



**BENEMERITA UNIVERSIDAD AUTONOMA DE
PUEBLA**

FACULTAD DE CIENCIAS DE LA COMPUTACION

**Búsqueda de Tendencias Económicas en un
Sistema de Base de Datos Financiero**

TESIS PROFESIONAL

Que para obtener el título de

LICENCIADA EN CIENCIAS DE LA COMPUTACION

Presenta:

Mónica Flores Martínez

Asesor:

M.C. Yolanda Moyao Martínez

Puebla, Pue.

Noviembre 2007



CONTENIDO

1. Marco Teórico	4
1.1 Ingeniería del Software.....	5
1.1.1 Definición del termino ingeniería de software.....	5
1.2 Software.....	6
1.2.1 Características del Software.....	6
1.2.2 Aplicaciones del Software.....	7
1.2.3 Visión general de ingeniería de software.....	7
1.3 Modelos de proceso de software.....	8
1.3.1 El modelo lineal secuencial.....	8
1.3.2 El modelo DRA.....	9
1.3.3 El modelo en espiral.....	10
1.3.4 El modelo de prototipos.....	11
1.3.5 El modelo de cascada.....	12
1.4 Base de Datos.....	13
1.4.1 Abstracción de datos.....	14
1.4.2 Usuarios de la base de datos.....	16
1.4.3 Tipos de bases de datos.....	16
1.5 Modelo de datos.....	17
1.5.1 Modelo lógico basado en objetos.....	17
1.5.1.1 Modelo entidad-relación (E-R).....	17
1.5.1.2 Tipos de relaciones.....	18
1.5.1.3 Llaves primarias.....	19
1.5.2 Modelos lógicos basados en registros.....	20
1.5.3 Modelo relacional.....	20
1.6 Normalización.....	21
1.6.1 Formas de normalización.....	22
1.7 Definiciones de términos financieros.....	22
1.8 Arquitectura cliente-servidor.....	24
1.9 Código abierto.....	26
1.9.1 Apache.....	26
1.9.2 PHP.....	26
1.9.3 MySQL.....	26
2. Análisis de Requerimientos	27
2.1 Planteamiento del problema.....	28
2.2 Requerimientos del sistema.....	29
2.2.1 Requerimientos funcionales.....	29
2.2.2 Requerimientos no funcionales.....	30
2.3 Alcance.....	30
2.4 Metodología.....	30
2.5 Recursos del sistema.....	30
2.6 Requerimientos de la interfaz.....	31
2.7 Diseño del sistema.....	32
2.7.1 Diagrama del flujo de datos (D.F.D).....	32



3. Diseño de la base de datos.....	38
3.1 Diseño conceptual de la base de datos.....	38
3.1.1 Descripción de entidades y relaciones.....	38
3.1.2 Diccionario de datos.....	41
3.1.3 Modelo entidad-relación.....	46
3.1.4 Diseño lógico.....	49
4. Implementación y pruebas.....	50
4.1 Implementación.....	51
4.2 Herramientas del sistema.....	51
4.2.1 Conectividad de la base de datos.....	51
4.2.2 Altas en la base de datos.....	52
4.2.3 Bajas en la base de datos.....	54
4.2.4 Actualización en la base de datos.....	55
4.2.5 Consultas en la base de datos.....	57
4.2.6 Generación de gráficos.....	59
4.3 Módulos del sistema.....	60
4.4 Interfaces.....	61
4.5 Pruebas.....	68
Conclusión.....	79
Trabajos a futuro.....	79
Bibliografía.....	80



INTRODUCCIÓN

Los sistemas de información apoyados en Bases de Datos son necesarios para manejar grandes volúmenes de información, que demandan también metodologías y sistemas de control eficientes y aplicables.

El volumen de información generado ha ido creciendo de forma acelerada, convirtiéndose en un problema la distribución de la misma. Gracias a Internet, ahora se cuenta con herramientas que permiten la distribución de la información en forma digital, pero son poco eficientes para la búsqueda económica principalmente.

Por falta de transparencia de información que no se tenía en nuestro país era más difícil tener acceso a la información financiera, en los últimos años se abrieron estas fuentes pero aún no existe una forma didáctica de manejar ese tipo de información que sea rápida, confiable y funcional.

Siempre ha sido necesario estar informados sobre la economía que día a día va rigiendo a nuestro país, principalmente los indicadores económicos, que posteriormente permiten conocer las tendencias económicas del mercado, es la más buscada por los economistas; ya que por medio de estas se encargan de crear una buena estrategia para los fines que así convienen.

Algunas fuentes, aún cuando cuentan con grandes cantidades de información sobre los indicadores económicos, no contemplan en ellas la actualización de la información mensual, las predicciones y las tendencias de nuestra economía. Por lo anterior es necesario un Sistema que satisfaga las necesidades de los economistas.

Las necesidades aquí planteadas pueden ser atendidas con la implementación de un Sistema eficiente, flexible para el control, acceso y administración de la información financiera, que contemple los principales indicadores económicos, además de la visualización de las tendencias y las predicciones en forma gráfica, utilizando una herramienta importante como es Internet, que permite a todo tipo de persona y desde cualquier parte tener acceso a la información del sistema, a economistas, y estudiantes de economía, o cualquier otra persona interesado en el tema.



CAPÍTULO I

MARCO TEÓRICO



1.1 INGENIERÍA DEL SOFTWARE

La **Ingeniería de Software** es una disciplina o área de la informática o Ciencias de la Computación que ofrece métodos o técnicas para desarrollar y mantener software de calidad que resuelven problemas de todo tipo. Hoy día es cada vez más frecuente la consideración de la Ingeniería de Software como una nueva área de la ingeniería, y el ingeniero de software comienza a ser una profesión implantada en el mundo laboral internacional, con derechos, deberes y responsabilidad desde de cumplir con una, ya reconocida consideración social en el mundo empresarial y por suerte, para estas personas con un brillante futuro.

La Ingeniería de software trata con áreas muy diversas de la informática y de las Ciencias de la Computación tales como, construcción de compiladores, sistemas operativos o desarrollos en Intranet o Internet, abordando todas las fases del ciclo de vida del desarrollo de cualquier tipo de sistemas de información aplicables a una infinidad de áreas, tales como: negocios, investigación científica, medicina, producción, logística, banca, control de tráfico, meteorología, el mundo del derecho, la red de redes Internet, redes Intranet y Extranet, etc.[1]

1.1.1 DEFINICIÓN DEL TÉRMINO INGENIERIA DE SOFTWARE

El término Ingeniería se define en el Diccionario de la Real Academia Española de la Lengua como un conjunto de conocimientos y técnicas que permiten aplicar el saber científico a la utilización de la materia y de las fuentes de energía y el término ingeniero se define como persona que profesa o ejerce la ingeniería. De igual manera la Real Academia de Ciencias Exactas, Físicas, Naturales de España define el término Ingeniería como conjunto de conocimientos y técnicas cuya aplicación permite la utilización racional de los materiales y de los recursos naturales, mediante invenciones, construcciones y mediante otras realizaciones provechosas para el hombre.

Evidentemente la Ingeniería de Software es una nueva Ingeniería, parece lógico que reúna las propiedades citadas en las definiciones anteriores. Sin embargo, ni el Diccionario de la Real Academia Española de la Lengua ni la Real Academia Española de Ciencias han incluido todavía el término en sus últimas ediciones; en consecuencia vamos a recurrir para su definición más precisa a algunos de los autores más acreditados que comenzaron en su momento a utilizar el término o bien en las definiciones dadas por organismos internacionales profesionales de prestigio tales como IEEE o ACM. Así, se ha seleccionado las siguientes definiciones de Ingeniería de Software.

Definición. Ingeniería de Software es la aplicación práctica del conocimiento científico en el diseño y construcción de programas de computadoras y la documentación asociada requerida para desarrollar, operar (funcionar) y mantenerlos. Se conoce también como desarrollo de software o producción de software.[2]



1.2 SOFTWARE

El software de computadora es el producto que diseñan y construyen los ingenieros de software. Esto abarca programas que se ejecutan dentro de una computadora de cualquier tamaño y arquitectura, documentos que comprenden formularios virtuales e impresos y datos que combinan números, texto y también incluyen representaciones de información de audio, video, imágenes.

Construir software de computadora es como si construyéramos cualquier otro producto satisfactorio, aplicando un proceso que conduce a un resultado de alta calidad que satisface las necesidades de la gente que usará el producto.

El software es un elemento del sistema que es lógico en lugar de físico. Por tanto el software tiene unas características considerablemente distintas a las del hardware.[3]

1.2.1 CARACTERÍSTICAS DEL SOFTWARE

- *El software se desarrolla, no se fabrica en un sentido clásico.* Aunque existen similitudes entre el desarrollo del software y la construcción de hardware, ambas actividades son fundamentalmente diferentes. En ambas actividades la buena calidad se adquiere mediante un buen diseño y dependen de las personas, pero la relación entre las personas dedicadas y el trabajo realizado es completamente diferente para el software. Las dos actividades requieren de la construcción de un producto pero los enfoques son diferentes. Los costes del software se encuentran en la Ingeniería. Esto significa que los proyectos de software no se pueden gestionar como si fueran proyectos de fabricación.
- *El software no se estropea.* El software no es susceptible a los males del entorno que hacen que el hardware se estropee. Cuando un componente de hardware se estropea, se sustituye por una pieza de repuesto. No hay piezas de repuesto para el software, cada fallo en el software indica un error en el diseño máquina ejecutable, por tanto, el mecanismo del software tiene una complejidad considerablemente mayor que la del mantenimiento del hardware.
- *Aunque la industria tiende a ensamblar componentes, la mayoría del software se construye a medida.* A medida que la disciplina del software evoluciona, se crea un grupo de componentes de diseño estándar. El componente de software debería diseñarse e implementarse para que pueda volver a ser utilizado en muchos programas diferentes. Se ha extendido la visión de reutilización para abarcar no solo los algoritmos, sino también estructuras de datos. Los componentes reutilizables modernos encapsulan tanto datos como procesos que se aplican a los datos, permitiendo al ingeniero del software crear nuevas aplicaciones a partir de las partes reutilizables.



1.2.2 APLICACIONES DEL SOFTWARE

El software puede aplicarse en cualquier situación en que se haya definido previamente un conjunto específico de pasos procedimentales. 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 información de entrada y salida.

- *Software de sistemas.* El software de sistemas es un conjunto de programas que han sido desarrollados para servir a otros programas.
- *Software de tiempo real.* El software que mide/analiza/controla sucesos del mundo real conforme ocurren, se denomina de tiempo real.
- *Software de gestión.* El procesamiento de información comercial constituye la mayor de las áreas de aplicación del software.
- *Software de ingeniería y científico.* El software de ingeniería y científico está caracterizado por los algoritmos de manejo de números.
- *Software empotrado.* Reside en memoria de sólo lectura y se utiliza para controlar productos y sistemas de los mercados industriales y de consumo.
- *Software de computadoras personales.* El procesamiento de textos, las hojas de cálculo, los gráficos por computadora, multimedia, gestión de base de datos, aplicaciones financieras, redes de acceso a base de datos externa.
- *Software de Inteligencia Artificial.* Hace uso de algoritmos no numéricos para resolver problemas complejos para los que no son adecuados el cálculo o el análisis directo.

1.2.3 VISION GENERAL DE INGENIERÍA DE SOFTWARE

El trabajo que se asocia a la ingeniería de software se puede dividir en tres fases genéricas, con independencia del área de aplicación, tamaño, complejidad del proyecto. Cada fase se encuentra con una o varias cuestiones de las destacadas anteriormente.

La *fase de definición* se centra en el qué. Es decir, durante la definición, el que desarrolla el software intenta identificar que información ha de ser procesada, que función y rendimiento se desea, que comportamiento del sistema, que interfaces van a ser establecidas, que restricciones de diseño existen y que criterios de validación se necesitan para definir un sistema correcto.

La *fase de desarrollo* se centra en el cómo. Es decir, durante el desarrollo de un ingeniero del software intenta definir como han de diseñarse las estructuras de datos, como han de implementarse la función dentro de una arquitectura de software, como han de implementarse los detalles procedimentales, como han de caracterizarse interfaces, como ha de traducirse el diseño de un lenguaje de programación y como ha de realizarse la prueba.

La *fase de mantenimiento* se centra en el cambio que se va asociado a la corrección de errores, a las adaptaciones requeridas a medida que evoluciona el entorno del software y a cambios debidas a las mejoras producidas por los requisitos cambiantes del cliente. Durante la fase de mantenimiento se encuentran cuatro tipos de cambios: corrección, adaptación, mejora, prevención.



1.3 MODELOS DE PROCESO DE SOFTWARE

Para resolver los problemas reales de una industria, un ingeniero del software o equipo de ingenieros deben incorporar una estrategia de desarrollo que acompañe al proceso, métodos, capas de herramientas y las fases genéricas. Esta estrategia a menudo se llama modelo de proceso o paradigma de ingeniería del software. Se selecciona un modelo de proceso para la ingeniería del software según la naturaleza del proyecto y de la aplicación, los métodos y las herramientas a utilizarse, los controles y entregas que se requieren.

Todo el desarrollo del software se puede caracterizar como un bucle de resolución de problemas. En el que se encuentran cuatro etapas distintas: status quo, definición de problemas, desarrollo técnico e integración de soluciones. Status quo, representa el estado actual del suceso, la definición de problemas, identifica el problema específico a resolver, el desarrollo técnico, resuelve el problema a través de la aplicación de alguna tecnología, integración de soluciones, ofrece los resultados. La figura 1.1 muestra las fases de un bucle de resolución de problemas.

