.1.1 Introducción**

Este proceso permitirá introducir la contraseña el cual será requisito para poder acceder al sistema.

##### **3.1.2.1.2. ENTRADA**

Dato proporcionado por el usuario:

- ✓ Usuario
- ✓ Contraseña

##### **3.1.2.1.3. Proceso**

Se mostrara en pantalla una interfaz, la cual permitirá ingresar el usuario y la contraseña, la cual será requisito para poder utilizar en sistema.

- **Usuario:** Cadena de caracteres pertenecientes a una persona capaz de darle privilegio de acceder al sistema.
- **Contraseña:** Cadena de caracteres cifrada pertenecientes a una persona capaz de darle privilegio de acceder al sistema.

##### **3.1.2.1.4. Salida**

Los datos antes mencionados serán suficientes para acceder al sistema.

#### **3.1.2.2. Interfaces externas**

##### **3.1.2.2.1. Interfaces de usuario**

La previa validación de la autenticidad del administrador debe realizarse de manera interactiva por medio del sistema.

##### **3.1.1.2.1. Interfaces hardware**

Se podrá utilizar cualquier ordenador capaz de soportar la instalación del sistema.

##### **3.1.1.2.1. Interfaces software**

El proceso interactúa con la base de datos de ficheros maestros.

### 3.1.1.2.1. Interfaces de comunicaciones

No existe una interfaz de comunicación en este sistema.

## 3.1.3. Ingresar o modificar contraseña

### 3.1.3.1 Especificación

#### 3.1.3.1.1 Introducción

Este proceso permitirá al usuario adquirir una nueva contraseña, la cual permitirá ejecutar el sistema por primera vez, también se podrán modificar si hay alguna contraseña existente.

#### 3.1.3.1.2. Entrada

Dato proporcionado por el usuario:

- ✓ Usuario
- ✓ Contraseña

#### 3.1.3.1.3. Proceso

Se mostrara en pantalla una interfaz, la cual permitirá ingresar el usuario y la contraseña, la cual será requisito para poder utilizar en sistema.

- **Usuario:** Cadena de caracteres pertenecientes a una persona capaz de darle privilegio de acceder al sistema.
- **Contraseña:** Cadena de caracteres cifrada pertenecientes a una persona capaz de darle privilegio de acceder al sistema.

#### 3.1.3.1.4. Salida

Los datos antes mencionados serán suficientes para acceder al sistema.

### 3.1.3.2. Interfaces externas

#### 3.1.3.2.1. Interfaces de usuario

La manipulación y actualización de datos se hará de forma interactiva por pantalla.

#### 3.1.1.2.1. Interfaces hardware

Se podrá utilizar cualquier ordenador capaz de soportar la instalación del sistema.

#### **3.1.1.2.1. Interfaces software**

El proceso interactúa con la base de datos de ficheros maestros.

#### **3.1.1.2.1. Interfaces de comunicaciones**

No existe una interfaz de comunicación en este sistema.

### **3.1.4. Tipo de comprobante**

#### **3.1.4.1 Especificación**

##### **3.1.4.1.1 Introducción**

En este proceso le permitirá al usuario ingresar comprobantes como Ingresos, Egresos, Ajustes y Diarios los cuales son partes de un asiento Diario.

##### **3.1.4.1.2. Entrada**

Dato proporcionado por el sistema:

- ✓ Fecha

Dato proporcionado por el usuario:

- ✓ Ingresos
- ✓ Egresos
- ✓ Ajustes
- ✓ Diarios

##### **3.1.4.1.3. Proceso**

Se mostrara en pantalla una interfaz, la cual permitirá ingresar los tipos de comprobantes que el usuario desee.

- **Ingresos:** Son las entradas que se realicen en la empresa cada día, por ejemplo las ventas.
- **Egresos:** Son los gastos que se realicen en la empresa cada día, por ejemplo las compras de papelería, fertilizantes etc.
- **Ajustes:** Cierre de mes.
- **Diario:** Son los depósitos que se hacen en bancos u organizaciones.

#### **3.1.4.1.4. Salida**

Con todos los datos antes mencionados, bastara para poder realizar los tipos de comprobantes los cuales serán almacenados en la base de dato.

#### **3.1.4.2. Interfaces externas**

##### **3.1.4.2.1. Interfaces de usuario**

La manipulación de datos se hará de forma interactiva por pantalla.

##### **3.1.1.2.1. Interfaces hardware**

Se podrá utilizar cualquier ordenador capaz de soportar la instalación del sistema.

##### **3.1.1.2.1. Interfaces software**

El proceso interactúa con la base de datos de ficheros maestros.

##### **3.1.1.2.1. Interfaces de comunicaciones**

No existe una interfaz de comunicación en este sistema.

### **3.1.5. Búsqueda de comprobante**

#### **3.1.5.1 Especificación**

##### **3.1.5.1.1 Introducción**

En este proceso se le permitirá al usuario hacer una búsqueda de comprobantes por medio del número de referencia del comprobante, para poder realizar informes.

##### **3.1.5.1.2. Entrada**

Dato proporcionado por el usuario:

✓ Número de referencia

##### **3.1.5.1.3. Proceso**

Se mostrara en pantalla una interfaz, la cual permitirá ingresar la fecha para poder hacer la búsqueda en la base de dato de los de comprobantes realizados en esa fecha.

##### **3.1.5.1.4. Salida**

Con todos los datos antes mencionados, bastara para poder ver y palpar en pantalla todos los comprobantes realizados en una fecha estipulada, la cual será introducida por el usuario.

#### **3.1.5.2. Interfaces externas**

##### **3.1.5.2.1. Interfaces de usuario**

---

La manipulación de datos se hará de forma interactiva por pantalla.

#### **3.1.1.2.1. Interfaces hardware**

Se podrá utilizar cualquier ordenador capaz de soportar la instalación del sistema.

#### **3.1.1.2.1. Interfaces software**

El proceso interactúa con la base de datos de ficheros maestros.

#### **3.1.1.2.1. Interfaces de comunicaciones**

No existe una interfaz de comunicación en este sistema.

### **3.1.6. Mostrar saldo de cuenta**

#### **3.1.6.1 Especificación**

##### **3.1.6.1.1 Introducción**

En este proceso se presenta por pantalla el saldo de una cuenta. El saldo de cuenta es la diferencia entre el DEBE y el HABER de una cuenta.

##### **3.1.6.1.2. Entrada**

Dato proporcionado por el usuario:

- ✓ Número de cuenta.

##### **3.1.6.1.3. Proceso**

Este proceso realizará una búsqueda de saldo de cuenta, una vez que el usuario introduzca el número de cuenta la cual pertenecerá el saldo buscado. Esta cuenta estará almacenada en la base de datos y su saldo de cuenta.

##### **3.1.6.1.4 Salida**

Con todos los datos antes mencionados, bastará para poder ver y palpar en pantalla el saldo de cuenta de una cuenta.

#### **3.1.6.2. Interfaces externas**

##### **3.1.6.2.1. Interfaces de usuario**

El tratamiento de datos se hará de forma interactiva por pantalla.

#### **3.1.1.2.1. Interfaces hardware**

Se podrá utilizar cualquier ordenador capaz de soportar la instalación del sistema.

### 3.1.1.2.1. Interfaces software

El proceso interactúa con la base de datos de ficheros maestros.

#### 3.1.1.2.1. Interfaces de comunicaciones

No existe una interfaz de comunicación en este sistema.

## 3.1.7. Información referente al periodo que cubre el informe

### 3.1.7.1 Especificación

#### 3.1.7.1.1 Introducción

Este proceso le permitirá al usuario introducir dos rangos de fechas (Un periodo), el cual será con el objetivo de realizar un corte a esa fecha y obtener resultados.

#### 3.1.7.1.2. Entrada

Dato proporcionado por el usuario:

- ✓ Fecha de inicio
- ✓ Fecha fin

#### 3.1.7.1.3. Proceso

Se mostrara en pantalla una interfaz la cual permitirá la introducción de una fecha al inicio y una fecha de fin, esta fecha será un dato capaz de devolver un informe de cierre mensual, anual o en el rango establecido. En la base de datos hay diferentes informes con fechas estipuladas, las cuales coincidirán con la fecha de solicitud de cierre mensual.

#### 3.1.7.1.4 Salida

Con el rango de fechas antes mencionadas se podrá obtener la información de los reportes de estados financieros solicitados.

### 3.1.7.2. Interfaces externas

#### 3.1.7.2.1. Interfaces de usuario

La captura de datos se hará de forma interactiva por pantalla.

#### 3.1.1.2.1. Interfaces hardware

Se podrá utilizar cualquier ordenador capaz de soportar la instalación del sistema.

#### 3.1.1.2.1. Interfaces software

El proceso interactúa con la base de datos de ficheros maestros.

#### 3.1.1.2.1. **Interfaces de comunicaciones**

No existe una interfaz de comunicación en este sistema.

### 3.1.8. **Generar reportes del sistema**

#### 3.1.8.1 **Especificación**

##### 3.1.8.1.1 **Introducción**

En este proceso permitirá la realización de diferentes reportes como; Balance General, Estado de pérdida o ganancia, Balanza comparativa, Informe de diario, Catalogo de cuentas, Comprobantes de Egreso, Comprobante de Ingreso y Comprobante de Ajuste. Todo lo antes mencionado se hace con el objetivo de saber cómo marcha la empresa mediante comparaciones con reportes anteriores.

##### 3.1.8.1.2. **Entrada**

Ninguna

##### 3.1.8.1.3. **Proceso**

Se mostrara en pantalla una lista de reportes de los cuales se podrán obtener, al escoger un reporte, con consultas de SQL se podrán recoger datos de diferentes tablas de la base de datos, de acuerdo al reporte que se pretende obtener.

##### 3.1.8.1.4. **Salida**

El formato de salida de cada reporte es conforme el tipo de reporte que se pretende realizar.

#### 3.1.8.a. **Balance general**

##### 3.1.8.a.1 **Especificación**

###### 3.1.8.a.1.1 **Introducción**

Este proceso se hace con el objetivo de mostrar la situación financiera y la denominación más adecuada, mediante de este se obtiene la información valiosa sobre su negocio.

###### 3.1.8.a.1.2. **Entrada**

Fechas de periodo.

###### 3.1.8.a.1.3. **Proceso**

Para realizar este tipo de reporte se harán consultas SQL las cuales son el objetivo de tomar datos almacenados en la base de dato, los cuales son:

- **NombreCta:** Describe el nombre del tipo de cuenta.
- **NumCta:** Describe el número de cuenta.
- **Debe:** Describe todos los gastos o perdidas del negocio.
- **Haber:** Refleja toda la utilidad de la empresa; con lo que cuenta la empresa describe la situación de la empresa.

#### 3.1.8.a.1.4. **Salida**

Con todos los dato antes mencionados el usuario podrá realizar un balance general, pudiendo así imprimirlo si así lo desee.

### 3.1.8.b. **Estado de pérdida o ganancia**

#### 3.1.8.b.1 **Especificación**

##### 3.1.8.b.1.1 **Introducción**

Este documento o reporte muestra ordenadamente y detalladamente la utilidad o pérdida del ejercicio. La primera parte consiste en analizar todos los elementos que entran en la compra de mercancía hasta determinar la utilidad o pérdida del ejercicio en ventas.

##### 3.1.8.b.1.2. **Entrada**

Fechas de periodo.

##### 3.1.8.b.1.3. **Proceso**

Para realizar este tipo de reporte se harán consultas SQL, las cuales son el objetivo de recuperar datos almacenados en la base de dato, los cuales son:

- **Nombre Cuenta:** Describe el nombre de la cuenta.
- **Saldo:** Describe el saldo total de una cuenta en una fecha determinada.

#### 3.1.8.b.1.4. **Salida**

Con todos los dato antes mencionados el usuario podrá realizar un reporte de perdida y ganancia de la empresa, pudiendo así imprimirlo si así lo desee.

#### 3.1.8.c. **Balanza comparativa**

##### 3.1.8.c.1 **Especificación**

###### 3.1.8.c.1.1 **Introducción**

Este documento o reporte muestra ordenadamente y detalladamente la comparación de la situación financiera con respecto a otras fechas.

###### 3.1.8.c.1.2. **Entrada**

Fechas de periodo.

###### 3.1.8.c.1.3. **Proceso**

Para realizar este tipo de reporte se harán consultas SQL, las cuales son el objetivo de recuperar datos almacenados en la base de dato, los cuales son:

- **Nombre Cuenta:** Describe el nombre de la cuenta.
- **Número de cuenta:** Identificación de una cuenta.
- **Debe:** Cuenta acreedora.
- **Haber:** Cuenta deudora.
- **Saldo:** Describe el saldo total de una cuenta en una fecha determinada.

###### 3.1.8.c.1.4. **Salida**

Con todos los dato antes mencionados el usuario podrá realizar un reporte de balanza comparativa, pudiendo así imprimirlo si así lo desee.

#### 3.1.8.d. **Informe diario**

##### 3.1.8.d.1 **Especificación**

###### 3.1.8.d.1.1 **Introducción**

En este proceso muestra una recopilación de la información de las transacciones diarias que se realicen en la empresa. Está orientada a la administración.