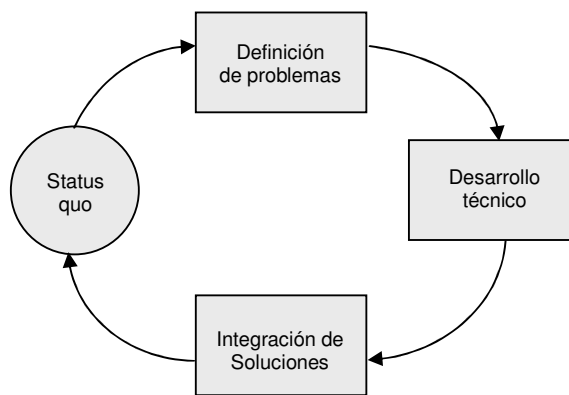


Figura 1.1 Las fases de un bucle de resolución de problemas

1.3.1 EL MODELO LINEAL SECUENCIAL

En la figura 1.2 se ilustra el modelo lineal secuencial para la ingeniería de software. El cliente utiliza el producto central (o sufre la revisión detallada). Como un resultado de utilización y/o evaluación, se desarrolla un plan para el incremento siguiente. El plan afronta la modificación del producto central a fin de cumplir mejor las necesidades del cliente y la entrega de cada incremento, hasta que se elabore el producto completo.

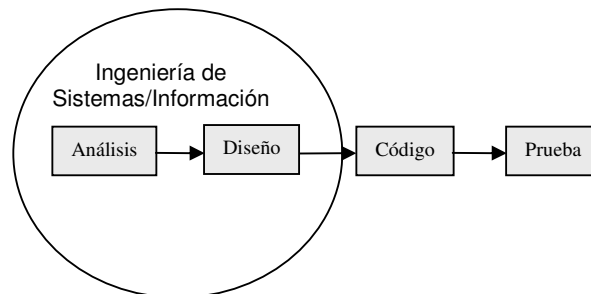


Figura 1.2 Modelo Lineal secuencial



Ingeniería y modelado de Sistemas/Información. Como el software siempre forma parte de un sistema mas grande, el trabajo comienza estableciendo requisitos de todos lo elementos del sistema y asignando al software algún subgrupo de estos requisitos.

Análisis de Requisitos. El proceso de reunión de requisitos se intensifica y se centra especialmente en el software.

Diseño. Este es realmente un proceso de muchos pasos que se centra en estructura de datos, arquitectura de software, representación de interfaz y detalle procedimental.

Generación de código. El diseño se debe traducir en una forma legible y por la máquina.

Pruebas. Una vez que se ha generado el código empiezan las pruebas del programa.

1.3.2 EL MODELO DRA

El Desarrollo Rápido de Aplicaciones (DRA) es un modelo de proceso del desarrollo del software lineal secuencial que enfatiza un ciclo de desarrollo extremadamente corto. El modelo DRA es una adaptación a alta velocidad del modelo lineal secuencial en que se logra el desarrollo rápido utilizando una construcción basada en componentes. Si se comprenden bien los requisitos y se limita el ámbito del proyecto, el proceso DRA permite al equipo de desarrollo crear un sistema completamente funcional dentro de los períodos cortos de tiempo. En la figura 1.3 se ilustra.[3]

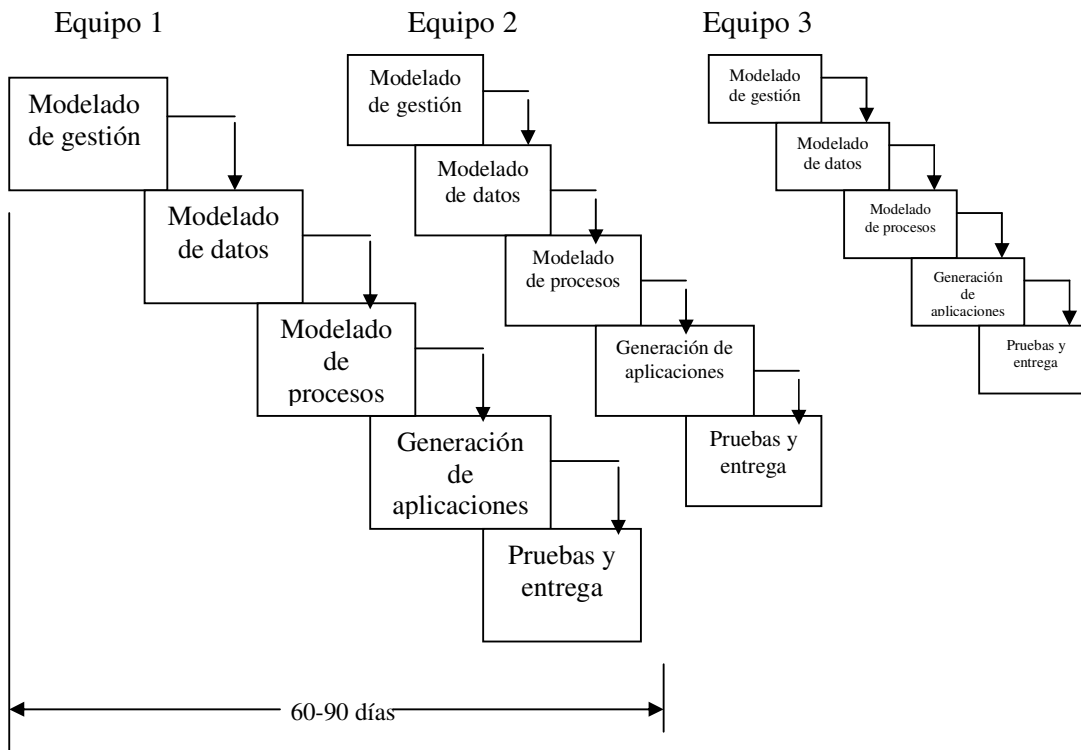


Figura 1.3 Modelo DRA



Modelado de gestión. El flujo de información entre las funciones de gestión se modela de forma que responda a las siguientes preguntas: ¿Qué información conduce al proceso de gestión? ¿Qué información se genera? ¿Quién la genera? ¿A dónde va la información? ¿Quién la procesa?.

Modelado de datos. El flujo de información definido como parte de la fase de modelado de gestión se define como un conjunto de objetos de datos necesarios para apoyar a la empresa. Se definen las características de cada uno de los objetos y las relaciones entre esos objetos.

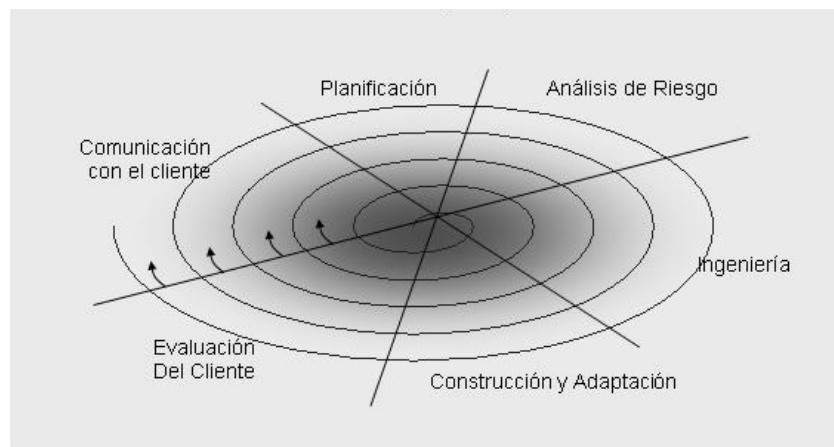
Modelado de proceso. Los objetos de datos detenidos en la fase de modelado de datos quedan transformados para lograr el flujo de información necesario para implementar una función de gestión. Las descripciones del proceso se crean para añadir, modificar, suprimir o recuperar un objeto de datos.

Generación de aplicaciones. El DRA asume la utilización de técnicas de cuarta generación. En lugar de crear software con lenguajes de programación de tercera generación, el proceso DRA trabaja para volver a utilizar componentes de programas ya existentes o crear componentes reutilizables.

1.3.3 EL MODELO EN ESPIRAL

El modelo espiral propuesto originalmente por Bohem, es un modelo de proceso de software evolutivo que acompaña la naturaleza interactiva de construcción de prototipos con los aspectos controlados y sistemáticos del modelo lineal secuencial. Se proporciona el potencial para el desarrollo rápido de versiones incrementales del software. En el modelo espiral, el software se desarrolla en una serie de versiones incrementales. Durante las primeras iteraciones, la versión incremental podría ser un modelo en papel o prototipo. Durante las últimas iteraciones, se produce versiones cada vez más complejas de ingeniería del sistema.

El modelo espiral se divide en un número de actividades estructurales, también llamada regiones de tareas. Generalmente, existen entre tres y seis regiones de tareas. La figura 1.4 representa un modelo en espiral que contiene seis regiones de tarea.



Modelo 1.4. Modelo en espiral



1.3.5 MODELO EN CASCADA

El más conocido, esta basado en el ciclo convencional de una ingeniería, el paradigma del ciclo de vida abarca las actividades que muestra la figura 1.6

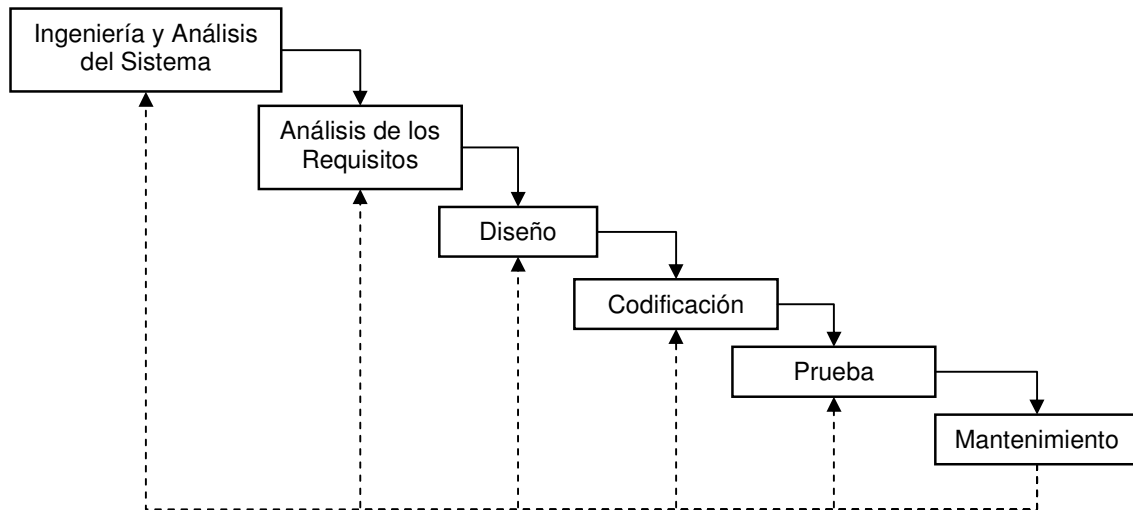


Figura 1.6 Modelo en cascada

Ingeniería y Análisis del Sistema: Debido a que el software es siempre parte de un sistema mayor el trabajo comienza estableciendo los requisitos de todos los elementos del sistema y luego asignando algún subconjunto de estos requisitos al software.

Análisis de los requisitos del software: El proceso de recopilación de los requisitos se centra e intensifica especialmente en el software. El ingeniero de software (analistas) debe comprender el ámbito de la información del software, así como la función, el rendimiento y las interfaces requeridas.

Diseño: El diseño del software se enfoca en cuatro atributos distintos del programa: la estructura de los datos, la arquitectura del software, el detalle procedimental y la caracterización de la interfaz. El proceso de diseño traduce los requisitos en una representación del software con la calidad requerida antes de que comience la codificación.

Codificación: El diseño debe traducirse en una forma legible para la máquina. El paso de codificación realiza esta tarea. Si el diseño se realiza de una manera detallada la codificación puede realizarse mecánicamente.

Prueba: Una vez que se ha generado el código comienza la prueba del programa. La prueba se centra en la lógica interna del software, y en las funciones externas, realizando pruebas que aseguren que la entrada definida produce los resultados que realmente se requieren.

Mantenimiento: El software sufrirá cambios después de que se entrega al cliente. Los cambios ocurrirán debido a que hayan encontrado errores, a que el software deba adaptarse a cambios del entorno externo (sistema operativo o dispositivos periféricos), o debido a que el cliente requiera ampliaciones funcionales o del rendimiento.



1.4 BASE DE DATOS

Una **base** o **banco de datos** es un conjunto de datos que pertenecen al mismo contexto 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 gracias al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos tienen formato electrónico, que ofrece un amplio rango de soluciones al problema de almacenar datos.

La creación de una base de datos debe ser realizada cuidadosamente procurando cumplir con una serie de objetivos (entre otros) que se detallan a continuación:

- Permitir un fácil acceso a la información. El sistema debe ser facilitador en alto rendimiento, la velocidad es un factor esencial así como la consistencia de los datos.
- Evitar la redundancia. La información almacenada ocupará irremediablemente un espacio en memoria por lo cual es de vital importancia eliminar la posibilidad de almacenar datos repetidos ya que adicionalmente podrían llevarnos a inconsistencias en la información.