#### 3.1.8.d.1.2. **Entrada**

Dato proporcionado por el sistema:

- ✓ Fecha

Dato proporcionado por el usuario:

- ✓ Concepto
- ✓ Nombre de cuenta
- ✓ Tipo Comprobante
- ✓ Número de referencia
- ✓ Número de cuenta
- ✓ Debe
- ✓ Haber

#### 3.1.8.d.1.3. **Proceso**

Para realizar este tipo de reporte el sistema nos brinda la fecha y luego hacer consultas SQL, las cuales son el objetivo de recuperar datos almacenados en la base de dato, que son:

- **Nombre Cuenta:** Describe el nombre de la cuenta.
- **Concepto:** Es una descripción del uso de los comprobantes en función del dinero.
- **Tipo de comprobante:** Describe el tipo de comprobante.
- **Número de referencia:** Número único que describe a un comprobante en común independiente del tipo de comprobante.
- **Número de cuenta:** Describe el número de la cuenta.
- **Debe:** Son los ingresos de un negocio, de la empresa. Utilidad de la empresa.
- **Haber:** Describe todos los gastos o pérdidas del negocio.

#### 3.1.8.d.1.4. **Salida**

Una vez almacenados los datos, el sistema mostrara el informe sobre asiento de diario, el cual podrá guardar en cualquier directorio de su PC o simplemente imprimir.

#### 3.1.8.e. **Catalogo de cuenta**

##### 3.1.8.e.1 **Especificación**

###### 3.1.8.e.1.1 **Introducción**

Este proceso permitirá realizar el reporte de catalogo de cuentas que es una lista o numeración pormenorizada y clasificada de los conceptos que integran el Activo, Pasivo y Capital Contable, así como los ingresos y egresos de una entidad económica.

###### 3.1.8.e.1.2. **Entrada**

Dato proporcionado por el sistema:

- ✓ Fecha

Dato proporcionado por el usuario:

- ✓ Nombre de cuenta
- ✓ Número de cuenta
- ✓ Nivel

###### 3.1.8.e.1.3. **Proceso**

Se mostrara en pantalla una interfaz de captación de datos, en donde el usuario podrá ingresar los datos que son necesarios para dicho reporte, como son; Fecha, Nombre de cuenta, Numero de cuenta.

- **Nombre de cuenta:** Dato obligatorio, este dato es el encargado de describir el nombre de la cuenta.
- **Número de cuenta:** Dato obligatorio, este dato es el encargado de describir el número de cuenta.

###### 3.1.8.e.1.4. **Salida**

---

Con todos los datos antes definidos, se podrá realizar y almacenar la cuenta correspondiente en la base de dato BBDD\_SCCSJ.

### 3.1.8.f. Ingresos

#### 3.1.8.f.1 Especificación

##### 3.1.8.f.1.1 Introducción

En este proceso muestra una recopilación de la información de las transacciones diarias que se realicen en la empresa. Está orientada a la administración.

##### 3.1.8.f.1.2. Entrada

Dato proporcionado por el sistema:

- ✓ Fecha

Dato proporcionado por el usuario:

- ✓ Concepto
- ✓ Nombre de cuenta
- ✓ Tipo Comprobante
- ✓ Número de referencia
- ✓ Número de cuenta
- ✓ Debe
- ✓ Haber

##### 3.1.8.f.1.3. Proceso

Para realizar este tipo de reporte el sistema nos brinda la fecha y luego hacer consultas SQL, las cuales son el objetivo de recuperar datos almacenados en la base de dato, que son:

- **Nombre Cuenta:** Describe el nombre de la cuenta.
- **Concepto:** Es una descripción del uso de los comprobantes en función del dinero.
- **Tipo de comprobante:** Describe el tipo de comprobante.
- **Número de referencia:** Número único que describe a un comprobante en común independiente del tipo de comprobante.

- **Número de cuenta:** Describe el número de la cuenta.
- **Debe:** Son los ingresos de un negocio, de la empresa. Utilidad de la empresa.
- **Haber:** Describe todos los gastos o pérdidas del negocio.

#### 3.1.8.f.1.4. **Salida**

Una vez almacenados los datos, el sistema mostrara el informe sobre el comprobante de ingresos, el cual podrá guardar en cualquier directorio de su PC o simplemente imprimir, si así lo desea.

#### 3.1.8.g. **Egresos**

##### 3.1.8.g.1 **Especificación**

###### 3.1.8.g.1.1 **Introducción**

En este proceso se muestra una recopilación de la información de las transacciones diarias que se realicen en la empresa. Está orientada a la administración.

###### 3.1.8.g.1.2. **Entrada**

Dato proporcionado por el sistema:

- ✓ Fecha

Dato proporcionado por el usuario:

- ✓ Concepto
- ✓ Nombre de cuenta
- ✓ Tipo Comprobante
- ✓ Número de referencia
- ✓ Número de cuenta
- ✓ Debe
- ✓ Haber

### 3.1.8.g.1.3. Proceso

Para realizar este tipo de reporte el sistema nos brinda la fecha y luego hacer consultas SQL, las cuales son el objetivo de recuperar datos almacenados en la base de dato, que son:

- **Nombre Cuenta:** Describe el nombre de la cuenta.
- **Concepto:** Es una descripción del uso de los comprobantes en función del dinero.
- **Tipo de comprobante:** Describe el tipo de comprobante.
- **Número de referencia:** Número único que describe a un comprobante en común independiente del tipo de comprobante.
- **Número de cuenta:** Describe el número de la cuenta.
- **Debe:** Son los ingresos de un negocio, de la empresa. Utilidad de la empresa.
- **Haber:** Describe todos los gastos o pérdidas del negocio.

### 3.1.8.g.1.4. Salida

Una vez almacenados los datos, el sistema mostrara el informe sobre el comprobante de egresos, el cual podrá guardar en cualquier directorio de su PC o simplemente imprimir, si así lo desea.

## 3.1.8.h. Ajuste

### 3.1.8.h.1 Especificación

#### 3.1.8.h.1.1 Introducción

En este proceso muestra una recopilación de la información de las transacciones diarias que se realicen en la empresa. Está orientada a la administración.

#### 3.1.8.h.1.2. Entrada

Dato proporcionado por el sistema:

✓ Fecha

Dato proporcionado por el usuario:

- ✓ Concepto
- ✓ Nombre de cuenta
- ✓ Tipo Comprobante
- ✓ Número de referencia
- ✓ Número de cuenta
- ✓ Debe
- ✓ Haber

#### 3.1.8.h.1.3. **Proceso**

Para realizar este tipo de reporte el sistema nos brinda la fecha y luego hacer consultas SQL, las cuales son el objetivo de recuperar datos almacenados en la base de dato, que son:

- **Nombre Cuenta:** Describe el nombre de la cuenta.
- **Concepto:** Es una descripción del uso de los comprobantes en función del dinero.
- **Tipo de comprobante:** Describe el tipo de comprobante.
- **Número de referencia:** Número único que describe a un comprobante en común independiente del tipo de comprobante.
- **Número de cuenta:** Describe el número de la cuenta.
- **Debe:** Son los ingresos de un negocio, de la empresa. Utilidad de la empresa.
- **Haber:** Describe todos los gastos o pérdidas del negocio.

#### 3.1.8.h.1.4. **Salida**

---

Una vez almacenados los datos, el sistema mostrara el informe sobre el comprobante de ajuste, el cual podrá guardar en cualquier directorio de su PC o simplemente imprimir, si así lo desea.

### 3.1.8.2. Interfaces externas

#### 3.1.8.2.1. Interfaces de usuario

Todos los datos a introducir se realizaran por pantalla, en donde habrá una interacción entre interfaces.

#### 3.1.8.2.2. Interfaces hardware

Se podrá utilizar cualquier ordenador capaz de soportar la instalación del sistema.

#### 3.1.8.2.3. Interfaces software

Las interfaces de las cuales estará compuesto la aplicación, tendrán relación con interfaces del sistema operativo.

#### 3.1.8.2.4. Interfaces de comunicaciones

No existe una interfaz de comunicación en este sistema.

## 3.2. Requisitos de funcionamiento

- **Requisitos estáticos:** Únicamente un sólo usuario puede hacer uso de la aplicación, por limitaciones de personal y como medida de seguridad.
- **Requisitos dinámicos:** El tiempo de respuesta del sistema deberá darse en plazos adecuados.

## 3.3. Restricciones de diseño

Cada interfaz del sistema deberá contener controles necesarios para el buen funcionamiento del mismo, habrá una interfaz base, es decir será la que tendrá controles que permitirán tener acceso a otras interfaces.

### **3.4. Atributos**

#### **3.4.1. Seguridad**

El sistema estará protegido. En el sistema habrá un usuario que tendrá privilegio de crear una cuenta de acceso, sin la misma, el sistema no podrá ser utilizado.

#### **3.4.2. Mantenimiento**

En caso que se haga algún cambio los cuales produzcan alteraciones a los requisitos que inicialmente se plantearon deberán ser reflejados en este documento.

#### **3.4.3. Ayuda**

Este manual cuenta con las características fundamentales del funcionamiento del sistema SCCSJ, es útil como guía rápida. Todo el personal está totalmente capacitado para el manejo de este sistema.

### **3.5. Otros requisitos**

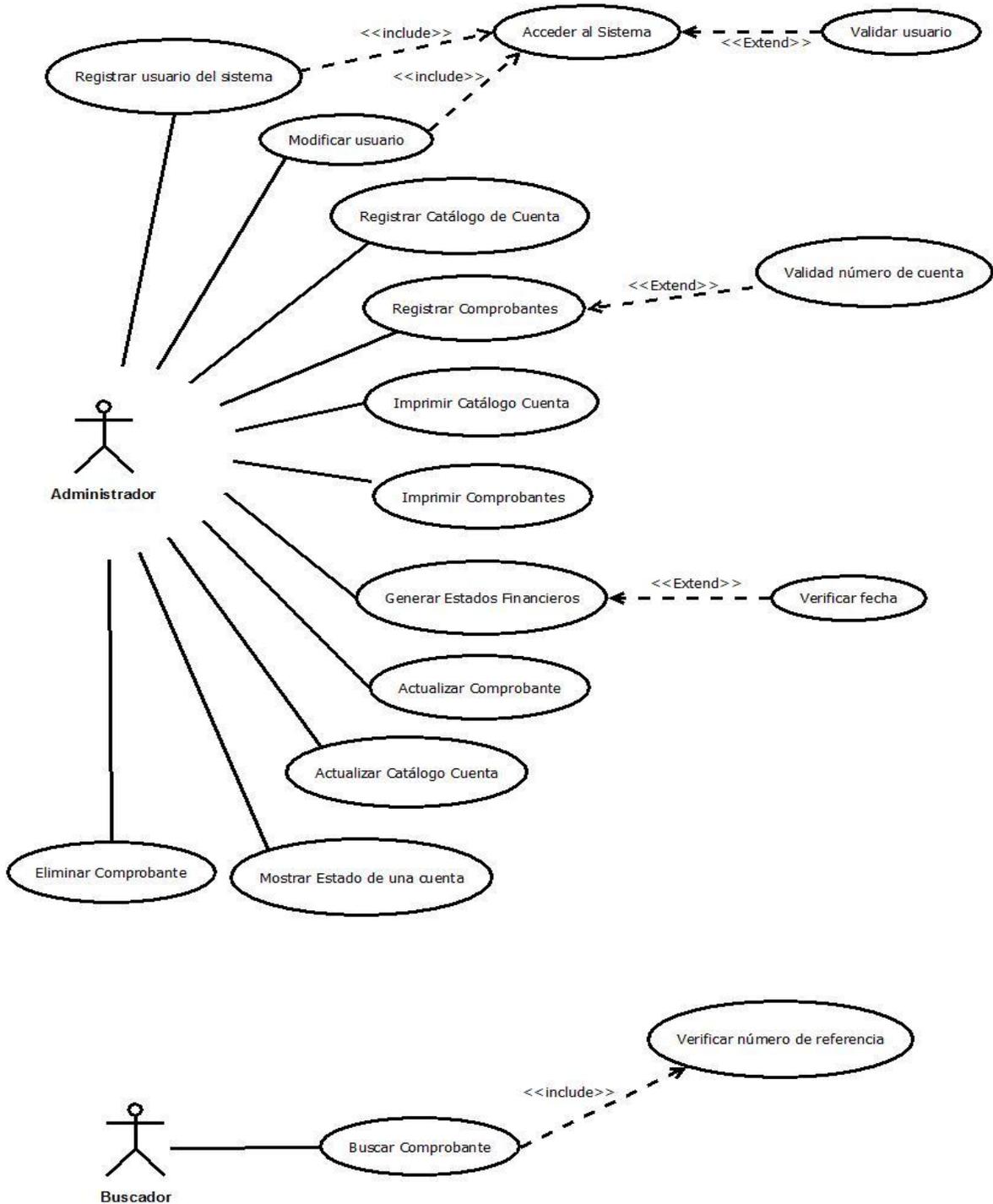
#### **3.5.1. Bases de datos**

Habrà una Base Datos llamada BBDD\_SCCSJ en la cual se almacenara la informaci3n contable de la empresa Cafetalera San Jos3, S.A.

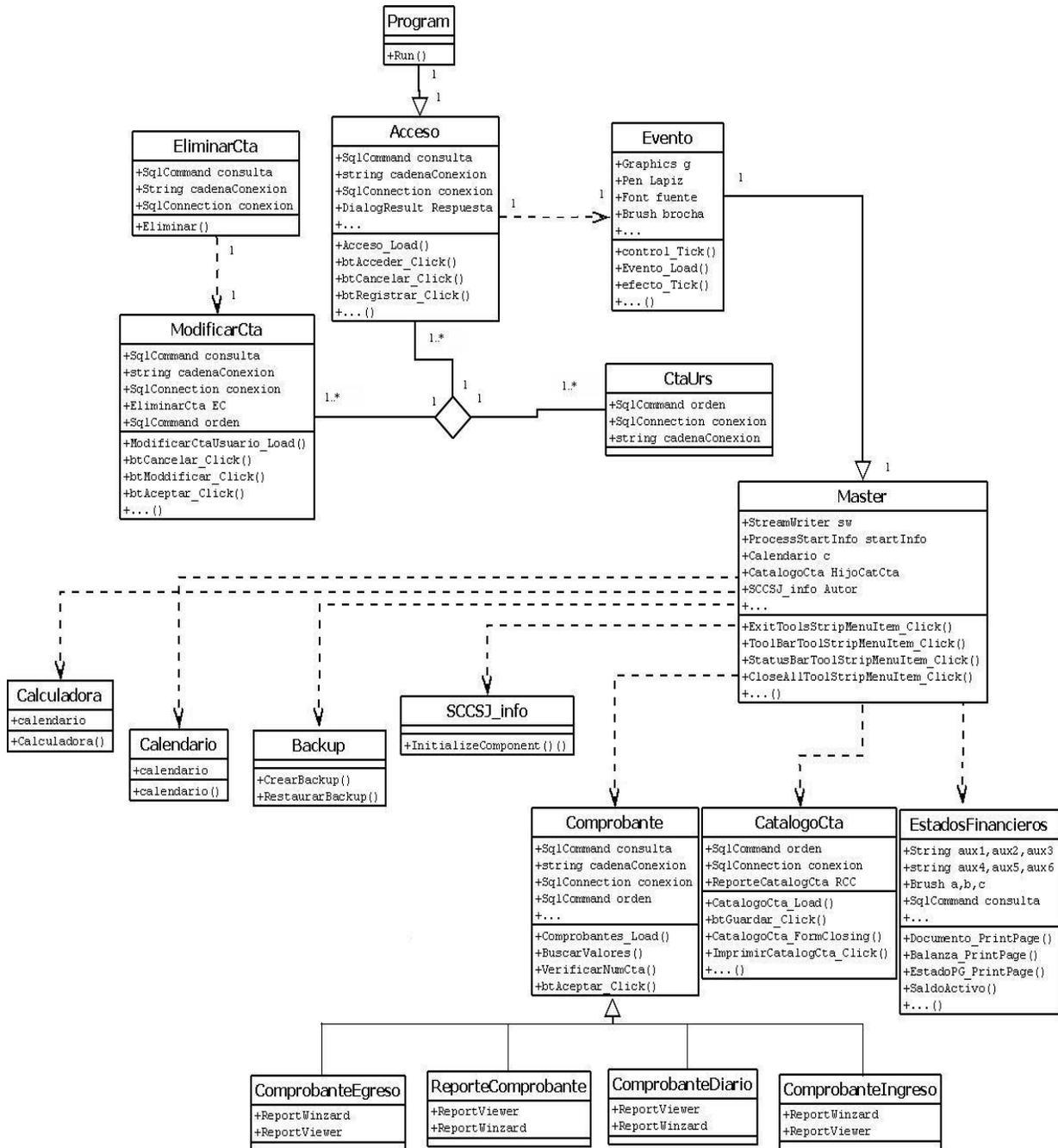
#### **3.5.2. Operaciones**

Todas las operaciones que se realizaran van a estar relacionadas con la Base de Datos, las mismas ser3n de acuerdo al resultado esperado.