Un sistema de gestión de bases de datos (SGBD) consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a dichos datos. La colección de datos, normalmente denominada base de datos, contiene información acerca de una empresa particular. El primer objetivo de un SGBD es proporcionar un entorno que sea tanto práctico como eficiente de usar en la recuperación y el almacenamiento de información de la base de datos. Las propiedades de los sistemas gestores de bases de datos se estudian en informática.[8]

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.

Los sistemas de bases de datos (SBD) se diseñan para gestionar grandes cantidades de información. La gestión de los datos implica tanto la definición de estructuras para almacenar la información como la provisión de mecanismos para la manipulación de la misma. Los SBD deben proporcionar la fiabilidad de la información almacenada, a pesar de las caídas del sistema o del intento de acceso sin autorización.

Anteriormente se utilizaba el sistema de procesamiento de archivos en el cual toda la información de la empresa se almacenaba en archivos permanentes manejados por un sistema operativo convencional, se tenían que escribir diversos programas de aplicación para extraer registros y añadir registros a los archivos adecuados.

Los principales inconvenientes de la organización en un sistema de procesamiento de archivos son:



Redundancia e inconsistencia de datos. Como los programas y archivos son creados por diferentes personas a través del tiempo, estos pueden tener diferentes formatos y escritos en diferentes lenguajes de programación y la información puede estar duplicada en los diversos lugares. Esto conduce a un almacenamiento y coste de acceso más altos y puede conducir a inconsistencia de datos.

Dificultad en el acceso de datos. Algunas peticiones no previstas en el sistema original, no hay un programa de aplicación a la mano para hacerlo. Sin embargo hay programas para mostrar la lista total y las opciones son: primero de esa lista sacar la información manualmente y segundo, mandar a un programador a hacer un programa que haga esa consulta. Y por cada consulta se debería hacer lo mismo.

Aislamiento de datos. La creación de nuevos programas de aplicación es muy difícil ya que los archivos pueden estar en diversos formatos y se tiene que recopilar la información dispersa en cada uno de los archivos.

Problemas de integridad. Los valores de los datos almacenados en la base de datos deben satisfacer ciertos tipos de restricciones de consistencia. Por ejemplo, el saldo de ciertos tipos de cuentas bancarias no puede nunca ser inferior a una cantidad predeterminada. Los desarrolladores hacen cumplir esas restricciones en el sistema añadiendo el código correspondiente en los diversos programas de aplicación.

Problemas de atomicidad. Los sistemas informáticos, como cualquier otro dispositivo mecánico o eléctrico, esta sujeto a fallos. En muchas aplicaciones es crucial asegurar que, si se produce algún fallo, los datos se restauren al estado consistente que existía antes del fallo. Es decir que si al existir algún fallo en el sistema de datos y el usuario esta haciendo alguna transacción, que ésta ocurra en su totalidad o que no ocurra en absoluto. Resulta difícil asegurar la atomicidad en los sistemas convencionales de procesamiento de archivos.

Problemas de seguridad. No todos los usuarios de un sistema de base de datos deben poder acceder a todos los datos. Por ejemplo, en un sistema bancario, el personal de nomina solo necesita ver la parte de la base de datos que contiene información acerca de los diferentes empleados del banco. No necesitan tener acceso a la información acerca de las cuentas de clientes. Pero, como los programas de aplicación se añaden al sistema de procesamiento de datos de una forma ad hoc, es difícil cumplir tales restricciones de seguridad.

Aunque las bases de datos pueden contener muchos tipos de datos, algunos de ellos se encuentran protegidos por las leyes de varios países. Por ejemplo en España, los datos personales se encuentran protegidos por la Ley Orgánica de Protección de Datos de Carácter Personal (LOPD). [4]

1.4.1 ABSTRACCIÓN DE DATOS

El propósito principal de un sistema de bases de datos permite acceder y modificar estos archivos. El propósito principal de un sistema de bases de datos es proporcionar a los usuarios una visión abstracta de los datos. Es decir, el sistema esconde ciertos detalles de como se almacenan y mantienen los datos.



Para que el sistema sea útil, debe recuperar los datos eficientemente. Esto ha conducido al diseño de estructuras complejas para la representación de los datos en la base de datos. Los desarrolladores esconden la complejidad a través de los niveles de abstracción para simplificar la interacción de los usuarios con el sistema.

Nivel físico. El nivel más bajo de abstracción, describe como se almacenan realmente los datos. En el nivel físico se describen en detalle las estructuras de datos complejas de bajo nivel.

Nivel lógico. El siguiente nivel de abstracción describe como se almacenan en la base de datos y que relaciones existen entre esos datos.

Nivel de vistas. Este nivel sólo describe parte de la base de datos completa. A muchos usuarios no les importará el nivel lógico, solo querrán ver parte de la información. Para la simplificación de la interacción del usuario se crea un nivel de vistas.

Los componentes de gestión de almacenamiento proporciona la interfaz entre los datos de bajo nivel almacenados en la base de datos y los programas de aplicación y el envío de consultas al sistema. El gestor de almacenamiento tiene:

- Gestor de autorizaciones de integridad que comprueba que se satisfagan las ligaduras de integridad y la autorización de los usuarios para acceder a los datos.
- Gestor de transacciones que asegura que la base de datos quede en un estado consistente a pesar de los fallos del sistema y que las ejecuciones de transacciones concurrentes ocurran sin conflictos.
- Gestor de archivos que gestiona la reserva de espacio de almacenamiento de disco y las estructuras de los datos usadas para representar la información almacenada en el disco.
- Gestor de memoria intermedia que es responsable de traer los datos del disco de almacenamiento a memoria principal y decidir que datos traer a memoria caché.

También se necesitan diversas estructuras de datos como parte de la implementación física del sistema.

- Archivos de datos que almacenan la base de datos en sí.
- Diccionario de datos que almacena los metadatos acerca de la estructura de la base de datos.
- Índices que proporcionan acceso rápido a elementos de datos que tienen valores particulares.
- Datos estáticos que almacenan información estadística sobre los datos en la base de datos, el procesador de consultas usa esta información para seleccionar las formas eficientes para ejecutar una consulta. [4]



1.4.2 USUARIOS DE LA BASE DE DATOS

Hay tres diferentes tipos de usuarios de un Sistema de Base de Datos, diferenciados por la forma en que ellos esperan interactuar con el sistema.

- Programadores de aplicaciones. Responsables de escribir los programas de aplicación de base de datos en algún lenguaje de alto nivel. Estos programas acceden a la base de datos emitiendo la solicitud apropiada al manejador de base de datos. Los programas pueden ser aplicaciones convencionales por lotes o pueden ser aplicaciones en línea, cuyo propósito es permitir al usuario final el acceso a la base de datos desde una estación de trabajo o terminal en línea.
- Usuarios finales. Interactúan con el sistema desde estaciones de trabajo o terminales en línea. Un usuario final puede acceder a la base de datos a través de las aplicaciones en línea mencionadas anteriormente, o bien puede usar una interfaz proporcionada como parte integral del software del sistema de base de datos.
- Administrador de la base de datos (DBA). Es la persona que tiene el control central del sistema. [5]

1.4.3 TIPOS DE BASES DE DATOS

Las bases de datos pueden clasificarse de varias maneras, de acuerdo al criterio elegido para su clasificación:

Bases de datos estáticas. Éstas son bases de datos de sólo lectura, utilizadas primordialmente para almacenar datos históricos que posteriormente se pueden utilizar para estudiar el comportamiento de un conjunto de datos a través del tiempo, realizar proyecciones y tomar decisiones.

Bases de datos dinámicas. Éstas son bases de datos donde la información almacenada se modifica con el tiempo, permitiendo operaciones como actualización y adición de datos, además de las operaciones fundamentales de consulta. Un ejemplo de esto puede ser la base de datos utilizada en un sistema de información de una tienda de abarrotes, una farmacia, un videoclub, etc.

Bases de datos bibliográficas. Sólo contienen un representante de la fuente primaria, que permite localizarla. Un registro típico de una base de datos bibliográfica contiene información sobre el autor, fecha de publicación, editorial, título, edición, de una determinada publicación, etc. Puede contener un resumen o extracto de la publicación original, pero nunca el texto completo, porque sino estaríamos en presencia de una base de datos a texto completo (o de fuentes primarias). Como su nombre lo indica, el contenido son cifras o números. Por ejemplo, una colección de resultados de análisis de laboratorio, entre otras.

Bases de datos de texto completo. Almacenan las fuentes primarias, como por ejemplo, todo el contenido de todas las ediciones de una colección de revistas científicas.

Directorios. Un ejemplo son las guías telefónicas en formato electrónico. [5]



1.5 MODELO DE DATOS

Modelo. Un modelo de datos es básicamente una "descripción" de algo conocido como *contenedor de datos* (algo en donde se guarda la información), así como de los métodos para almacenar y recuperar información de esos contenedores. Los modelos de datos no son cosas físicas: son abstracciones que permiten la implementación de un sistema eficiente de *base de datos*; por lo general se refieren a algoritmos, y conceptos matemáticos. [2]

- Modelo lógico basado en objetos.
- Modelo lógico basado en registro.
- Modelo físico de datos.

1.5.1 MODELO LÓGICO BASADO EN OBJETOS

Se basan para describir datos en el nivel conceptual y de visión. Se caracterizan porque proporcionan una capacidad de estructuración bastante flexible y permiten especificar restricciones de datos explícitamente. Los modelos son:

- a) Entidad Relación
- b) Orientado a Objeto
- c) Binario
- d) Semántica de Datos
- e) Infológico
- f) Funcional de Datos

1.5.1.1. MODELO ENTIDAD – RELACIÓN (E-R)

Denominado por sus siglas como E-R. Este modelo representa la realidad a través de un esquema gráfico empleando la terminología de **entidades**, que son objetos, que existen y son los elementos principales que identifican, el problema a resolver con el diagrama y se distinguen de otros por sus características particulares denominadas **atributos**, el enlace que rige la unión de las entidades esta representada por la **relación** del modelo.

Las entidades pueden ser de dos tipos:

Tangibles. Son todos aquellos objetos físicos que podemos ver, tocar o sentir.

Intangibles. Todos aquellos eventos u objetos conceptuales que no podemos ver, aun sabiendo que existen.

La figura 1.7 muestra los símbolos usados en el diagrama E-R.



Descripción	Simbólicamente se representan
Entidades	
Entidad débil	
Interrelación	
Interrelación débil	
Atributo	
Atributo Llave	
Atributo Multivaluado	
Atributo derivado	
Atributo compuesto	

Figura 1.7. Símbolos usados en el diagrama E-R

1.5.1.2. TIPOS DE RELACIONES

Existen 4 tipos de **relaciones** que pueden establecerse entre entidades, las cuales se establecen con: cuantas entidades de tipo B se puede relacionar una entidad de tipo A.

Relación uno a uno. Se presenta cuando existe una relación, como su nombre lo indica uno a uno, denominado también relación de matrimonio. Una entidad del tipo A solo se puede relacionar con una entidad del tipo B, y viceversa. Esto se puede observar en la figura 1.8.

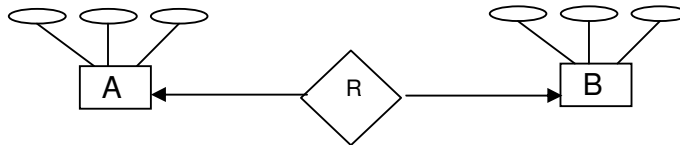


Figura 1.8 Relación uno a uno

A: Representa a una entidad de cualquier tipo diferente a una entidad B.

R: En el diagrama representa a la relación que existe entre las entidades.

El extremo de la flecha que se encuentra punteada indica el uno de la relación, en este caso, una entidad A ligada a una entidad B.

Relación uno a muchos. Significa que una entidad del tipo A puede relacionarse con cualquier cantidad de entidades del tipo B, y una entidad del tipo B solo puede estar relacionada con una entidad del tipo A. Esto se puede observar en la figura 1.9.

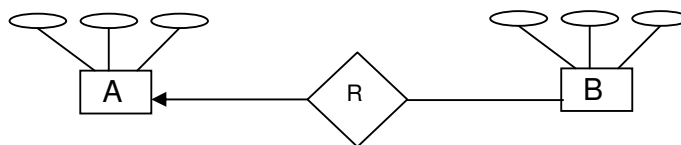


Figura 1.9 Relación uno a muchos



Relación muchos a uno. Indica que una entidad del tipo B puede relacionarse con cualquier cantidad de entidades del tipo A, mientras que cada entidad del tipo A solo puede relacionarse con solo una entidad del tipo B. Esto se puede observar en la figura 1.10.

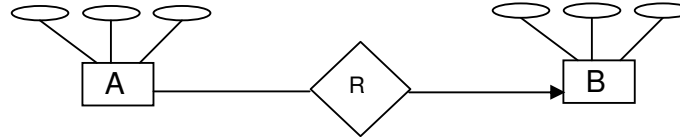


Figura 1.10 Relación muchos a uno

Muchos a muchos. Establece que cualquier cantidad de entidades del tipo A puede estar relacionados con cualquier cantidad de entidades del tipo B. Esto se puede observar en la figura 1.11.

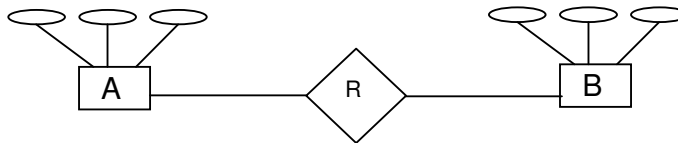


Figura 1.11 Relación muchos a muchos

A los tipos de relaciones antes descritos, también se le conoce como cardinalidad. La cardinalidad nos especifica los tipos de relaciones que existen entre las entidades en el modelo E-R y establecer con esto las validaciones necesarias para conseguir que los datos de la instancia (valor único en un momento dado de una base de datos) correspondan con la realidad.

Otra clase de limitantes lo constituye la dependencia de existencia. Refiriéndonos a las mismas entidades A y B, decimos que si la entidad A depende de la existencia de la entidad B, entonces A es dependiente de la existencia por B, si eliminamos a B tendríamos que eliminar por consecuente la entidad A, en este caso B es la entidad Dominante y A es la entidad subordinada.

1.5.1.3. LLAVES PRIMARIAS

Como ya se ha mencionado anteriormente, la distinción de una entidad entre otra se debe a sus atributos, lo cual lo hacen único. Una llave primaria es aquel atributo el cual consideramos clave para la identificación de los demás atributos que describen a la entidad. Por ejemplo, si consideramos la entidad alumno, podríamos tener los siguientes atributos: nombre, semestre, especialidad, dirección, número de control, de todos estos atributos el que podremos designar como llave primaria es el número de control, ya que es diferente para cada alumno y este los identifica en la institución.

Claro que puede haber más de un atributo que pueda identificarse como llave primaria en este caso se selecciona la que consideremos más importante, los demás atributos son denominados **llaves secundarias**. Una clave o llave primaria es indicada gráficamente en el modelo E-R con una línea debajo del nombre del atributo.



1.5.2 MODELOS LÓGICOS BASADOS EN REGISTROS

Se utilizan para describir datos en los niveles conceptual y físico. Estos modelos utilizan registros e instancias para representar la realidad, así como las relaciones que existen entre estos registros (ligas) o apuntadores. A diferencia de los modelos de datos basados en objetos, se usan para especificar la estructura lógica global de la base de datos y para proporcionar una descripción a nivel más alto de la implementación. Los tres modelos de datos más ampliamente aceptados son:

- Modelo Relacional
- Modelo de Red
- Modelo Jerárquico

1.5.3 MODELO RELACIONAL

En este modelo se representan los datos y las relaciones entre estos, a través de una colección de tablas, en las cuales los renglones (tuplas) equivalen a cada uno de los registros que contendrá la base de datos y las columnas corresponden a las características (atributos) de cada registro localizado en la tupla.

El modelo relacional se ocupa de tres aspectos de los datos:

- Su estructura
- Su integridad
- Su manipulación

Estructura de Datos Relacionales, a manera de ejemplo en la figura 1.12 se presenta la tabla correspondiente a proveedores con los términos que se definen en la estructura de datos relacional.

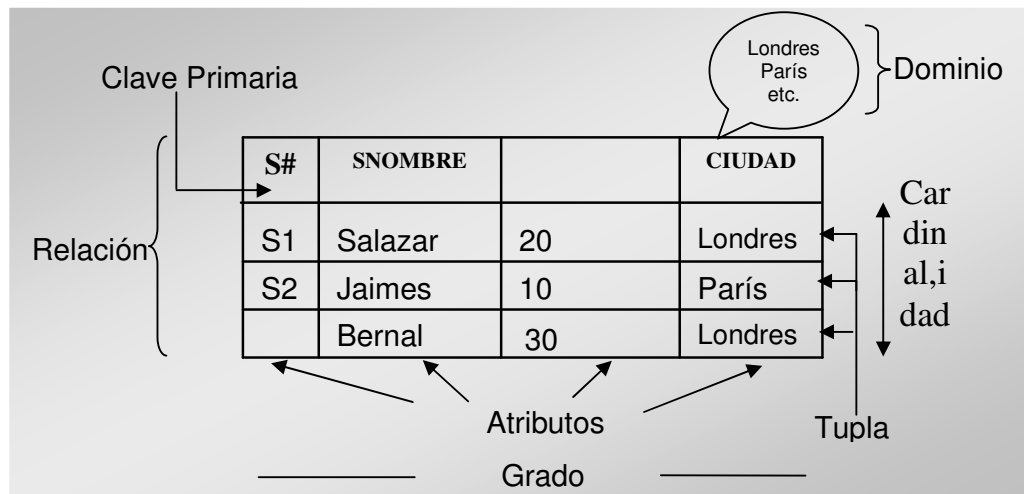


Figura 1.12 Modelo Relacional

En particular una **relación** corresponde a lo que comúnmente se le llama una tabla.



Una tupla corresponde a una fila de esa tabla y un atributo a una columna. El número de tuplas se denomina cardinalidad y el número de atributos se llama grado.

La llave primaria (“clave primaria”) es un identificador único para la tabla. Es decir, una columna o combinación de columnas con la siguiente propiedad: nunca existen dos filas de la tabla con el mismo valor en esa columna o combinación de columnas.

Por el último el dominio es una colección de valores, de los cuales uno a más atributos (columnas) obtiene sus valores reales.[6]

1.6 NORMALIZACIÓN

La normalización es un proceso que consiste en comprobar que las tablas (también denominadas relaciones en terminología propia del modelo relacional de datos) definidas cumplen determinadas condiciones. Se pretende garantizar la no-existencia de redundancia y una cierta coherencia en la representación mediante un esquema relacional de las entidades y relaciones del modelo conceptual (diagrama E-R)

Mediante la normalización se puede solucionar diversos errores en el diseño de la base de datos. También facilita el trabajo posterior del administrador de la base de datos y de los desarrolladores de aplicaciones.

Su finalidad es reducir las inconsistencias y redundancias de los datos, facilitar el mantenimiento y evitar las anomalías en la manipulación de datos. El objetivo será obtener un modelo normalizado que represente las entidades normalizadas y las interrelaciones existentes entre ellas. Para ello se toman como punto teórico de partida el concepto de dependencia funcional que dice: “un atributo B depende funcionalmente de otro atributo A, de la misma entidad si a cada valor de A le corresponde sólo un valor de B”. Lo anterior se completa mediante la dependencia fundamental completa y la dependencia transitiva.

El procedimiento de normalización consiste en someter a las tablas que representan entidades a un análisis formal para ver si cumplen, o no las restricciones necesarias que aseguren evitar los problemas citados con anterioridad. A mayor nivel de normalización, mayor calidad en la organización de los datos y menor peligro para la integridad de los datos. Este procedimiento consiste en ir alcanzando formas normales.

Para lograr la consistencia e integridad de los datos, es necesario mejorar las relaciones. Estas mejoras deben dar como resultado tablas equivalentes, y poseer siempre tres propiedades: conservación de la información (de atributos y de tuplas), conservación de dependencias y mínima redundancia de los datos.

Las mejoras introducidas obligan a plantear hasta que Forma Normal es necesario llegar. Es decir, que nivel de depuración. Normalmente, es recomendable alcanzar la máxima Forma Normal, aunque luego es muy probable que restricciones existentes, de algún tipo, obliguen a retroceder a un nivel inferior de normalización, o incluso a cierto nivel de “desnormalización”. [5]



1.6.1 FORMAS DE NORMALIZACIÓN

Primera forma normal (FN1). Se pretende garantizar la no-existencia de grupos repetitivos. Un grupo repetitivo es un conjunto de atributos de igual semántica en el problema y dominio, que se toman valores distintos para la misma clave. Cualquier esquema que tenga claves correctas esta seguro en FN1.

Segunda forma normal (FN2). Si esta en FN1 y cada atributo de la tabla que no forma parte de la clave dependen funcionalmente de forma completa de la clave primaria. Es decir, depende de toda la clave y no de ningún subconjunto de ella. Se pretende garantizar una correcta elección de claves y eliminar redundancias. Si la clave está formada por un único atributo entonces ese esquema estará seguro en segunda forma normal.

Tercera forma normal (FN3). Si esta en FN2 y cada atributo no primo de la tabla no dependen funcionalmente de forma transitiva de la clave primaria.

BCFN (Forma Normal de Boyce-Codd). Una relación está en la forma normal de Boyce-Codd si, y sólo si, todo determinante es una clave candidata.

La 2FN y la 3FN eliminan las dependencias parciales y las dependencias transitivas de la clave primaria. Pero ese tipo de dependencias todavía puede existir sobre otras claves candidatas, si éstas existen. La BCFN es más fuerte que la 3FN, por lo tanto, toda relación en BCFN está en 3FN.[6]

Además de la clasificación por la función de las bases de datos, éstas también se pueden clasificar de acuerdo a su modelo de administración de datos.

1.7 DEFINICIONES DE TÉRMINOS FINANCIEROS

TENDENCIAS. Trayectoria esperada de comportamiento de los activos financieros, realizada en base a hipótesis obtenidas mediante el análisis técnico y fundamental de los mercados financieros.

INDICADOR. Cifra representativa de la situación económica para un período determinado. Estos indicadores son seguidos por los inversores para tomar sus decisiones de compra o de venta sobre los distintos activos financieros o los bienes tangibles.

PREDICCIÓN. Se trata del análisis racional de lo que va a suceder, en otras palabras, es mediante un cálculo poder saber como se comportaran los indicadores en el futuro. [11]

PARIDAD. Relación existente entre una moneda y su equivalente en oro. Relación entre los tipos de cambio de dos monedas respectivamente. Paridad del poder adquisitivo. Teoría que propugna que los tipos de cambios deben permitir convertir los precios extranjeros en precios nacionales, es decir, la conversión de una cantidad de dinero en otra divisa debe realizarse de forma tal que se mantenga su capacidad adquisitiva en los dos países.



PRODUCTO INTERNO BRUTO (PIB) Este indicador se define como el valor de mercado de la producción de bienes y servicios de un país, antes de deducir la depreciación de los bienes de capital. Los factores más importantes que determinan el volumen del PIB son: nivel de fuerza laboral, número de personas que trabajan, cantidad de días y horas laboradas durante el período, productividad de la mano de obra, nivel de precios, tecnología y equipo utilizado en la producción.

Al modificarse el tipo de cambio, al Producto Interno Bruto registrará, en el corto plazo, un ligero aumento provocado por una relación favorable de intercambio generada por una disminución en los precios de los productos generados en las economías en las que se registra una variación negativa del tipo de cambio, ello debido a que el valor de su tipo de cambio

INFLACIÓN. Se define como el rápido incremento en el nivel de precios de los bienes y servicios, debido al aumento generalizado del gasto total en relación a la oferta de bienes y servicios del mercado. Usualmente se asocia el término de inflación con el aumento en los salarios y de allí en los costos de producción (llamada inflación de costos) o a una expansión de la cantidad de dinero superior al crecimiento de la producción de mercancías y una correspondiente reducción en el poder de compra de los consumidores, debido a que los precios aumentan más que el ingreso de las familias (llamada inflación monetaria). Las causas más comunes de la inflación son una elevada demanda de crédito, un alto gasto del gobierno por encima de sus ingresos, demandas continuas de incremento salarial, tendencia a gastar debido a la creencia que el poder de compra se reducirá en el futuro.

La inflación se mide comparando los niveles de precios de diferentes períodos. Estos niveles de precios usualmente se refieren a dos conceptos que toman en consideración los productos y servicios que son consumidos por las familias, conocido como precios al consumidor, o productos y servicios que son utilizados por las empresas para agregarles algún valor, conocido como precios al productor.

ÍNDICE NACIONAL DE PRECIOS AL CONSUMIDOR. Se define como el promedio ponderado de los bienes de un conjunto específico de bienes y servicios consumidos por las familias, conocido como la canasta básica o de mercado, el cual es convertido a una serie de tiempo que relaciona los precios de un período con los precios de otro período. Las ponderaciones se basan en la importancia relativa que las familias asignan al gasto, de acuerdo al nivel de sus ingresos.

REMESAS. Son las cantidades de dinero enviado por migrantes a sus países de origen en toda América Latina. Las cantidades de dinero son tan grandes cada año que en algunos países han desplazado a las exportaciones tradicionales del principal rubro de ingresos de la economía nacional. [10]



1.8 ARQUITECTURA CLIENTE-SERVIDOR

ARQUITECTURA. Un entramado de componentes funcionales que aprovechando diferentes estándares, convenciones, reglas y procesos, permite integrar una amplia gama de productos y servicios informáticos, de manera que pueden ser utilizados eficazmente dentro de la organización.

CLIENTE. Es el que inicia un requerimiento de servicio. El requerimiento inicial puede convertirse en múltiples requerimientos de trabajo a través de redes LAN o WAN. La ubicación de los datos o de las aplicaciones es totalmente transparente para el cliente.

SERVIDOR. Es cualquier recurso de cómputo dedicado a responder a los requerimientos del cliente. Los servidores pueden estar conectados a los clientes a través de redes LANs o WANs, para proveer de múltiples servicios a los clientes y ciudadanos tales como impresión, acceso a bases de datos, fax, procesamiento de imágenes, etc.

Normalmente en Internet se usa la arquitectura cliente-servidor. Este tipo de organización se basa en que entre todos los ordenadores que están en la red, unos ofrecen servicios (los llamados servidores) y otros usan esos servicios (los denominados clientes). Como ejemplo cuando están visionando estas páginas, están accediendo a un servicio (pidiendo una página WEB concreta) que les ofrece nuestro servidor de páginas WEB (sirviéndole la página solicitada). Por lo tanto, su ordenador es un cliente y el que hospeda estas páginas es un servidor. En la figura 1.13 se muestra un ejemplo de esta arquitectura.