### XIV. DIAGRAMA DE CASO DE USO

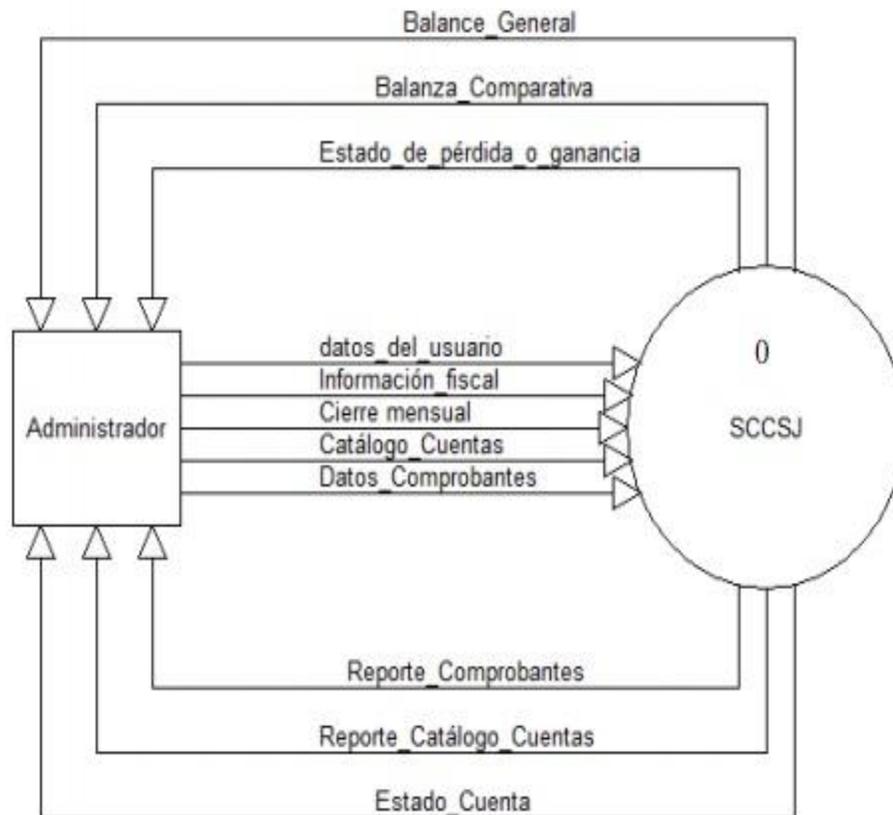


### XV. Diagrama de Clases

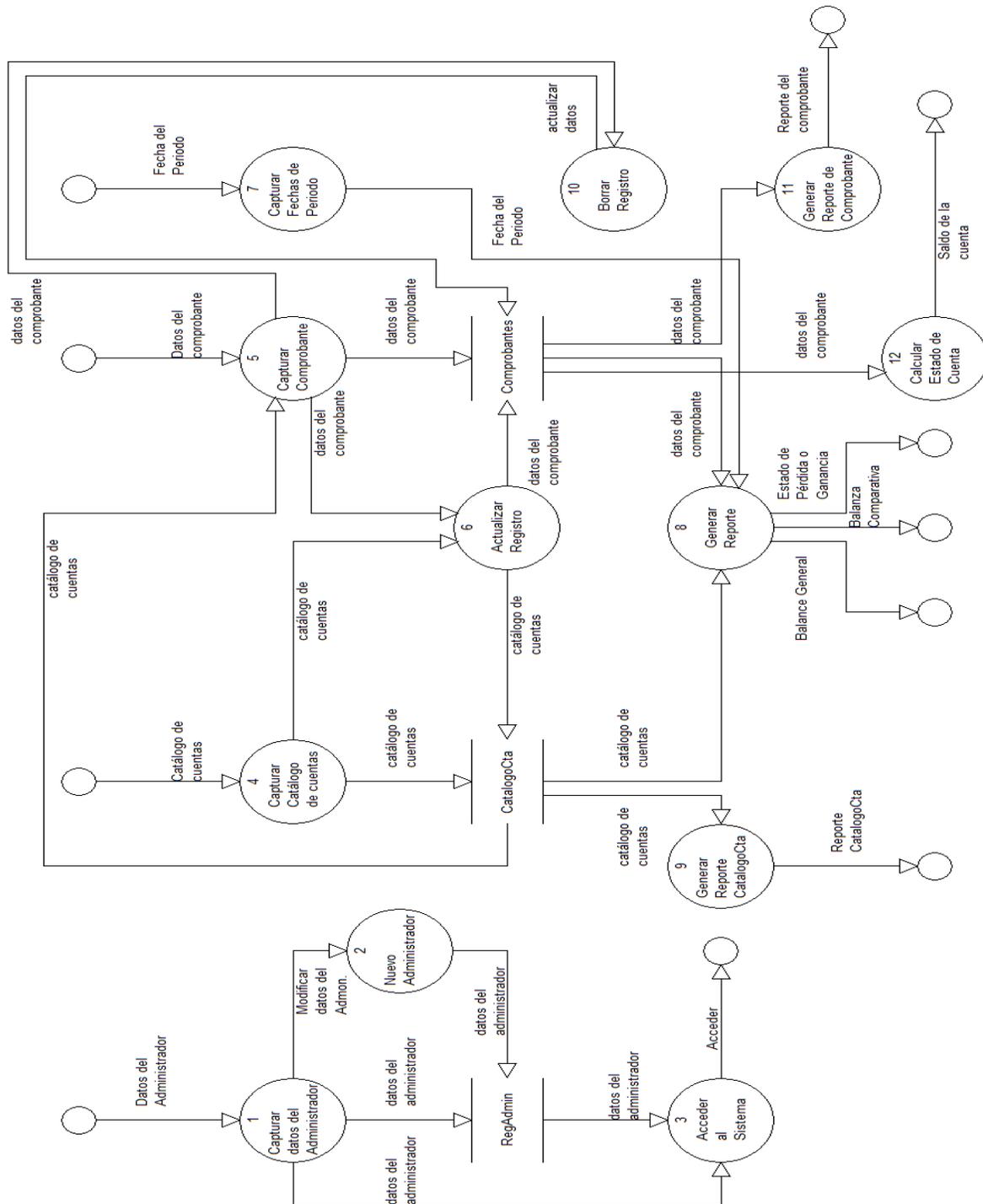


## XVI. DIAGRAMA DE FLUJO DE DATOS

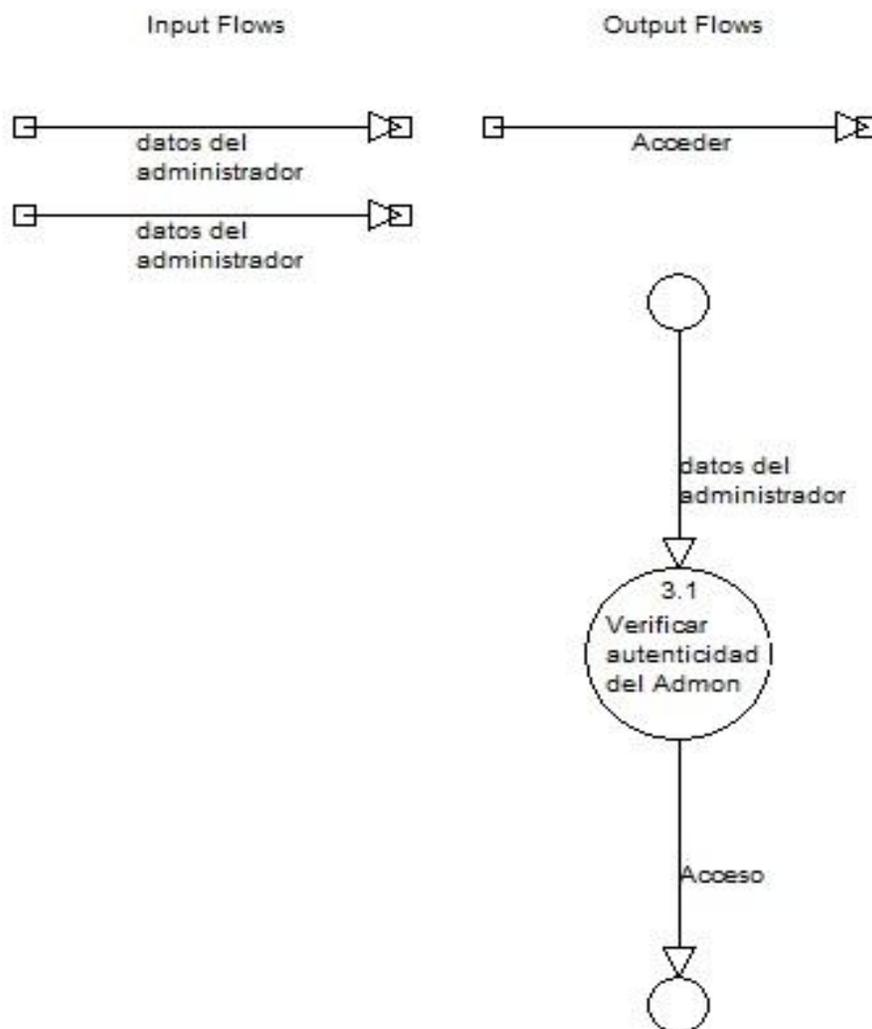
### Nivel 0

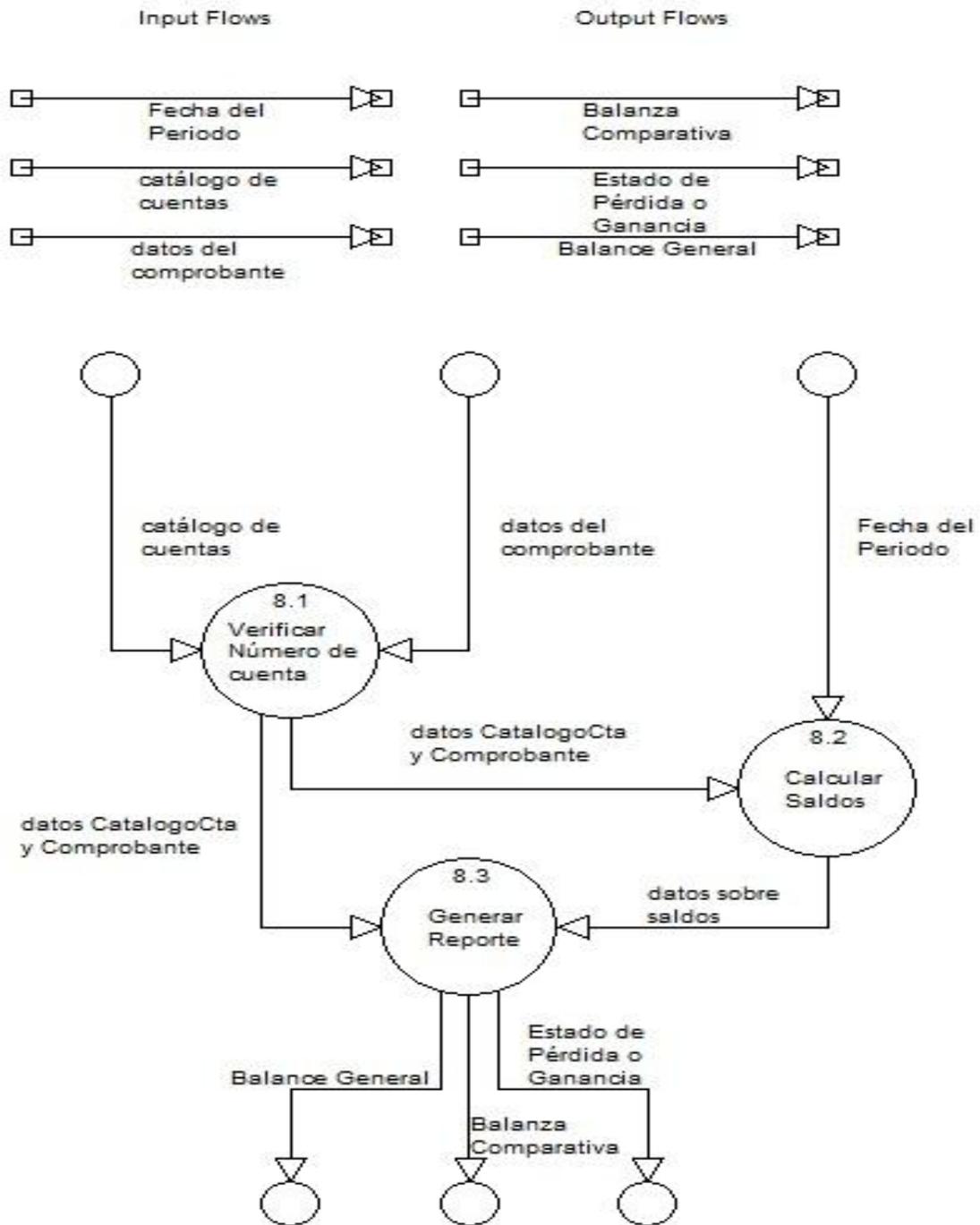


# Nivel 1

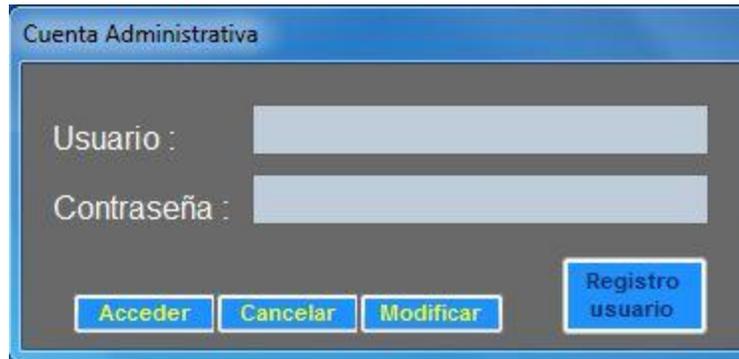


## Nivel 2





## XVII. CODIFICACIÓN

The image shows a Windows-style dialog box titled "Cuenta Administrativa". It has a dark gray background with a blue border. There are two text input fields: the first is labeled "Usuario :" and the second is labeled "Contraseña :". Below the input fields, there are four buttons: "Acceder", "Cancelar", "Modificar", and "Registro usuario". The "Registro usuario" button is positioned to the right of the other three.

**Figura 12.1.** Formulario de acceso al sistema general

Este Formulario pertenece a la clase Acceso y nos permite entre otras cosas validar y dar acceso al Formulario Master o General, además de llamar a diferentes formularios que nos ayudarán a crear una cuenta de administrador o modificar sus datos.

```
private void Acceso_Load(object sender, EventArgs e)
{
try
{
SqlCommand consulta;
string cadenaConexion;
SqlConnection conexion;

cadenaConexion = "Data Source=.\sqlexpress;" + "Initial
Catalog=BBDD_SCCSJ;Integrated Security=True";
conexion = new SqlConnection(cadenaConexion);
conexion.Open();

consulta = new SqlCommand("select * from RegAdmin", conexion);

lector = consulta.ExecuteReader();

while (lector.Read())
{
```

```
if (lector["estado"].ToString() == "Activado")
    btRegistrar.Enabled = false;
}

lector.Close();
}
catch (Exception c)
{
    MessageBox.Show(c.Message.ToString());
}
}

private void btAcceder_Click(object sender, EventArgs e)
{
    try
    {
        SqlCommand consulta;
        string cadenaConexion;
        SqlConnection conexion;

        cadenaConexion = "Data Source=.\sqlexpress;" + "Initial
        Catalog=BBDD_SCCSJ;Integrated Security=True";
        conexion = new SqlConnection(cadenaConexion);
        conexion.Open();

        consulta = new SqlCommand("select * from RegAdmin", conexion);

        lector = consulta.ExecuteReader();

        if (ctUsr.Text.Length > 0)
        {
            if (ctPass.Text.Length > 0)
            {
                while (lector.Read())
                {
                    if (lector["usr"].ToString() == ctUsr.Text)
                    {
                        if (lector["pass"].ToString() == ctPass.Text)
                        {
                            Evento F = new Evento();
                            F.Show();
                            this.Hide();
                            correcto = true;
                        }
                    }
                }
            }
        }
    }
}
```

```
}  
  
}  
}  
}  
  
if (!correcto)  
{  
    MessageBox.Show("Se deben completar los datos y asegurarse que sean correctos",  
        "Autenticación de datos", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);  
  
    contador = contador + 1;  
  
    if (contador == 3)  
    {  
  
        MessageBox.Show("Se han terminado las oportunidades, gracias", "Seguridad de datos",  
            MessageBoxButtons.OK, MessageBoxIcon.Stop);  
  
        Close();  
    }  
  
    ctUsr.Clear();  
    ctPass.Clear();  
}  
lector.Close();  
}  
catch (Exception c)  
{  
    MessageBox.Show(c.Message.ToString());  
}  
}  
  
private void btCancelar_Click(object sender, EventArgs e)  
{  
    ctUsr.Clear();  
    ctPass.Clear();  
    DialogResult Respuesta;  
    Respuesta = MessageBox.Show("¿Desea salir de la aplicación?", "Cuenta de usuario",  
        MessageBoxButtons.YesNo, MessageBoxIcon.Exclamation);  
  
    if (Respuesta == DialogResult.Yes)  
    {
```

```

Close();
}
}

private void btRegistrar_Click(object sender, EventArgs e)
{
    ctUsr.Clear();
    ctPass.Clear();
    CtaUsr RCU = new CtaUsr();
    btRegistrar.Enabled = RCU.Respuesta;
    RCU.Show();
}

```

**Figura 12.2.** Formulario para la modificación de datos de una cuenta administrativa existente.

Este formulario pertenece a la clase `ModificarCtaUsuario` y el objetivo de este es poder hacer modificaciones a una cuenta administrativa ya existente, por lo que se deberá comprobar su existencia y posteriormente ingresaremos los nuevos datos de nuestra cuenta administrativa.

```

private void ModificarCtaUsuario_Load(object sender, EventArgs e)
{
    NuevaCta.Enabled = false;
    btAceptar.Enabled = false;
    DatosActuales.Enabled = true;
}

private void btModdificar_Click(object sender, EventArgs e)
{

```

```
try
{
    SqlCommand consulta;
    string cadenaConexion;
    SqlConnection conexion;

    if (ctUsr.Text.Length == 0)
    {
        if (ctNuevoUsr.Text.Length == 0)
        {
            if (ctPass.Text.Length == 0)
            {
                if (ctNuevoPass.Text.Length == 0)
                {
                    MessageBox.Show("Para realizar el cambio de su cuenta, debe rellenar
todos los campos", "Acceso denegado", MessageBoxButtons.OK, MessageBoxIcon.Error);
                    return;
                }
            }
        }
    }

    cadenaConexion = "Data Source=.\sqlexpress;" + "Initial
Catalog=BBDD_SCCSJ;Integrated Security=True";
    conexion = new SqlConnection(cadenaConexion);
    conexion.Open();

    consulta = new SqlCommand("select * from RegAdmin", conexion);

    lector = consulta.ExecuteReader();

    lector.Read();

    if (lector["usr"].ToString() == ctUsr.Text)
    {
        if (lector["pass"].ToString() == ctPass.Text)
        {
            NuevaCta.Enabled = true;
            DatosActuales.Enabled = false;
            btAceptar.Enabled = true;
        }
    }
    else
    {
```

```
        MessageBox.Show("Usuario o Contraseña, no son los datos correctos",
"Acceso denegado", MessageBoxButtons.OK, MessageBoxIcon.Error);
        return;
    }
}

catch (Exception ms)
{
    MessageBox.Show(ms.Message.ToString());
}
}

private void btAceptar_Click(object sender, EventArgs e)
{
    EliminarCta EC = new EliminarCta();
    EC.Eliminar();

    try
    {
        SqlCommand orden;
        SqlConnection conexion;

        string cadenaConexion = "Data Source=.\sqlexpress;" + "Initial
Catalog=BBDD_SCCSJ;Integrated Security=True";

        conexion = new SqlConnection(cadenaConexion);

        orden = new SqlCommand("insert into RegAdmin
values(@usr,@pass,@estado)", conexion);

        orden.Parameters.Add(new SqlParameter("@usr", SqlDbType.VarChar, 15));
        orden.Parameters["@usr"].Value = ctNuevoUsr.Text;

        orden.Parameters.Add(new SqlParameter("@pass", SqlDbType.VarChar, 15));
        orden.Parameters["@pass"].Value = ctNuevoPass.Text;

        orden.Parameters.Add(new SqlParameter("@estado", SqlDbType.VarChar, 10));
        orden.Parameters["@estado"].Value = "Activado";

        MessageBox.Show("Datos almacenados correctamente", "Autenticación de
usuario", MessageBoxButtons.OK, MessageBoxIcon.Information);

        orden.Connection.Open();
        orden.ExecuteNonQuery();
        orden.Connection.Close();
    }
}
```

```

        btAceptar.Enabled = false;
        Close();
    }

    catch (Exception ms)
    {
        MessageBox.Show(ms.Message.ToString());
    }
}

```

**Figura 12.3.** Formulario para la creación de una cuenta administrativa.

Este Formulario pertenece a la clase CtaUrs y consiste en registrar a un único usuario administrativo, el cual tendrá control absoluto del sistema.

```

private void btAceptar_Click(object sender, EventArgs e)
{
    try
    {
        SqlCommand orden;
        SqlConnection conexion;

        string cadenaConexion = "Data Source=.\sqlexpress;" + "Initial
        Catalog=BBDD_SCCSJ;Integrated Security=True";

        conexion = new SqlConnection(cadenaConexion);

        if (ctUsr.Text.Length > 0)
        {
            if (ctPass.Text.Length > 0)