Fig. 1.13 Ejemplo de arquitectura cliente-servidor



1.9 CÓDIGO ABIERTO

PHP, Apache y MySQL forman parte del grupo de programas de software de código abierto. Básicamente, el movimiento de código abierto es una colaboración de algunos de los mejores profesionales de la programación informática. Al permitir el libre intercambio de la información, los programadores de todo el mundo contribuyen en la creación de un programa completo y eficaz disponible para todos. Mediante estas contribuciones, se corrigen problemas, se realizan mejoras y un buen programa se convierte, con el tiempo, en un programa excelente. [7]

La preferencia por los programas de código abierto se debe a las siguientes razones:

- Son gratuitos
- Son compatibles entre plataformas
- No deben restringir el uso de otros programa
- Deben ser diversos

1.9.1 APACHE

El nombre de Apache viene de **A PatCH** y **sErver**. Apache actúa como servidor Web. Su labor principal consiste en analizar cualquier archivo solicitado por un navegador y mostrar los resultados correctos en función del código del archivo. Apache es muy potente y puede realizar prácticamente cualquier tarea que se le pida.

Entre las funciones de las opciones de servidor que incluye destacamos las siguientes:

- Páginas protegidas con contraseñas para múltiples usuarios.
- Páginas de errores personalizadas.
- Representación de código en numerosos niveles de HTML y la posibilidad de determinar a que nivel el navegador puede aceptar el contenido.
- Registros des uso y de errores en diferentes formatos personalizables.
- Alojamiento virtual de diferentes direcciones IP asignadas al mismo servidor.
- Directivas Directory Index a varios archivos.
- Reescritura de URL sin límite fijo.

Su flexibilidad, prestaciones y, evidentemente, su precio, lo han convertido en la principal opción. Se puede utilizar para alojar un sitio Web para el público general, una intranet de una empresa o simplemente para probar páginas personales antes de cargarlas a un servidor seguro en otro equipo.



1.9.2 PHP

PHP es un lenguaje programación (originario del nombre **PHP Tools**, o **Personal Home Page Tools**) de secuencias de comandos del lado del servidor que permite que un sitio Web sea realmente dinámico. PHP (preprocesador de hipertexto). Su flexibilidad y su curva de aprendizaje relativamente corta lo convierten en uno de los lenguajes de programación más conocidos. Su popularidad sigue en aumento debido a la acogida por parte de empresas y particulares como alternativa al lenguaje ASP de Microsoft y porque sus ventajas superan a los costes.

El **lenguaje PHP** tiene la característica de poder mezclarse con el lenguaje HTML. **PHP**, al contrario que este último, se interpreta y ejecuta directamente en el servidor en el que está albergada la página Web, con lo que el visitante a la misma únicamente recibe el resultado buscado por el código en el que está escrito.

1.9.3 MYSQL

MySQL es el generador de bases de datos que permiten a PHP y a Apache trabajar de forma conjunta para acceder y mostrar datos en un formato legible en un navegador. Se trata de un servidor SQL (Lenguaje de Consulta estructurado) diseñado para grandes cargas y que puede procesar consultas muy complejas. Como sistema de bases de datos relacional, MySQL permite combinar multitud de tablas diferentes para optimizar la eficacia y la velocidad.

A continuación enumeramos algunas funciones más conocidas:

- Uso de varias CPU a través de subprocesos del núcleo.
- Funcionamiento entre plataformas.
- Diferentes tipos de columnas que abarcan prácticamente cualquier tipo de datos.
- Agrupación de funciones para realizar cálculos matemáticos y tareas de ordenamiento.
- Comandos que permiten mostrar, de forma sencilla y suscita, información sobre las bases de datos al administrador.
- Nombres de funciones que no afectan a los nombres de tablas o columnas.



CAPÍTULO II

ANÁLISIS DE REQUERIMIENTOS



2.1 PLANTEAMIENTO DEL PROBLEMA

De esta problemática surge la idea de realizar un proyecto de tesis en el cual se aplique el desarrollo de un Sistema de Búsqueda de Tendencias Económicas en un Sistema de Base de Datos Financiero. Por lo que en este trabajo se propone como objetivo el desarrollo de un sistema flexible que permita acceder a la información de los principales indicadores económicos.

Los usuarios son estudiantes de economía, economistas o en todo caso personas interesadas en el tema, que requieren información sobre los indicadores y tendencias económicas.

El Sistema financiero, debe estar organizado y debe ofrecer los servicios de búsqueda y recuperación de la información vía Web. Además, se mostrarán los gráficos de las tendencias y predicciones de algunos indicadores.

2.2 REQUERIMIENTOS DEL SISTEMA

- Administrar la información del comportamiento de los indicadores de las principales tendencias económicas de nuestro país.
- Generar histogramas.
- Crear una página Web.
- Proporcionar consultas de los gráficos de tendencias y predicciones económicas.
- Mantener actualizada la información de los indicadores económicos.

2.2.1 REQUERIMIENTOS FUNCIONALES

- El sistema debe contemplar los siguientes aspectos:
 - **Almacenamiento y Clasificación de los indicadores.** La captura de la información se debe llevar a cabo por el Administrador del Sistema, clasificando los indicadores económicos en:
 - Paridad peso – Dólar
 - Producto Interno Bruto
 - Remesas
 - Índice de Precios al consumidor
 - Inflación
 - **Búsqueda y consulta.** La información almacenada en la base de datos debe ser accesible a los usuarios de manera eficiente, ofreciendo los servicios de búsqueda de datos.

La búsqueda de la información se refiere a buscar los indicadores por mes, o en algunos casos por año.



-
- **Interfaz para la presentación de Resultados y gráficos.** La interfaz debe ser amigable y de fácil acceso y los gráficos entendibles para el usuario. Se refiere a la generación de los gráficos por medio de la base de datos.
 - **Administración y Control de la base de datos.** El Administrador de la Base de datos es un usuario con derechos de acceso, quien puede realizar las tareas de administración correspondientes y de mantenimiento, el administrador ingresará al Sistema mediante un nombre de usuario y contraseña.

El sistema permite que el Administrador de Sistema configure el Servidor y maneje la información concediendo derechos a los usuarios.

2.2.2 REQUERIMIENTOS NO FUNCIONALES

Los aspectos que pueden limitar el funcionamiento del sistema son:

- El que no haya acceso al Servidor
- No es posible garantizar el acceso adecuado por tráfico de la red.

2.3 ALCANCE

A través del Sistema Financiero antes mencionado, las personas tendrán acceso a la información que ellos necesitan, y podrán consultarlo en cualquier momento desde una computadora conectada a la Web, para dar solución a sus necesidades.

De esta manera se pondrá en marcha el Sistema de Búsqueda de Tendencias Financieras de modo funcional y flexible

2.4 METODOLOGIA

Se usó el modelo en cascada para organizar el proceso de desarrollo del sistema, es decir, se realizó siguiendo las etapas de análisis, diseño, codificación, prueba y mantenimiento.

Inicialmente se realizó la recopilación de la información y el análisis de requerimientos para comprender las necesidades de los usuarios, y así llegar a elegir la forma en que se dará solución al problema, y la creación del documento de requerimientos.

Posteriormente se usó el modelo conceptual y el modelo relacional, para definir la estructura, integridad, manipulación de los datos.

A continuación, se realizó la codificación en donde se genera el programa que compone el sistema, las interfaces, así mismo la Base de Datos.

Finalmente, se realizaron las pruebas, para verificar que todos los componentes del sistema funcionan correctamente y asegurar que la entrada produce los resultados que realmente se requerían.



2.5 RECURSOS DEL SISTEMA

- La información fue almacenada en un servidor de Base de Datos que tendrá los servicios del Servidor Web o APACHE y se puede tener acceso mediante una computadora.
- El Sistema se desarrolló en un lenguaje enfocado en aplicaciones a la Web, como PHP, HTML, y MySQL.
- El Sistema cuenta con un entorno gráfico, sencillo y amigable en línea para consultar la Información Financiera.

- Los requerimientos mínimos para el Servidor son:

Software

- Tener instalado un Sistema Operativo
- Tener instalado un Servidor Web o Apache
- Tener instalado el programa PHP
- Tener instalado el programa MySQL

Hardware

- Una computadora Intel Pentium III o superior, puede ser alguna compatible.
- 128 Mb en RAM como mínimo
- 1Gb de memoria libre en disco duro

- Los requerimientos mínimos para el Cliente son:

Software

- Tener instalado un Sistema Operativo
- Tener instalado un navegador de Internet.

Hardware

- Una computadora Intel Pentium III o superior, también puede ser compatible.
- Una conexión para internet
- 128 Mb en RAM como mínimo.

2.6 REQUERIMIENTOS DE LA INTERFAZ

Interfaz de usuario

Como la aplicación fue desarrollada y dirigida a la ejecución en un entorno visual, la interacción entre el usuario y la aplicación se realizó mediante pantallas típicas en un entorno de tipo formulario con ventanas, botones, etiquetas, etc.



Interfaces del Software

Se aplicó la arquitectura Cliente-Servidor.

El servidor debe contar con una base de datos, con un Servidor de la Web o APACHE y herramientas de programación para Internet como los lenguajes PHP, HTML y MySQL.

El Sistema debe utilizar un Administrador de Base de Datos

Interfaces del Hardware

No es necesario que la computadora que tome la función de servidor tenga un mínimo de espacio en el disco duro debido a que el Sistema funcionará mediante vía Web o APACHE, pero si deberá contar con una conexión a Internet.

Interfaces de Comunicación

La computadora donde se implantó el Sistema deberá contar con el protocolo necesario para una conexión al Servidor de la Web o APACHE como TCP/IP.

2.7 DISEÑO DEL SISTEMA

2.7.1 DIAGRAMA DE FLUJO DE DATOS (D.F.D)

El Diagrama de Flujo de Datos representa cada uno de los procesos del sistema, estos son una técnica para especificar requerimientos, representan una técnica de análisis estructurada grafica. Además se refinan en niveles. El primer nivel consta de un solo proceso que representa el sistema completo.

D.F.D. NIVEL 0:

Este Sistema cuenta con cuatro funciones principales:

- 1) Altas
- 2) Actualización
- 3) Bajas
- 4) Búsqueda
- 5) Graficar

Se representan mediante el D.F.D, iniciando con el diagrama de Nivel 0 como se muestra en la figura 2.1. El Sistema recibe información de entidades externas, la cual es procesada y genera información a entidades externas.

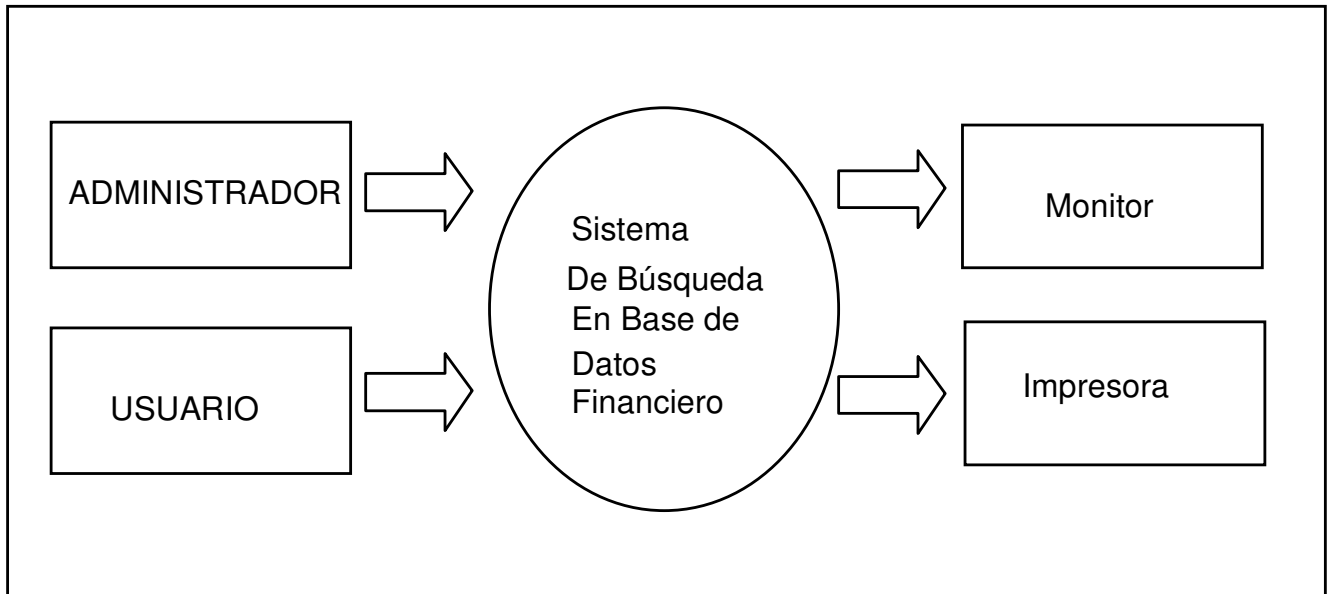


Figura 2.1 Nivel 0

D.F.D. NIVEL 1:

A continuación se describe el flujo de la información más detallado, mostrando las entradas y salidas mediante la figura 2.2

- 1) El administrador accesa con un nombre de usuario y una contraseña, si no se reconoce la contraseña, visualiza un mensaje de error, en caso contrario el Administrador puede modificar los datos.
- 2) El Administrador puede dar de alta, baja y hacer actualizaciones en Paridad Peso-Dólar, Producto Interno Bruto, Remesas, Índice Nacional de Precios al Consumidor, Inflación y Tendencias.
- 3) Se generan histogramas de las tendencias y predicciones.
- 4) El usuario pide buscar o consultar y el sistema muestra en pantalla o en impresora los resultados.

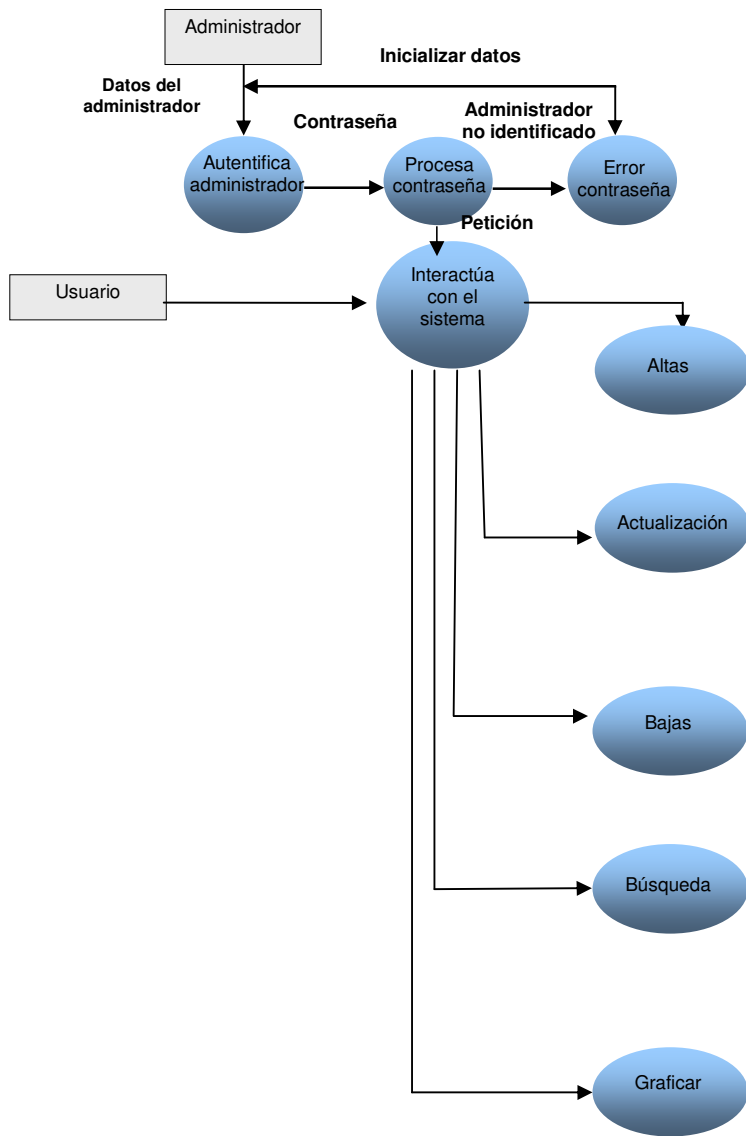


Figura 2.2 D.F.D. Nivel 1



D.F.D. NIVEL 2 :

En las figuras 2.3, 2.4, 2.5, 2.6 y 2.7 se describen cada una de las funciones que realiza el Sistema.