```

```
{
    if (ctConfPass.Text.Length > 0)
    {
        if (ctPass.Text == ctConfPass.Text)
        {
            orden = new SqlCommand("insert into RegAdmin
values(@usr,@pass,@estado)", conexion);

            orden.Parameters.Add(new SqlParameter("@usr",
SqlDbType.VarChar, 15));
            orden.Parameters["@usr"].Value = ctUsr.Text;

            orden.Parameters.Add(new SqlParameter("@pass",
SqlDbType.VarChar, 15));
            orden.Parameters["@pass"].Value = ctPass.Text;

            orden.Parameters.Add(new SqlParameter("@estado",
SqlDbType.VarChar,10));
            orden.Parameters["@estado"].Value = "Activado";

            MessageBox.Show("Datos almacenados correctamente",
"Autenticación de usuario", MessageBoxButtons.OK, MessageBoxIcon.Information);

            orden.Connection.Open();
            orden.ExecuteNonQuery();
            orden.Connection.Close();

            Close();
            Respuesta = false;
        }
        else
            MessageBox.Show("Compruebe que los datos sean correctos",
"Autenticación de datos", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}
else
    MessageBox.Show("Para poder acceder al sistema debe registrarse", "Acceso
al sistema", MessageBoxButtons.OK, MessageBoxIcon.Hand);
}

catch (Exception c)
{
    MessageBox.Show(c.Message.ToString());
}
```

```

    }
}

```



**Figura 12.4.** Presentación del nombre del sistema con efectos de animación.

Este es un formulario que pertenece a la clase Evento, nos muestra el nombre del sistema SCCSJ (Sistema Contable Cafetalera San José), el cual está controlado por un temporizador y está dotado de una pequeña animación de efecto visual.

```

private void efecto_Tick(object sender, EventArgs e)
{
    Graphics g = CreateGraphics();

    Font fuente = new Font("Times new Roman", 90, FontStyle.Bold,
GraphicsUnit.Point);

    string texto = "SCCSJ";
    SizeF TamTexto = new SizeF(g.MeasureString(texto, fuente));

    PointF TextStart = new PointF(Convert.ToSingle(ClientSize.Width -
TamTexto.Width) / 2, Convert.ToSingle(ClientSize.Height - TamTexto.Height) / 2);

```

```
PointF GradientStart = new PointF(0, 0);
PointF GradientEnd = new PointF(intCurrentGradientShift, 150);
LinearGradientBrush Brocha = new LinearGradientBrush(GradientStart,
GradientEnd, Color.Blue, Color.Brown);
Brush brocha = new HatchBrush(HatchStyle.Sphere, Color.Brown);

Pen Lapiz = new Pen(brocha, 5);

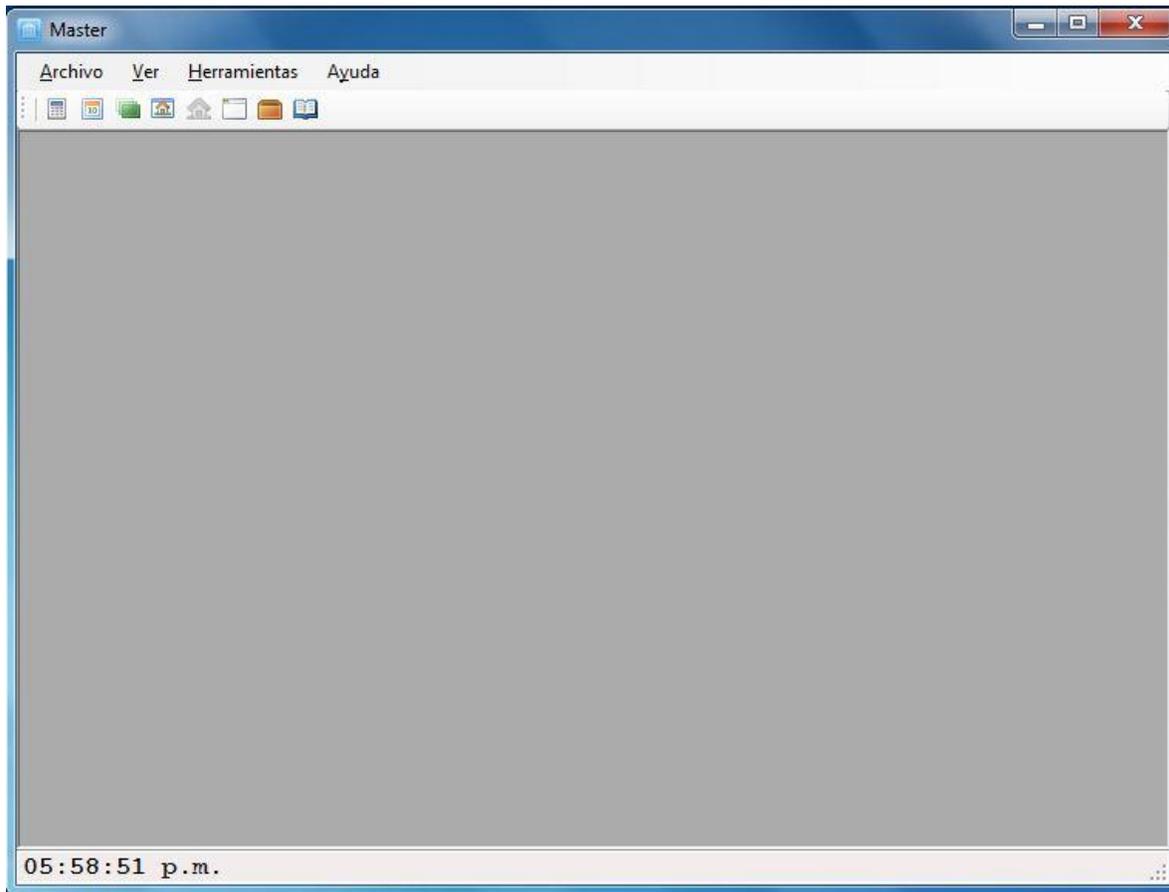
g.DrawEllipse(Lapiz, 40, 40, 460, 200);

g.DrawString(texto, fuente, Brocha, TextStart);

g.Dispose();

intCurrentGradientShift += intGradientStep;

if (intCurrentGradientShift == 500)
{
    intGradientStep = -2;
}
else if (intCurrentGradientShift == -50)
{
    intGradientStep = 2;
}
}
```



**Figura 12.5.** En este formulario master se concentra toda la funcionalidad del sistema SCCSJ.

Este es un Formulario padre que acoge algunos formularios llamados hijos que tienen que ver meramente con el tratamiento de la información sobre el registro contable, este pertenece a la clase Master. Además, posee algunas herramientas de mucha utilidad como la calculadora y el calendario, para conocer más de la aplicación podemos consultar al menú ayuda donde encontrará toda la información que necesita saber.

```
private void CloseAllToolStripMenuItem_Click(object sender, EventArgs e)
{
    foreach (Form childForm in MdiChildren)
    {
        childForm.Close();
    }
}
```

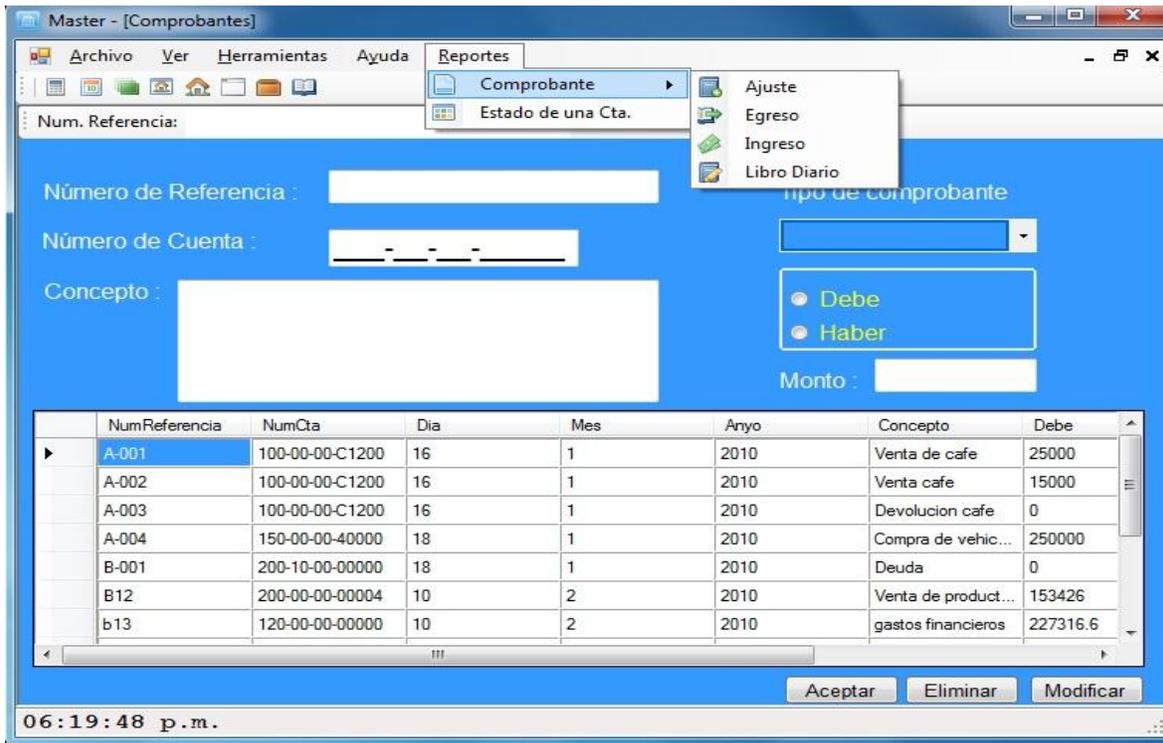
```
}

private void btCalc_Click(object sender, EventArgs e)
{
    try
    {
        if (!System.IO.File.Exists(@"c:\Windows\system32\calc.exe"))
        {
            StreamWriter sw = new StreamWriter(@"c:\windows\system32\calc.exe");
            sw.Close();
        }
        ProcessStartInfo startInfo = new
ProcessStartInfo(@"c:\windows\system32\calc.exe");
        startInfo.UseShellExecute = true;
        Process.Start(startInfo);
    }
    catch (Exception C)
    {
        MessageBox.Show(C.Message.ToString());
    }
}

private void tiempo_Tick(object sender, EventArgs e)
{
    hr.Text = DateTime.Now.ToLongTimeString();
}

private void btCalendario_Click(object sender, EventArgs e)
{
    Calendario c = new Calendario();
    c.Show();
}

private void btComprobante_Click(object sender, EventArgs e)
{
    Comprobantes HijoCompbt = new Comprobantes();
    HijoCompbt.MdiParent = this;
    HijoCompbt.Show();
}
```



**Figura 12.6.** El formulario de comprobante se encarga de registrar los movimientos de la empresa y generar reportes.

Este formulario pertenece a la clase Comprobantes y en el se lleva a cabo todas las operaciones de movimiento en las cuentas según su naturaleza, con el previo conocimiento sobre el tema. Aunque no es recomendable hacer modificaciones sobre una cuenta registrada, sabemos que se podrían cometer errores humanos y a criterio del administrador se podrán hacer las modificaciones pertinentes y si fuese necesario, eliminar por completo ese registro.