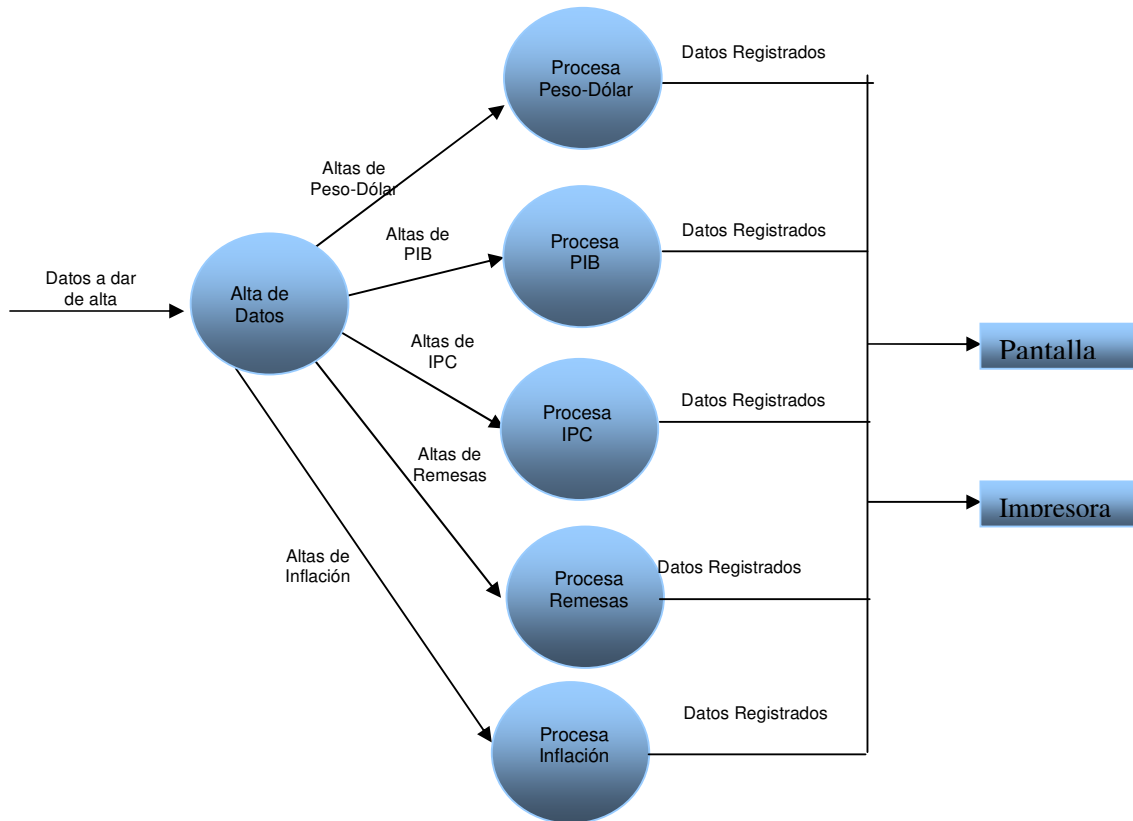


Figura 2.3 D.F.D Nivel 2 Altas

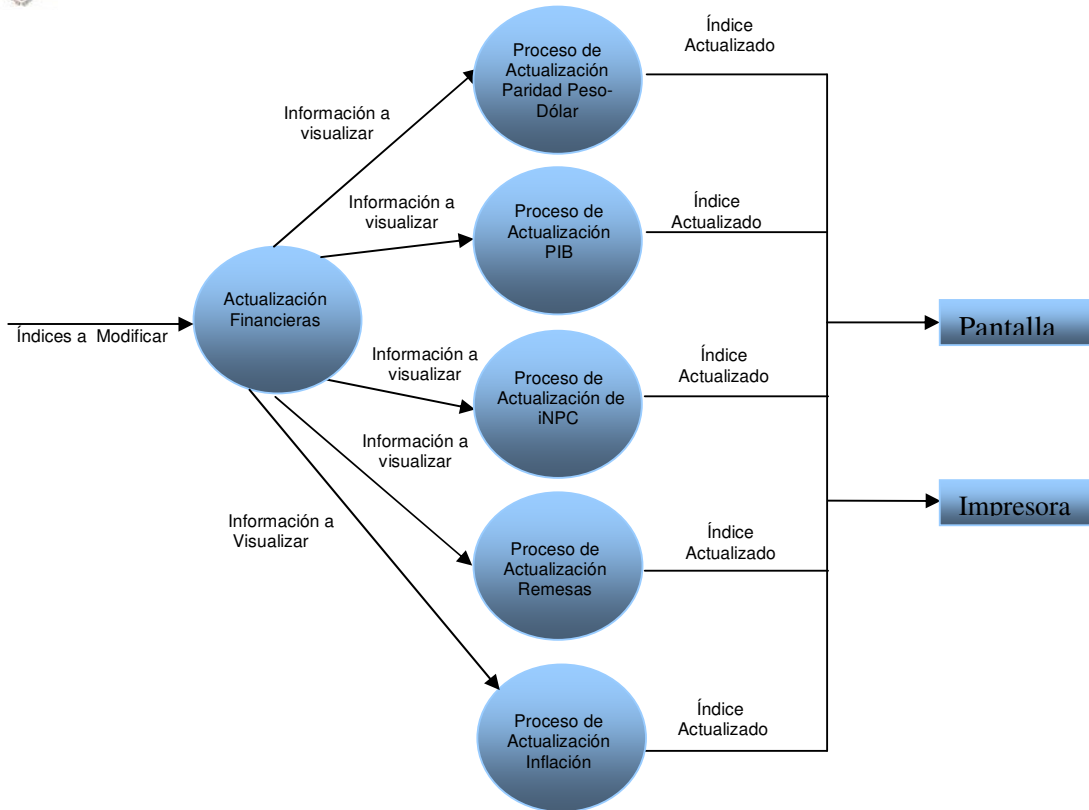


Figura 2.4 D.F.D Nivel 2. Actualización

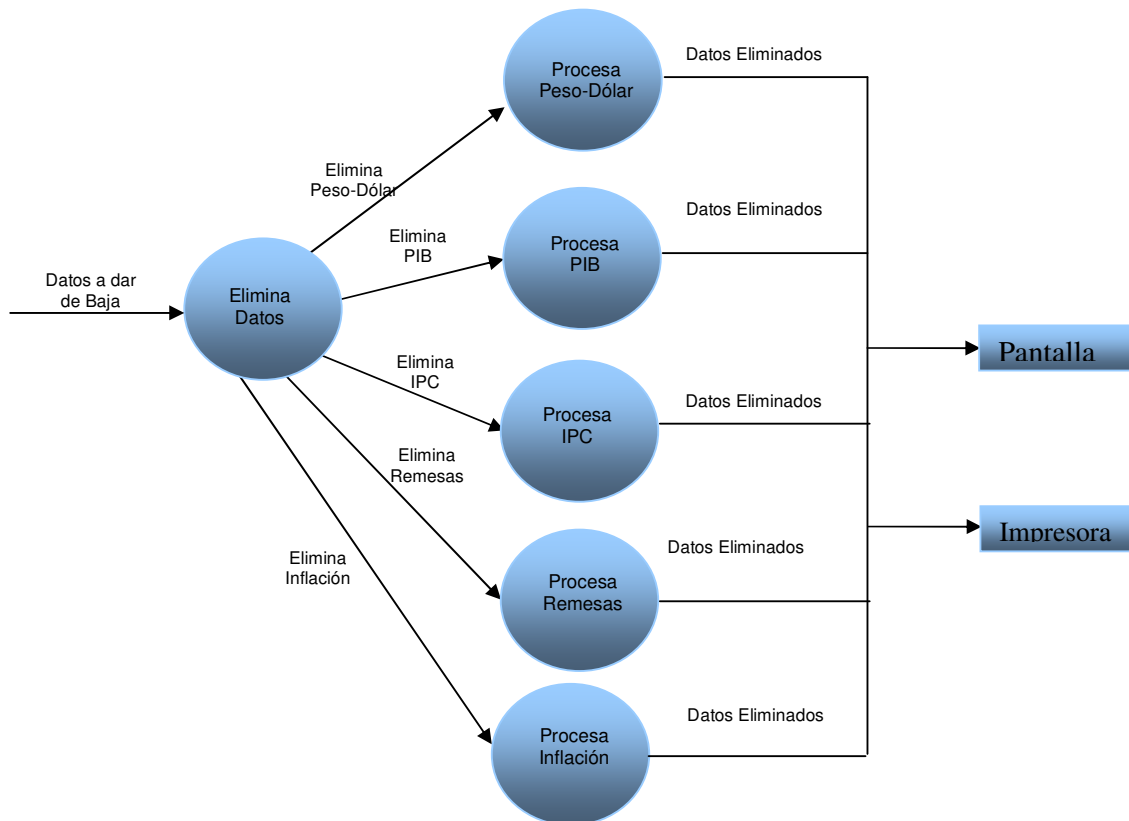


Figura 2.5 D.F.D Nivel 2 Bajas

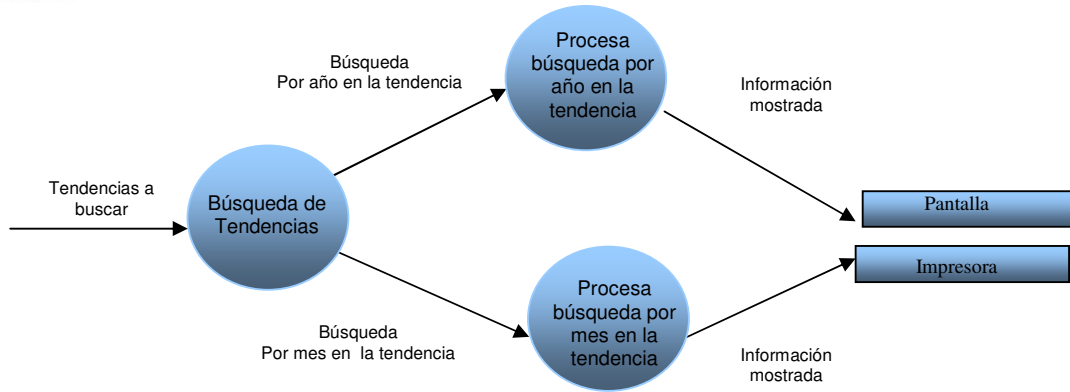


Figura 2.6 D.F.D Nivel 2. Búsquedas

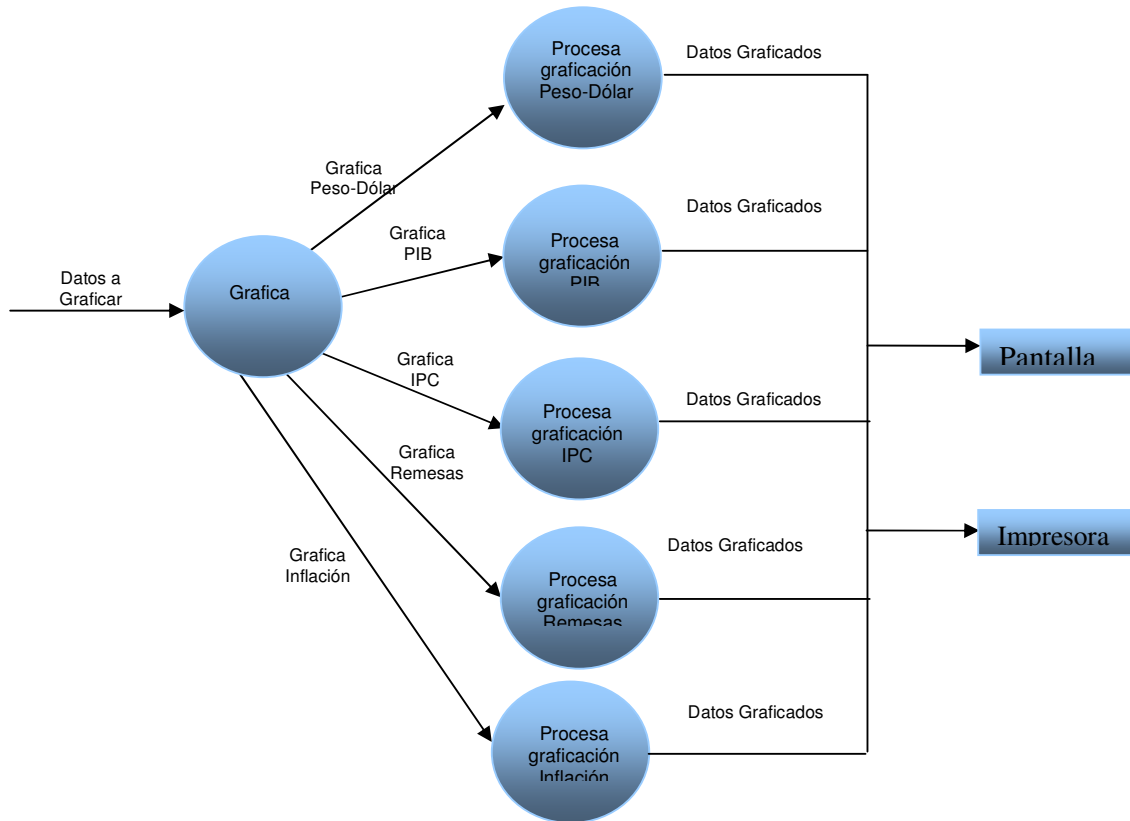


Figura 2.7 D.F.D Nivel 2. Graficación



CAPÍTULO III

DISEÑO DE LA BASE DE DATOS



3.1 DISEÑO DE LA BASE DE DATOS

El diseño de una base de datos es un proceso complejo que abarca decisiones a muy distintos niveles. El diseño se controla mejor si se descompone el problema en subproblemas y se resuelve cada uno de estos subproblemas independientemente, utilizando técnicas específicas. Así, el diseño de una base de datos se descompone en diseño conceptual y diseño lógico.

El diseño conceptual parte de las especificaciones de requisitos de usuario y su resultado es el esquema conceptual de la base de datos. Un esquema conceptual es una descripción de alto nivel de la estructura de la base de datos. Un modelo conceptual es un lenguaje que se utiliza para describir esquemas conceptuales; el objetivo del diseño conceptual es describir el contenido de información de la base de datos.

El diseño lógico parte del esquema conceptual y da como resultado un esquema lógico. Un esquema lógico es una descripción de la estructura de la base de datos en términos de las estructuras de datos que puede procesar un tipo de sistema de bases de datos.

Un modelo lógico es un lenguaje usado para especificar esquemas lógicos como lo es el modelo relacional que se usa posteriormente. [9].

3.2 DISEÑO CONCEPTUAL DE LA BASE DE DATOS

En esta etapa se construyó un esquema de la información para tener más claro lo que contendrá nuestro sistema, de esta manera se describen los datos así como, atributos y relaciones.

3.2.1 DESCRIPCION DE ENTIDADES Y RELACIONES

En el diseño conceptual se definen las entidades y la relación o las relaciones existentes entre las entidades como se muestra en la Tabla 2.1

Entidad	DESCRIPCION	Atributos	Cardinalidad	Entidades con las que se relaciona
anual	La relación que existe es que cada indicador se presenta anualmente.	Id_anual, year, ppd, remesas, pib, inpc	UNO A VARIOS	Ppd_tot, pib_tot, ind_tot, remesas, inflación. trimestral
ppd_tot	Se relaciona principalmente porque son tipos de cambio	ID_ppt, Periodo_ppt Promedio_p, Final_p	UNO A UNO UNO A UNO VARIOS A UNO	interbancario tc_obligaciones anual
tc_obligaciones	La relación que existe es que es tipo de cambio	Id_tc, Periodo_tc, Fecha_det,	UNO A UNO	ppd_tot



		Fecha_dof ND_tc		
interbancario	La relación que existe es que es tipo de cambio	Id_i, Periodo_i, Ap_compra, Ci_compra, Ap_venta, Ci_venta, Max, Min, ND_i	UNO A UNO	Ppd_tot.
pib_tot	Se relaciona porque es la forma en que se puede representar el PIB.	ID_pibt, Periodo_pibt, Total_pib	UNO A UNO UNO A UNO UNO A UNO VARIOS A UNO	sec_eco, div_ec e ind_man anual
sec_eco	La relación es porque es una forma de representar el PIB.	Id_se, Periodo_se, A_S_P, Industrial, Servicios, ND_se	UNO A UNO	pib_tot
div_ec	La relación es porque es una forma de representar el PIB.	Id_da, Periodo, Agropecuario, Minería, Industria, Construcción, Electricidad, Comercio, Transporte, Ser_fin, Ser_com, Cargo_servicios, Imp_netos_sub, Val_agr, Total, ND_da	UNO A UNO	pib_tot
ind_man.	La relación es porque es una forma de representar el PIB.	Id_im , Periodo_im, Productos, Textiles, Ind_madera, Papel, Sust_quim, Min_mo_met, Ind_met, Prod_met, Otras, Total ND_im	UNO A UNO	pib_tot
ind_total.	Se relaciona porque es la forma en que se puede representar el INPC.	ID_int, Periodo_int, Indice_total	UNO A UNO UNO A UNO UNO A UNO VARIOS A UNO	p_indices origen_bienes durab_bienes anual
p_indices.	La relación es porque es una forma de representar el INPC.	Id_imp, Periodo_in Indice, Subyacente, Canasta, Mercancías.	UNO A UNO	ind_tot
origen_bienes.	La relación es porque es una	Id_ob, Periodo_ob,	UNO A UNO	ind_tot



	forma de representar el INPC.	Primario, Terciario, Secundario		
durab_bienes.	La relación es porque es una forma de representar el INPC.	Id_db, Periodo_ob, Durables, No_durables, Mercancías, Servicios	UNO A UNO	ind_tot
remesas.	Se relaciona con anual porque también se presenta anualmente	Id_re, Periodo_r, Orden_dinero, Cheques, Transferencias, Efectivo, Total, ND_r, Id_re	VARIOS A UNO	anual.
inflación.	Se relaciona con anual porque también se presenta anualmente	Id_in, Periodo_in, Mensual, Acumulada, ND_in	VARIOS A UNO	anual.
trimestral	Se relaciona porque es una forma de representar el anual	<u>Id_tri</u> , <u>rem_tri</u> , <u>ten_rem</u> , <u>par_tri</u> , <u>ten_par</u> , <u>inpc_tri</u> , <u>ten_inpc</u> , <u>inf_tri</u> , <u>ten_inf</u> , <u>pib_tri</u> , <u>ten_pib</u>	VARIOS A UNO	anual

Tabla 2.1 Descripción de entidades



3.1.2 DICCIONARIO DE DATOS

Un diccionario de datos contiene las características lógicas de los datos que se van a utilizar en la Base de Datos, incluyendo el nombre del atributo, descripción, tipo de dato, longitud, valor por defecto, si el dato es nulo, compuesto, derivado multivaluado y el dominio. El diccionario de datos se muestra en la figura 2.7. donde se encuentra la lista de todos los elementos que forman parte de la Base de Datos.

NOMBRE	DESCRIPCIÓN	TIPO DE ENTIDAD/ATRIBUTO	TIPO DE DATO	LONGITUD	NULO	DOMINIO
anual	Es el total anual de cada uno de los indicadores económicos que se encuentran en nuestro sistema	Fuerte				
Id_anual	Es el identificador o llave primaria de la tabla	Simple monovalorado	Int	3	No	Los números enteros
year	El año donde se muestra la información	Simple monovalorado	Int	4	No	A partir del año 1995
ppd	Es el indicador anual de paridad peso-dólar.	Simple monovalorado	Float	10	No	Los números decimales
pib	Es el indicador anual de producto interNo bruto	Simple monovalorado	Varchar	12	No	Son números separados con coma.
pemasas	Es el indicador anual de remesas	Simple monovalorado	Float	12	No	Números decimales
inpc	Es el indicador anual de índice nacional de precios al consumidor	Simple monovalorado	Flota	12	No	Números decimales
inflación	Es el indicador anual de inflación, Se presenta en porcentaje.	Simple monovalorado	Float	12	No	Números decimales
ppd_tot	Representa la paridad peso-dólar promedio y final total mensual,	Fuerte				
ID_ppd	Es el identificador o llave primaria de la tabla	Simple monovalorado	Int	3	No	Los números enteros
Anual_id_anual	Es la llave foránea que resulta de la relación que hay con la tabla anual	Simple monovalorado	Int	3	No	Los números enteros
Periodo_ppt	Es el mes y el año en que se mostrara la información	Simple monovalorado	Varchar	9	No	Se compone de letras y números
promedio_p	Es la paridad total promedio del mes	Simple monovalorado	Float	10	No	Números decimales
final_p	Es la paridad total final del mes	Simple monovalorado	Float	10	No	Números decimales
tc_obligaciones	Esta entidad se refiere al tipo de cambio para solventar obligaciones	Fuerte				
Id_tc	Es el identificador o llave primaria de la tabla	Simple monovalorado	Int	3	No	Números enteros
Ppd_tot_ID_ppt	Es la llave foránea	Simple monovalorado	Int	3	No	Números enteros
Periodo_tc	Las primeras 3 letras de los meses junto con el año	Simple monovalorado	Varchar	9	No	Letras y números
Fecha_det	Es la cantidad determinada por el Banco de México en un promedio de las cotizaciones del mercado de cambios al mayoreo para operaciones liquidables el segundo día hábil bancario siguiente.	Simple monovalorado	Float	10	No	Números decimales
Fecha_dof	Es la publicación un día hábil bancario después de la determinación.	Simple monovalorado	Float	6	No	Números decimales
interbancario	Esta entidad se refiere al tipo de cambio interbancario	Fuerte				