Podremos realizar una búsqueda de registro ingresando el número de referencia para cotejar los datos que tenemos en nuestro sistema y los documentos físicos. También, podemos consultar el saldo de una cuenta determinada y generar los diferentes reportes.

```
private void Comprobantes_Load(object sender, EventArgs e)
{
    this.comprobantesTableAdapter.Fill(this.dataSetComprobantes.Comprobantes);
}
```

```
ListComprobante.Items.Add("Ajuste");
ListComprobante.Items.Add("Diario");
ListComprobante.Items.Add("Egreso");
ListComprobante.Items.Add("Ingreso");

}

public void VerificarNumCta(string cta)
{
    try
    {
        SqlCommand consulta;
        string cadenaConexion;
        SqlConnection conexion;

        cadenaConexion = "Data Source=.\sqlexpress;" + "Initial
Catalog=BBDD_SCCSJ;Integrated Security=True";
        conexion = new SqlConnection(cadenaConexion);
        conexion.Open();

        consulta = new SqlCommand("select NumCta from CatalogoCta", conexion);

        lector = consulta.ExecuteReader();

        while (lector.Read())
        {
            if (cta == lector["NumCta"].ToString())
            {
                existe = "verdadero";
                lector.Close();
                return;
            }
        }

        lector.Close();
    }
    catch (Exception sm)
    {
        MessageBox.Show(sm.Message.ToString());
    }
}

private void btAceptar_Click(object sender, EventArgs e)
```

```
{  
  
    cuenta = NumCta.Text;  
  
    VerificarNumCta(cuenta);  
  
    if (existe == "verdadero")  
    {  
        try  
        {  
            SqlCommand orden;  
            SqlConnection conexion;  
  
            string cadenaConexion = "Data Source=.\sqlexpress;" + "Initial  
Catalog=BBDD_SCCSJ;Integrated Security=True";  
  
            conexion = new SqlConnection(cadenaConexion);  
  
            orden = new SqlCommand("insert into Comprobantes  
values(@NumReferencia,@NumCta,@Dia,@Mes,@Anyo,@Concepto,@Debe,@Haber,@  
Tipo)", conexion);  
  
            orden.Parameters.Add(new SqlParameter("@NumReferencia",  
SqlDbType.VarChar, 15));  
            orden.Parameters["@NumReferencia"].Value = ctRef.Text;  
  
            orden.Parameters.Add(new SqlParameter("@NumCta", SqlDbType.VarChar,  
15));  
            orden.Parameters["@NumCta"].Value = NumCta.Text;  
  
            orden.Parameters.Add(new SqlParameter("@Dia", SqlDbType.Int, 10));  
            orden.Parameters["@Dia"].Value = DateTime.Now.Day.ToString();  
  
            orden.Parameters.Add(new SqlParameter("@Mes", SqlDbType.Int, 10));  
            orden.Parameters["@Mes"].Value = DateTime.Now.Month.ToString();  
  
            orden.Parameters.Add(new SqlParameter("@Anyo", SqlDbType.Int, 10));  
            orden.Parameters["@Anyo"].Value = DateTime.Now.Year.ToString();  
  
            orden.Parameters.Add(new SqlParameter("@Concepto", SqlDbType.VarChar,  
100));  
            orden.Parameters["@Concepto"].Value = ctConcepto.Text;  
  
            if (RBHaber.Checked == true)  
            {
```

```
orden.Parameters.Add(new SqlParameter("@Haber", SqlDbType.Float,
10));
orden.Parameters["@Haber"].Value = ctMonto.Text;

orden.Parameters.Add(new SqlParameter("@Debe", SqlDbType.Float, 10));
orden.Parameters["@Debe"].Value = "0,00";

orden.Parameters.Add(new SqlParameter("@Tipo", SqlDbType.VarChar,
15));
orden.Parameters["@Tipo"].Value = ListComprobante.Text;
}
else
{

orden.Parameters.Add(new SqlParameter("@Haber", SqlDbType.Float,
10));
orden.Parameters["@Haber"].Value = "0,00";

orden.Parameters.Add(new SqlParameter("@Debe", SqlDbType.Float, 10));
orden.Parameters["@Debe"].Value = ctMonto.Text;

orden.Parameters.Add(new SqlParameter("@Tipo", SqlDbType.VarChar,
15));
orden.Parameters["@Tipo"].Value = ListComprobante.Text;
}

MessageBox.Show("Datos almacenados correctamente", "Registro de
comprobantes", MessageBoxButtons.OK, MessageBoxIcon.Information);

orden.Connection.Open();
orden.ExecuteNonQuery();
orden.Connection.Close();

this.Refresh();
this.Update();

this.comprobantesTableAdapter.Fill(this.dataSetComprobantes.Comprobantes);

ctConcepto.Clear();
ctMonto.Clear();
```

```
        ctRef.Clear();
        NumCta.Clear();

        BuscarValores(cuenta);

    }
    catch (Exception c)
    {
        MessageBox.Show(c.Message.ToString());
    }
}
else
    MessageBox.Show("No existe el Número de cuenta ingresada", "Número de
cuenta", MessageBoxButtons.OK, MessageBoxIcon.Error);
}

private void btEliminar_Click(object sender, EventArgs e)
{

    try
    {
        string valor = datosC.CurrentCell.Value.ToString();
        string msj = "El registro " + valor + " ha sido eliminado de la base de datos";

        SqlCommand orden;
        SqlConnection conexion;

        string cadenaConexion = "Data Source=.\sqlexpress;" + "Initial
Catalog=BBDD_SCCSJ;Integrated Security=True";

        conexion = new SqlConnection(cadenaConexion);

        orden = new SqlCommand("delete from Comprobantes where NumReferencia
like " + "'" + valor + "'", conexion);

        MessageBox.Show(msj, "Eliminación de registro(s)", MessageBoxButtons.OK,
MessageBoxIcon.Information);

        orden.Connection.Open();
        orden.ExecuteNonQuery();
        orden.Connection.Close();

    }
    catch (Exception m)
```

```
{
    MessageBox.Show(m.Message.ToString());
}

private void btModificar_Click(object sender, EventArgs e)
{
    try
    {
        this.comprobantesTableAdapter.Update(this.dataSetComprobantes.Comprobantes);
        MessageBox.Show("Su(s) registro(s) ha(n) sido actualido(s)", "Actualización de
los registros de comprobante", MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
    catch (Exception m)
    {
        MessageBox.Show(m.Message.ToString());
    }
}

private void btestadoActualDeCta_Click(object sender, EventArgs e)
{
    try
    {
        string ctaSeleccionada = datosC.CurrentCell.Value.ToString();

        SqlCommand consulta;
        string cadenaConexion;
        SqlConnection conexion;

        cadenaConexion = "Data Source=.\sqlexpress;" + "Initial
Catalog=BBDD_SCCSJ;Integrated Security=True";
        conexion = new SqlConnection(cadenaConexion);
        conexion.Open();

        consulta = new SqlCommand("select sum(Debe)-sum(Haber) as Resultado from
Comprobantes where NumCta like " + "" + ctaSeleccionada + """, conexion);

        lector = consulta.ExecuteReader();

        lector.Read();
    }
}
```

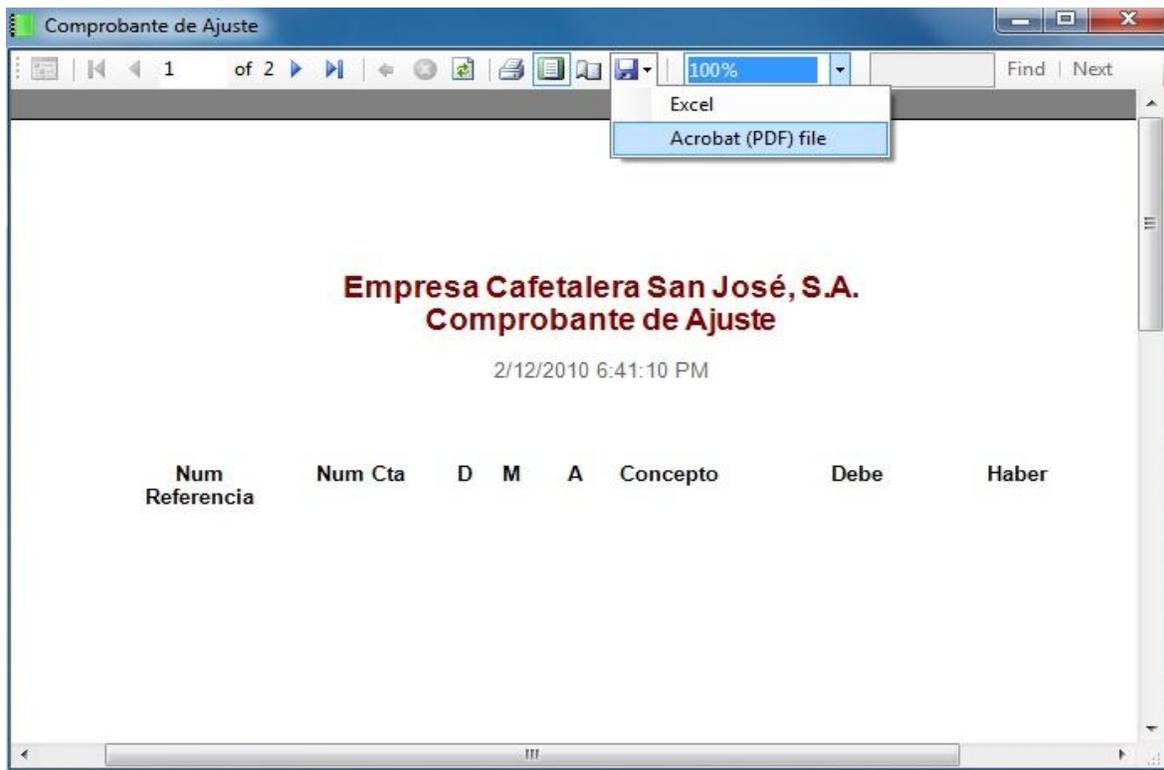
```
        string msj = "El saldo de la cta. " + ctaSeleccionada + " es C$ " +  
lector["Resultado"].ToString();
```

```
        MessageBox.Show(msj,"Info. sobre Cta.",MessageBoxButtons.OK,  
MessageBoxIcon.Asterisk);
```

```
        lector.Close();  
    }  
    catch (Exception c)  
    {  
        MessageBox.Show(c.Message.ToString());  
    }  
}
```

//Generar la busqueda

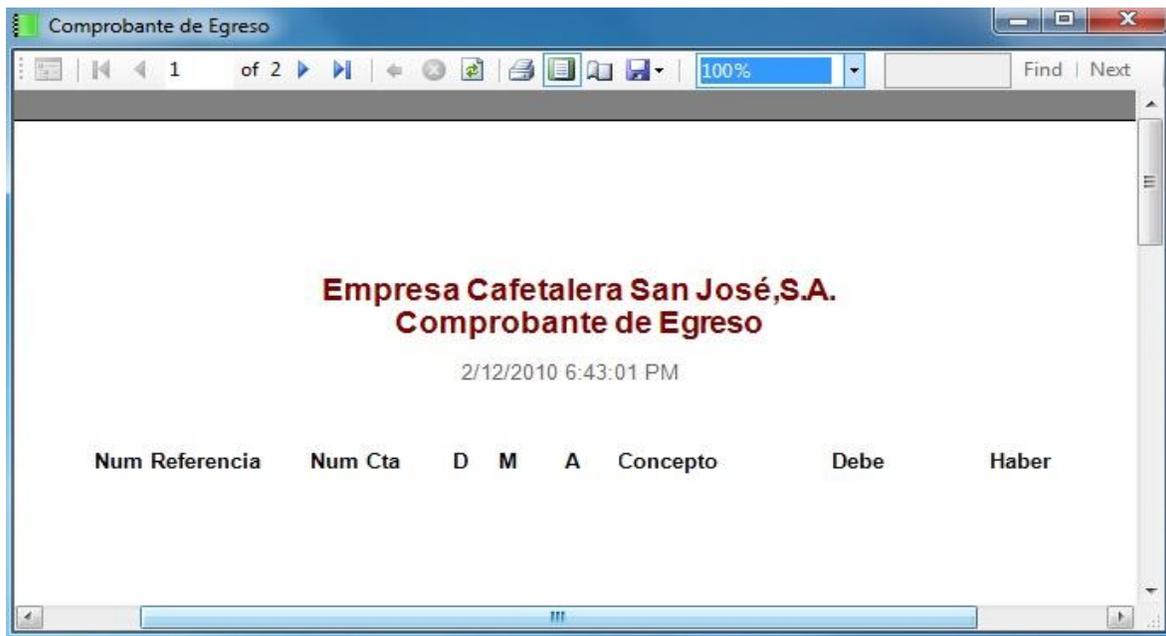
```
private void fillByToolStripButton_Click(object sender, EventArgs e)  
{  
    try  
    {  
        this.comprobantesTableAdapter.FillBy(this.dataSetComprobantes.Comprobantes,  
refToolStripTextBox.Text);  
    }  
    catch (System.Exception ex)  
    {  
        System.Windows.Forms.MessageBox.Show(ex.Message);  
    }  
}
```



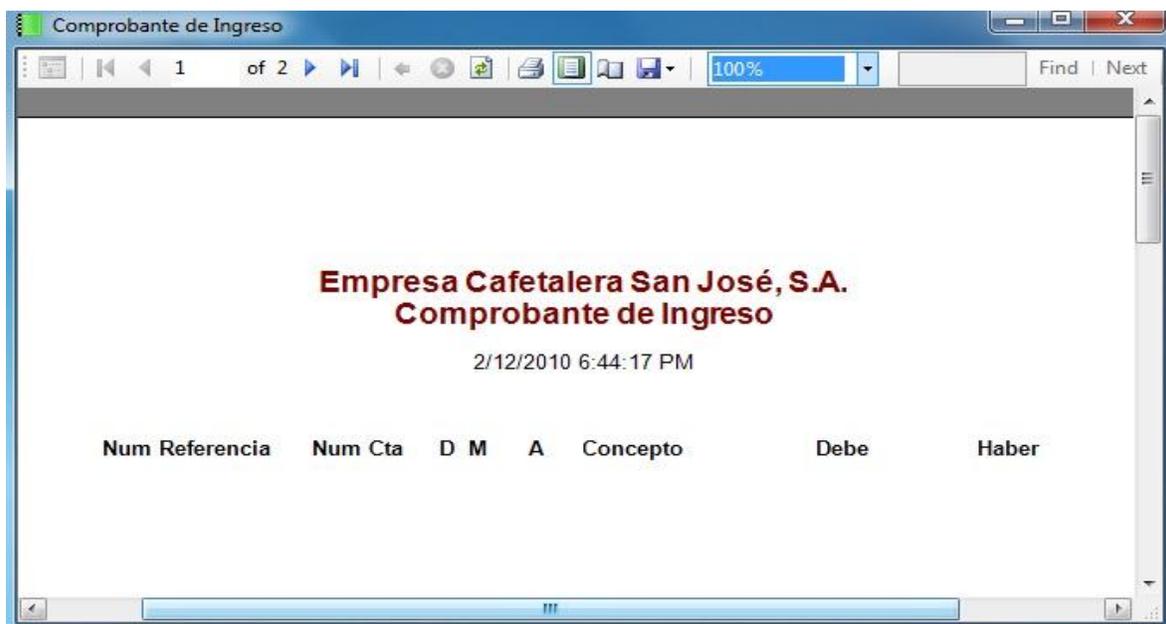
**Figura 12.7.** Formulario para generar el reporte del comprobante de ajuste y perteneciente a la clase ReportesComprobantes

| Num Referencia | Num Cta         | D  | M | A    | Concepto                     | Debe     | Haber   |
|----------------|-----------------|----|---|------|------------------------------|----------|---------|
| A-001          | 100-00-00-C1200 | 16 | 1 | 2010 | Venta de cafe                | 25000    | 0       |
| A-002          | 100-00-00-C1200 | 16 | 1 | 2010 | Venta cafe                   | 15000    | 0       |
| A-003          | 100-00-00-C1200 | 16 | 1 | 2010 | Devolucion cafe              | 0        | 12750.5 |
| A-004          | 150-00-00-40000 | 18 | 1 | 2010 | Compra de vehiculo           | 250000   | 0       |
| B-001          | 200-10-00-00000 | 18 | 1 | 2010 | Deuda                        | 0        | 32000   |
| B12            | 200-00-00-00004 | 10 | 2 | 2010 | Venta de productos agricolas | 153426   | 0       |
| b13            | 120-00-00-00000 | 10 | 2 | 2010 | gastos                       | 227316.6 | 0       |

**Figura 12.8.** Formulario para generar el reporte del comprobante de diario y perteneciente a la clase ComprobanteDiario



**Figura 12.9.** Formulario para generar el reporte del comprobante de Egreso y perteneciente a la clase ComprobanteEgreso



**Figura 12.10.** Formulario para generar el reporte del comprobante de Ingreso y perteneciente a la clase ComprobanteIngreso

Las cuatro figuras anteriores muestran los reportes que se pueden generar desde el formulario de Comprobantes, estos reportes se llevan a cabo usando la herramienta ReportWizard, el cual sirve de enlace entre una plantilla de documento y un determinado registro de la base de datos, para tal efecto se debe de configurar y posteriormente agregaremos esa plantilla a nuestro control ReportViewer de la interfaz (Vea. Marco Teórico).

| NombreCta       | Nivel | NumCta          |
|-----------------|-------|-----------------|
| Banco           | 1     | 000-80-00-00... |
| Capital social  | 3     | 140-00-00-00... |
| Gastos Admon.   | 5     | 230-00-00-00... |
| Gastos Fin.     | 5     | 120-00-00-00... |
| Gastos Prod.    | 5     | 100-00-00-00... |
| Proveedores     | 2     | 120-00-00-00... |
| Venta de tod... | 3     | 000-00-00-00... |
| ventas          | 4     | 200-00-00-00... |
| *               |       |                 |

**Figura 12.11.** Formulario para el registro del catálogo de cuentas.

Este formulario pertenece a la clase CatalogoCta, y se ocupa de registrar la cuentas con las cuales se llevarán a cabo las operaciones de la empresa, previamente el administrador debe crear un catálogo de cuentas y

posteriormente ingresarla al sistema donde le dará un nivel que determina su naturaleza.

- **Nivel1:** Activos
- **Nivel2:** Pasivos
- **Nivel3:** Capital y Patrimonio
- **Nivel4:** Ingresos y Ventas
- **Nivel5:** Gastos

Además, se podrá realizar el reporte del catálogo de cuenta y a criterio del administrador modificará los datos pero no los eliminará, al cerrar el formulario se efectuará tal tarea.

```
private void CatalogoCta_Load(object sender, EventArgs e)
{
    this.catalogoCtaTableAdapter.Fill(this.dataSetCatalogCta.CatalogoCta);

    listNivel.Items.Add("1");
    listNivel.Items.Add("2");
    listNivel.Items.Add("3");
    listNivel.Items.Add("4");
    listNivel.Items.Add("5");
}

private void btGuardar_Click(object sender, EventArgs e)
{
    try
    {
        SqlCommand orden;
        SqlConnection conexion;

        string cadenaConexion = "Data Source=.\sqlexpress;" + "Initial
Catalog=BBDD_SCCSJ;Integrated Security=True";

        conexion = new SqlConnection(cadenaConexion);

        orden = new SqlCommand("insert into CatalogoCta
values(@NombreCta,@NumCta,@Nivel)", conexion);
```

```
orden.Parameters.Add(new SqlParameter("@NombreCta", SqlDbType.VarChar,
80));
orden.Parameters["@NombreCta"].Value = ctNombreCta.Text;

orden.Parameters.Add(new SqlParameter("@NumCta", SqlDbType.VarChar,
15));
orden.Parameters["@NumCta"].Value = ctCodCta.Text;

orden.Parameters.Add(new SqlParameter("@Nivel", SqlDbType.Int, 10));
orden.Parameters["@Nivel"].Value = listNivel.Text;

MessageBox.Show("Datos almacenados correctamente", "Registro de Catálogo
de cuentas", MessageBoxButtons.OK, MessageBoxIcon.Information);

orden.Connection.Open();
orden.ExecuteNonQuery();
orden.Connection.Close();

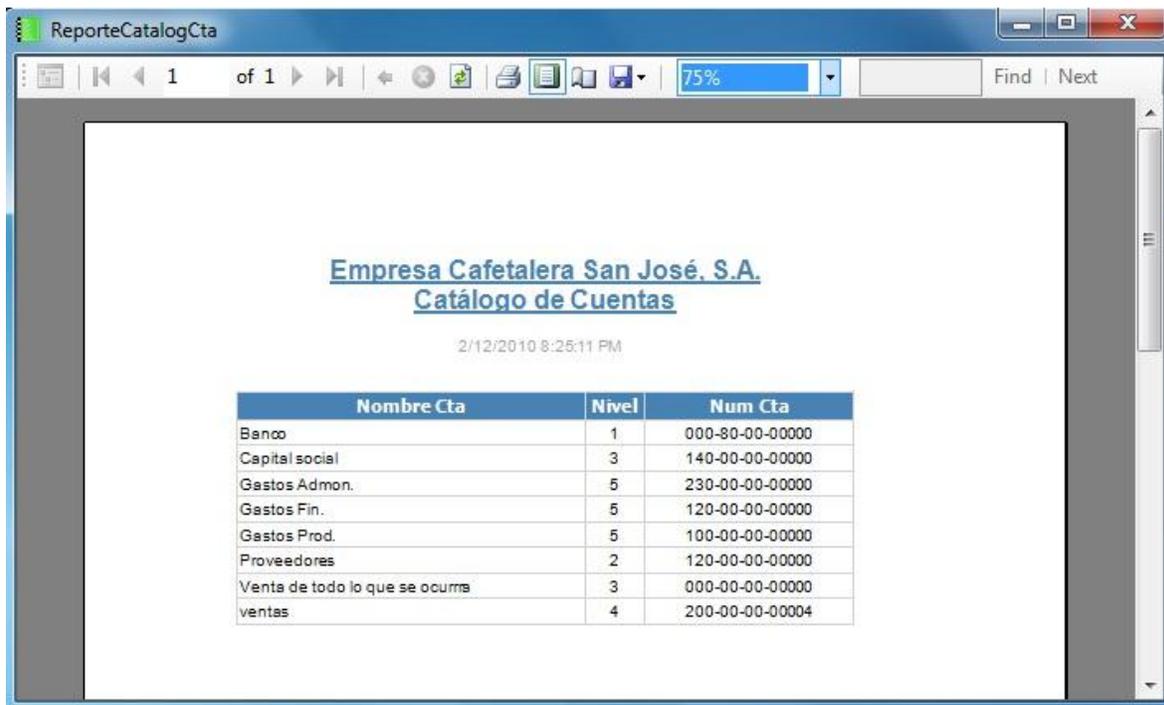
this.Refresh();
this.Update();

this.catalogoCtaTableAdapter.Fill(this.dataSetCatalogCta.CatalogoCta);

ctNombreCta.Clear();
ctCodCta.Clear();

}
catch (Exception c)
{
    MessageBox.Show(c.Message.ToString());
}
}

private void ImprimirCatalogCta_Click(object sender, EventArgs e)
{
    ReporteCatalogCta RCC = new ReporteCatalogCta();
    RCC.Show();
}
```



The screenshot shows a window titled 'ReporteCatalogCta' with a toolbar at the top containing navigation and search icons. The main content area displays the following information:

**Empresa Cafetalera San José, S.A.**  
**Catálogo de Cuentas**  
2/12/2010 8:25:11 PM

| Nombre Cta                     | Nivel | Num Cta         |
|--------------------------------|-------|-----------------|
| Banco                          | 1     | 000-80-00-00000 |
| Capital social                 | 3     | 140-00-00-00000 |
| Gastos Admon.                  | 5     | 230-00-00-00000 |
| Gastos Fin.                    | 5     | 120-00-00-00000 |
| Gastos Prod.                   | 5     | 100-00-00-00000 |
| Proveedores                    | 2     | 120-00-00-00000 |
| Venta de todo lo que se ocurra | 3     | 000-00-00-00000 |
| ventas                         | 4     | 200-00-00-00004 |

**Figura 12.12.** Formulario para el reporte del Catálogo de cuentas.

Este formulario pertenece a la clase ReporteCatalogCta y al igual que los reportes que genera el formulario de comprobantes este también sigue los mismos pasos.



The screenshot shows a window titled 'Reportes Financieros - SCCSJ'. It features two date selection sections: 'Fecha Inicio' and 'Fecha Fin'. Each section has a 'Mes' dropdown menu and an 'Año' text input field. Below these sections are three buttons: 'Balance General', 'Balanza Comparativa', and 'Estado de Pérdida o Ganancia'.

**Figura 12.13.** En este formulario se generarán los reportes de los estados financieros de la empresa.

Este formulario pertenece a la clase EstadoFinanciero, en el se ingresa un rango de fechas y según el tipo de reporte que se desee generar podrá seleccionarlo desde la interfaz, el resultado esta basado en las fechas por medio de consultas y al momento de generar el reporte este se mandará a imprimir directamente a la impresora.

```
private void btBalanzaAnual_Click(object sender, EventArgs e)
{
    if (fiM.Text.Length == 0)
    {
        MessageBox.Show("Completar todos los campos", "SCCSJ 2010",
        MessageBoxButtons.OK, MessageBoxIcon.Error);
        return;
    }

    if (ctfiAnyo.Text.Length == 0)
    {
        MessageBox.Show("Completar todos los campos", "SCCSJ 2010",
        MessageBoxButtons.OK, MessageBoxIcon.Error);
        return;
    }

    if (ffM.Text.Length == 0)
    {
        MessageBox.Show("Completar todos los campos", "SCCSJ 2010",
        MessageBoxButtons.OK, MessageBoxIcon.Error);
        return;
    }

    if (ctffAnyo.Text.Length == 0)
    {
        MessageBox.Show("Completar todos los campos", "SCCSJ 2010",
        MessageBoxButtons.OK, MessageBoxIcon.Error);
        return;
    }

    BalanzaComparativaAnual.Print();
    ctfiAnyo.Clear();
    ctffAnyo.Clear();
}
```



```

a1 = lector["NumCta"].ToString();
a2 = lector["NombreCta"].ToString();
a3 = lector["SD"].ToString();
a4 = lector["SH"].ToString();

e.Graphics.DrawString(a1, new Font("Arial", 12, FontStyle.Bold), b, 0, dx);
e.Graphics.DrawString("\t\t" + a2, new Font("Arial", 12, FontStyle.Bold), b,
0, dx);
e.Graphics.DrawString("\t\t\t\t\t" + a3, new Font("Arial", 12,
FontStyle.Bold), b, 0, dx);
e.Graphics.DrawString("\t\t\t\t\t\t\t\t\t" + a4, new Font("Arial", 12,
FontStyle.Bold), b, 0, dx);

cont2 = cont2 + 1;
dx = dx + (20 * cont2);
}

lector.Close();

cadCX = "Data Source=.\sqlexpress;" + "Initial
Catalog=BBDD_SCCSJ;Integrated Security=True";
CnX = new SqlConnection(cadCX);
CnX.Open();

consulta2 = new SqlCommand("select A.NumCta,A.NombreCta,sum(Debe) as
SD,sum(Haber) as SH from CatalogoCta as A,Comprobantes as B where (A.NumCta like
B.NumCta and Nivel=2 and Mes between " + fiM.Text + " and " + ffM.Text + " and Anyo
between " + ctfiAnyo.Text + " and " + ctffAnyo.Text + ") group by
A.NumCta,NombreCta", conexion);

lector2 = consulta2.ExecuteReader();

while (lector2.Read())
{
b1 = lector2["NumCta"].ToString();
b2 = lector2["NombreCta"].ToString();
b3 = lector2["SD"].ToString();
b4 = lector2["SH"].ToString();

e.Graphics.DrawString(b1, new Font("Arial", 12, FontStyle.Bold), b, 0, dx);
e.Graphics.DrawString("\t\t"+b2, new Font("Arial", 12, FontStyle.Bold), b, 0,
dx);
e.Graphics.DrawString("\t\t\t\t\t" + b3, new Font("Arial", 12,
FontStyle.Bold), b, 0,dx);

```

```

        e.Graphics.DrawString("\t\t\t\t\t\t\t" + b4, new Font("Arial", 12,
FontStyle.Bold), b, 0, dx);

        cont2 = cont2 + 1;
        dx = dx + (20 * cont2);
    }

```

//Realizar el código para dibujar los saldos

```

        lector2.Close();
    }
    catch (Exception s)
    {
        MessageBox.Show(s.Message.ToString());
    }
}

public void SaldoDebe()
{
    try
    {
        SqlCommand consulta;
        string cadenaConexion;
        SqlConnection conexion;

        cadenaConexion = "Data Source=.\sqlexpress;" + "Initial
Catalog=BBDD_SCCSJ;Integrated Security=True";
        conexion = new SqlConnection(cadenaConexion);
        conexion.Open();

        consulta = new SqlCommand("select sum(Debe) as TotalDebe from CatalogoCta
as A,Comprobantes as B where A.NumCta=B.NumCta and Mes between " + fiM.Text + "
and " + ffM.Text + " and Anyo between " + ctfiAnyo.Text + " and " + ctffAnyo.Text,
conexion);

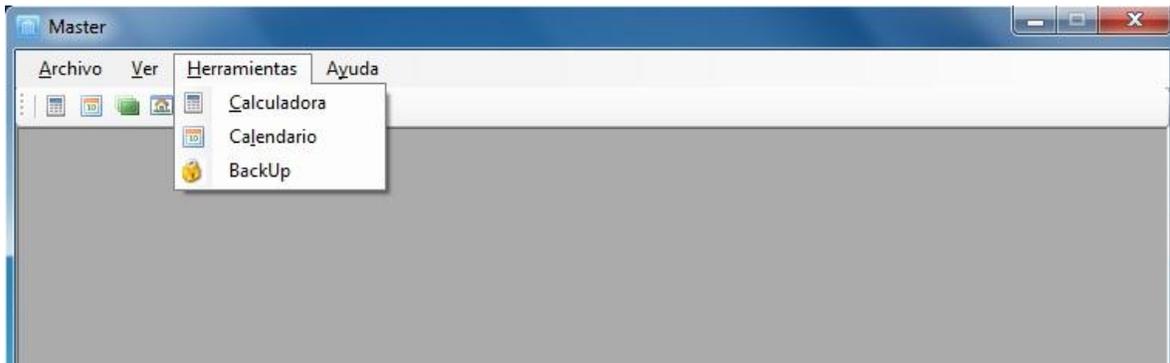
        lector = consulta.ExecuteReader();

        lector.Read();

        Tdebe = lector["TotalDebe"].ToString();
        lector.Close();
    }
}

```

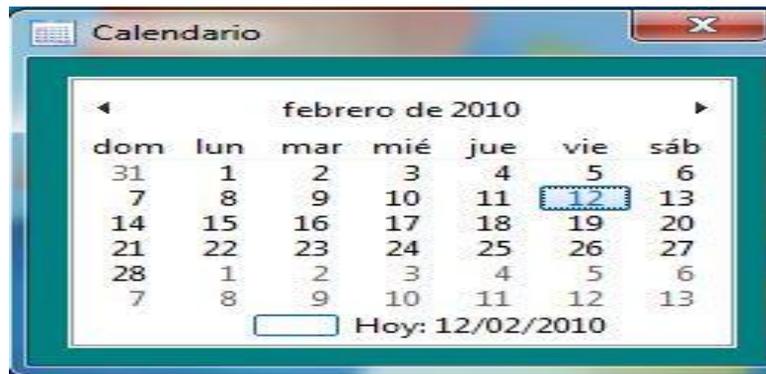
```
catch (Exception c)
{
    MessageBox.Show(c.Message.ToString());
}
}
```



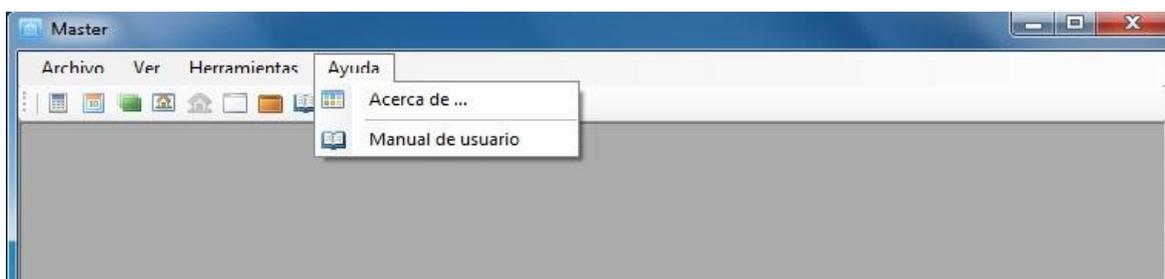
**Figura 12.14.** Vista del menú herramientas del formulario Master



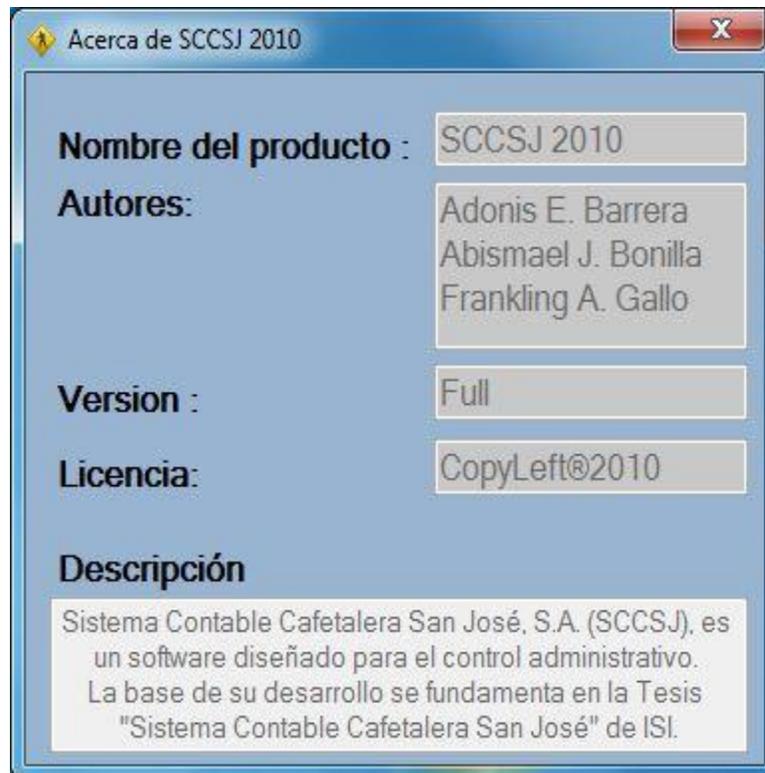
**Figura 12.15.** Calculadora del sistema



**Figura 12.16.** Formulario Calendario perteneciente a la clase del mismo nombre y que le permitirá al usuario administrador navegar por diferentes fechas.



**Figura 12.17.** Vista del menú ayuda del formulario master.



**Figura 12.18.** Información sobre el SCCSJ



**Figura 12.19.** Formulario para crear y restaurar copias de seguridad

Las copias de seguridad nos ayudan a tener un respaldo de la información, una vez que nuestra base de datos sufra cualquier daño por algún motivo, podremos recuperar los datos que ya antes fueron guardados.

Al igual como si fuésemos a realizar una consulta, ejecutaremos los siguientes comandos:

#### Para crear un BackUp

```
BACKUP DATABASE “Nombre de la base de datos”  
TO DISK = 'C:\Backup\“Nombre de la base de datos”.bak’  
WITH COPY_ONLY;
```

#### Restaurar una Base de datos

```
RESTORE DATABASE “Nombre de la base de datos”  
FROM DISK = 'C:\Backup\“Nombre de la base de datos”.bak';
```

**Nota:** La ruta puede ser cualquiera, no importa si es un pen drive o cualquier otro dispositivo de almacenamiento.

# **AneXos**

| <b>Empresa Cafetalera San José, S.A.</b> |                 |                          |
|--|-----------------|--------------------------|
| <b>Balance General</b>                   |                 |                          |
| <b>1/1/2010 - 30/2/2010</b>              |                 |                          |
| <b>Activos</b>                           |                 |                          |
| Banco                                    | <u>-845.98</u>  |                          |
| Total Activos                            | .....           | <u>-845.98</u><br>=====  |
| <b>Pasivos</b>                           |                 |                          |
| Proveedores                              | <u>227193.6</u> |                          |
| Total Pasivos                            | .....           | <u>227193.6</u><br>===== |
| <b>Capital y Patrimonio</b>              |                 |                          |
| Capital social                           | -23444          |                          |
| Utilidad o Deficit del periodo           | 2266643.44      |                          |
| Total Capital y Patrimonio               |                 | <u>2243199.44</u>        |
| Total Pasivo más capital                 |                 | <u>2470393.04</u>        |

**Empresa Cafetalera San José, S.A.**  
**Estado de Pérdida o Ganancias**  
**1/1/2010 - 30/2/2010**

**Ingresos y Ventas**

|                                |            |                   |
|--------------------------------|------------|-------------------|
| ventas                         | 153426     |                   |
| Total Utilidad Bruta S/Ventas  |            | <u>153426</u>     |
| Gastos Admon.                  | 267707.48  |                   |
| Gastos Fin.                    | 227193.6   |                   |
| Gastos Prod.                   | 1618316.36 |                   |
| Total Utilidad Bruta S/Ventas  |            | <u>2113217.44</u> |
| Utilidad o Deficit del Período |            | <u>2266643.44</u> |

**Empresa Cafetalera San José, S.A.**  
**Balanza Comparativa**  
**1/1/2010 - 30/2/2010**

|                 |                | <b>Debe</b>       | <b>Haber</b>      |
|-----------------|----------------|-------------------|-------------------|
| 000-80-00-00000 | Banco          | 0                 | 2470393.04        |
| 120-00-00-00000 | Proveedores    | <u>227316.6</u>   | <u>123</u>        |
|                 | <b>Totales</b> | <b>2494083.04</b> | <b>2494083.04</b> |

## XVIII. BIBIOGRAFIA

S. Pressman, Roger. Ingeniería del Software, Un enfoque moderno. Quinta edición Editorial: McGrawHill.

E. Kendall, Kenneth & E. Kendall, Julie. (2005) .Análisis y diseño de sistema, sexta edición. México: Pearson Educación.

Alarcón Raúl , (2000). Diseño orientado a objetos con UML. Grupo EIDOS Consultoría y Documentación Informática, S.L., 2000

Ferguson Jeff , Patterson Brian , Beres Jason, Boutquin Pierre &Gupta Meeta. La Biblia de C#, Anaya Multimedia.

Méndez A Carlos E. Metodología, Diseño y desarrollo del proceso de investigación. Tercera edición, Editorial: McGrawHill.

Romero Javier. Principios de contabilidad. Guajardo Cantún Gerardo. Contabilidad Financiera. Lara Flores Elías. Primer curso de contabilidad.

Microsoft Visual Studio 2008. MSDN Library 2008.

### **Sitios web:**

<http://msdn.microsoft.com/es-es/library>

<http://www.asp.net/learn/>

<http://www.programar.net/directory/details.aspx?id=2677>

<http://es.wikipedia.org/wiki/>



## **Sugerencias y Recomendaciones**

De antemano sabemos que siempre tendremos que mejorar nuestros sistemas, ya que no hay sistema perfecto, lo único que podemos hacer es poner mucho empeño en la elaboración de cualquier software y asegurarse que cumpla con los requerimientos especificados.

Para proyecto futuro, este sistema deberá contar con una nueva entidad exclusiva para el almacenamiento de la información de la empresa, de tal forma si esta llegase a cambiar de razón social pueda realizar sus reportes con el nuevo membrete. Además, se deberá expandir el sistema de forma que pueda manejarse las entradas y salidas a bodega.

## **XVI. CONCLUSION**

- ❖ Para el buen desarrollo de un sistema contable, se deben de tomar en cuenta los aspectos técnicos teóricos sobre la contabilidad, de tal forma que tengamos un dominio básico del funcionamiento de las operaciones que se llevaran a cabo en el sistema y también para poder codificar las instrucciones correctas para una eficaz utilidad.
- ❖ Además para poder crear un sistema capaz de cumplir con la necesidades de una empresa debemos recabar la información necesaria, que sea lo más clara posible y detallada, para que nuestro análisis se adecue de forma exacta a los requerimientos del sistema.
- ❖ No basta con crear un sistema, debemos documentar su diseño. Al igual que un plano de una casa, esto nos servirá de guía para nuevas modificaciones, verificación de errores y sobre todo será un respaldo, en el cual sería posible agregar nuevas funcionalidades que se integre sin problemas ni contratiempos a nuestro sistema.