Id_i	Es el identificador o llave primaria	Simple monovalorado	Int	3	No	Números enteros
Ppd_tot_ID_ppt	Es la llave foránea	Simple monovalorado	Int	3	No	Números enteros
Periodo_i	Las primeras letras del mes junto con el año	Simple monovalorado	Varchar	9	No	Números y letras
Ap_compra	El valor del dólar a la compra al abrir ventanillas	Simple monovalorado	Float	10	No	Los números decimales
Ap_venta	El valor del dólar a la venta al abrir ventanillas	Simple monovalorado	Float	10	No	Números decimales
Ci_compra	EL valor del dólar a la compra al cerrar ventanillas	Simple monovalorado	Float	10	No	Números decimales
Ci_venta	El valor del dólar a la cerrar ventanillas	Simple monovalorado	Float	10	No	Números decimales
Min	Es la cotización mínima en el mes	Simple monovalorado	Float	10	No	Números decimales
Max	Es la cotización máxima en el mes	Simple monovalorado	Float	10	No	Números decimales
ND_i	Se refiere al numero que diferencia el tipo de cambio interbancario: Mismo día=1 A 24 horas=2 A28 horas=3	Simple monovalorado	Int	1	No	Tres números: 1, 2 o 3.
pib_tot	Esta entidad representa en producto interno bruto total mensual. Se relaciona porque es la forma en que se puede representar el PIB	Simple monovalorado				
ID_ppd	Es el identificador o llave primaria de la tabla	Simple monovalorado	Int	3	No	Los números enteros
Anual_id_anual	Es la llave foránea que resulta de la relación que hay con la tabla anual	Simple monovalorado	Int	3	No	Los números enteros
Periodo_pibt	Es el mes y el año en que se mostrara la información	Simple monovalorado	Varchar	9	No	Se compone de letras y números
Total_pib	Es el producto interno bruto total del mes	Simple monovalorado	Varchar	13	No	Se compone de números separados por comas
sec_eco	Representa el producto interno bruto por sector económico.	Fuerte				
ID_se	Es el identificador o llave primaria, se auto incrementa	Simple monovalorado	Int	3	No	Números enteros
Pib_tot_ID_pibt	Es la llave foránea que resulta de la relación con pib_tot	Simple monovalorado	Int	3	No	Números enteros
Periodo_se	Las 3 primeras letras de los meses más en año. Se presenta trimestralmente	Simple monovalorado	Varchar	12	No	Números y letras
A_S_P	Se refiere al sector de agricultura, silvicultura y pesca	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Industrial	Se refiere al sector industrial	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Servicios	Se refiere al sector económico de servicios	Simple monovalorado	Varchar	13	No	Números representando



						miles con comas
ND_se	Numero diferenciador: Precios corrientes=1. Precios a 1993=2	Simple monovalorado	Int	1	No	Dos números: 1 o 2
div_ec	Representa el producto interno bruto por división de actividad económica	Fuerte				
ID_da	Es el identificador o llave primaria, se auto incrementa	Simple monovalorado	Int	3	No	Números enteros
Pib_tot_ID_pibt	Es la llave foránea que resulta de la relación con pib_tot	Simple monovalorado	Int	3	No	Números enteros
Periodo	Las 3 primeras letras de los meses más en año. Se presenta trimestralmente	Simple monovalorado	Varchar	12	No	Números y letras
Agropecuario	Se refiere a actividad economía de agropecuario y silvicultura	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Minería	Se refiere a la minería	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Industria	Se refiere a la Industria manufacturera	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Construcción	Se refiere a la construcción	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Electricidad	Se refiere a la electricidad, gas y agua	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Comercio	Se refiere al comercio, restaurantes y hoteles	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Transporte	Se refiere a transporte, almacenaje y comunicaciones	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Ser_fin	Se refiere a servicios financieros, seguros, actividades inmobiliarias y de alquiler	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Ser_com	Se refiere a servicios comunales, sociales y personales	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Cargo_servicios	Se refiere a los cargo por los servicios bancarios imputados	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Imp_netos_sub	Se refiere a los impuestos netos de subsidio	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Val_agr	Valor agregado	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Total	Es el total de la división de actividad económica	Simple monovalorado	Varchar	13	No	Números representando miles con comas
ND_da	Numero diferenciador: Precios corrientes=1 Precios a 1993=2	Simple monovalorado	Int	1	No	Dos números: 1 o 2
ind_man.	Representa el producto interno bruto por industria manufacturera	Fuerte				



ID_im	Es el identificador o llave primaria, se auto incrementa	Simple monovalorado	Int	3	No	Números enteros
Pib_tot_ID_pibt	Es la llave foránea que resulta de la relación con pib_tot	Simple monovalorado	Int	3	No	Números enteros
Periodo_im	Las 3 primeras letras de los meses más en año. Se presenta trimestralmente	Simple monovalorado	Varchar	12	No	Números y letras
Productos	Se refiere a productos alimenticios, bebidas y tabaco	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Textiles	Se refiere a la textiles, prendas de vestir e industria de cuero	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Ind_madera	Se refiere a la Industria de la madera y productos de madera	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Papel	Se refiere a papel, productos de papel, imprentas y editoriales	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Sust_quim	Se refiere a sustancias químicas, derivados del petróleo, productos de caucho y plástico	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Min_No_met	Se refiere a los productos de minerales No metálicos, excepto derivados del petróleo y carbón	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Ind_met	Se refiere a las industrias metálicas básicas	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Prod_met	Se refiere a los productos metálicos, maquinaria y equipo	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Otras	Se refiere a otras industrias manufactureras	Simple monovalorado	Varchar	13	No	Números representando miles con comas
Total	Se refiere al total de la industria manufacturera	Simple monovalorado	Varchar	13	No	Números representando miles con comas
ND_im	Se refiere a los impuestos netos de subsidio	Simple monovalorado	Varchar	13	No	Números representando miles con comas
ind_total.	Esta entidad representa el índice nacional de precios al consumidor	Fuerte				
ID_int	Es el identificador o llave primaria de la tabla	Simple monovalorado	Int	3	No	Los números enteros
Anual_id_anual	Es la llave foránea que resulta de la relación que hay con la tabla anual	Simple monovalorado	Int	3	No	Los números enteros
Periodo_int	Es el mes y el año en que se mostrara la información	Simple monovalorado	Varchar	9	No	Se compone de letras y números
Indice_total	Es el índice total del mes	Simple monovalorado	Flota	10	No	Números decimales
p_indices.	representa los principales índices de los que se compone el inpc.	Fuerte				
ID_inp	Es el identificador o llave primaria, se auto incrementa	Simple monovalorado	Int	3	No	Números enteros
ind_tot_ID	Es la llave foránea que resulta de la relación con pib_tot	Simple monovalorado	Int	3	No	Números enteros



Periodo_in	Las 3 primeras letras de los meses más en año. Se presenta trimestralmente	Simple monovalorado	Float	9	No	Números y letras
índice	Se refiere al índice de precios al consumidor	Simple monovalorado	Float	10	No	Números decimales
Subyacente	Se refiere al índice subyacente	Simple monovalorado	Float	10	No	Números decimales
Canasta	Se refiere al índice de canasta básica	Simple monovalorado	Float	10	No	Números decimales
Mercancías	Se refiere al índice de mercancías	Simple monovalorado	Float	10	No	Números decimales
origen_bienes.	representa el inpc por origen de los bienes.	Fuerte				
ID_ob	Es el identificador o llave primaria, se auto incrementa	Simple monovalorado	Int	3	No	Números enteros
ind_tot_ID	Es la llave foránea que resulta de la relación con pib_tot	Simple monovalorado	Int	3	No	Números enteros
Periodo_ob	Las 3 primeras letras de los meses más en año. Se presenta trimestralmente	Simple monovalorado	Float	9	No	Números y letras
Primario	Se refiere al sector primario	Simple monovalorado	Float	10	No	Números decimales
Secundario	Se refiere al sector secundario	Simple monovalorado	Float	10	No	Números decimales
Terciario	Se refiere al sector terciario	Simple monovalorado	Float	10	No	Números decimales
durab_bienes.	representa el inpc por durabilidad de los bienes.	Fuerte				
ID_db	Es el identificador o llave primaria, se auto incrementa	Simple monovalorado	Int	3	No	Números enteros
ind_tot_ID	Es la llave foránea que resulta de la relación con pib_tot	Simple monovalorado	Int	3	No	Números enteros
Periodo_ob	Las 3 primeras letras de los meses más en año. Se presenta trimestralmente	Simple monovalorado	Float	9	No	Números y letras
No_durables	Se refiere a los bienes No durables	Simple monovalorado	Float	10	No	Números decimales
Durables	Se refiere al los bienes durables	Simple monovalorado	Float	10	No	Números decimales
Servicios	Se refiere a los bienes de servios	Simple monovalorado	Float	10	No	Números decimales
Mercancías	Se refiere a las mercancías excluyendo petróleo	Simple monovalorado	Float	10	No	Números decimales
Remesas.	Esta tabla representa las remesas	Fuerte				
ID_re	Es el identificador o llave primaria de la tabla	Simple monovalorado	Int	3	No	Los números enteros
Anual_Id_anual	Es la llave foránea que resulta de la relación que hay con la tabla anual	Simple monovalorado	Int	3	No	Los números enteros
Periodo_r	Es el mes y el año en que se mostrara la información	Simple monovalorado	Varchar	9	No	Se compone de letras y números
Orden_dinero	Se refiere a el orden de dinero	Simple monovalorado	Float	10	No	Números decimales
Cheques	Se refiere a cheque personales	Simple monovalorado	Float	10	No	Números decimales
Transferencias	Se refiere a transferencias electrónicas	Simple monovalorado	Float	10	No	Números decimales
Efectivo	Se refiere a las remesas en efectivo	Simple monovalorado	Float	10	No	Números decimales
Total	Es el total de remesas	Simple monovalorado	Float	10	No	Números decimales
ND_r	Es el número diferenciador de las remesas, pueden ser 3 opciones. 1=remesas en millones de dólares	Simple monovalorado	Int	1	No	Los números enteros 1, 2 o 3.

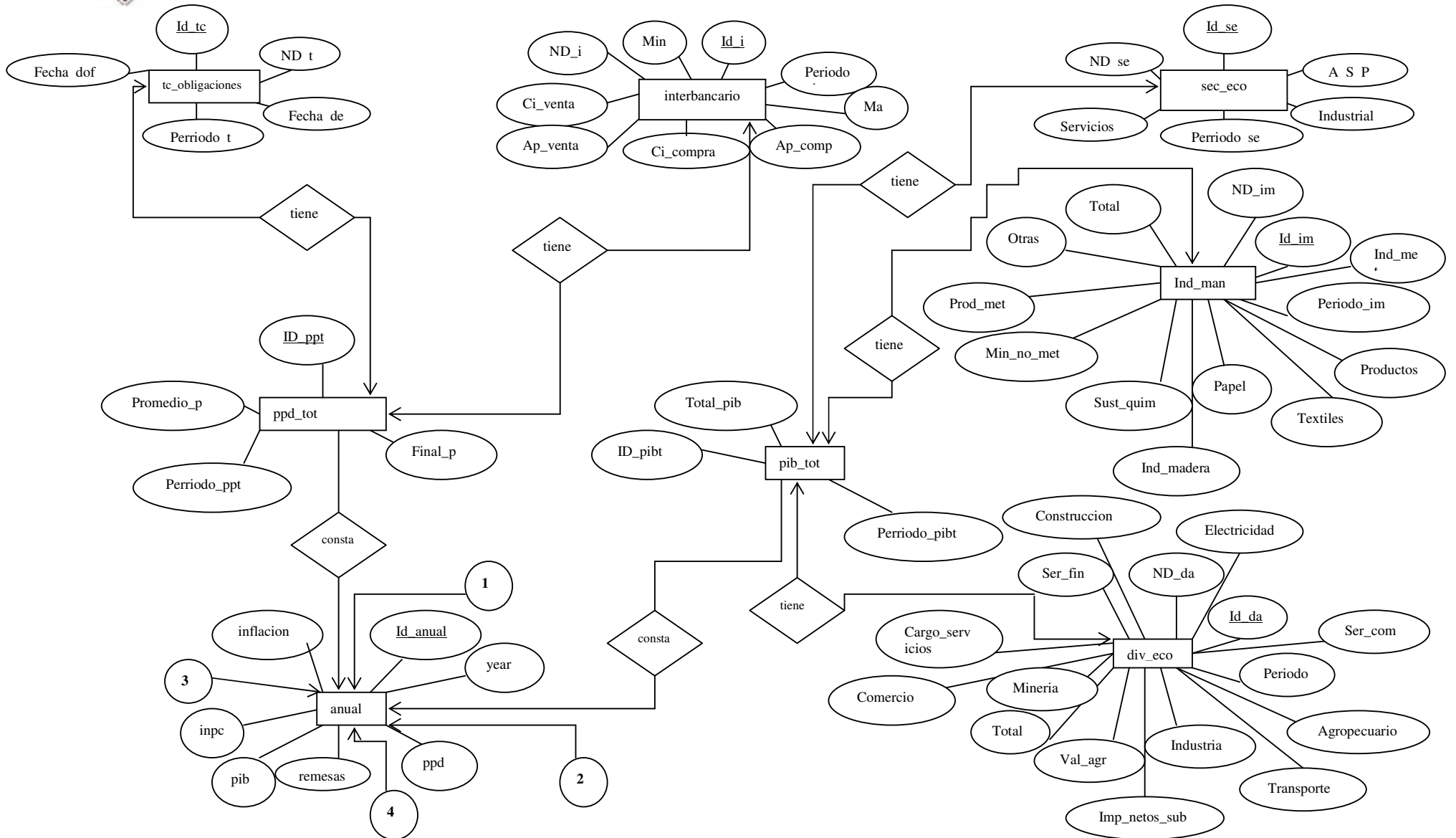


	2=remesas en miles de operaciones 3=remesas en dólares					
inflación.	Esta tabla representa la inflación	Fuerte				
ID_in	Es el identificador o llave primaria de la tabla	Simple monovalorado	Int	4	No	Los números enteros
Anual_Id_anual	Es la llave foránea que resulta de la relación que hay con la tabla anual	Simple monovalorado	Int	3	No	Los números enteros
Periodo_in	Es el mes y el año en que se mostrara la información	Simple monovalorado	Varchar	9	No	Se compone de letras y números
Mensual	Se refiere a la inflación mensual	Simple monovalorado	Float	10	No	Números decimales
Acumulada	Se refiere a la inflación acumulada	Simple monovalorado	Float	10	No	Números decimales
Annual	Se refiere a la inflación anual	Simple monovalorado	Float	10	No	Números decimales
ND_in	Se refiere al numero diferenciador de inflación 1= inflación de Inpc 2= inflación subyacente 3= inflación No subyacente 4=inflación de la canasta básica 5= inflación del inpp excluyendo petróleo	Simple monovalorado	Float	10	No	Números decimales
trimestral	En esta parte contiene las tendencias, y las predicciones de los indicadores	Fuerte				
Id_tri	Es el identificador de la tabla	Simple monovalorado	Int	1	No	Los números enteros
rem_tri	Remesas presentadas trimestralmente	Simple monovalorado	flotante	10	No	Números decimales
ten_rem	Tendencias de las remesas	Simple monovalorado	Flotante	10	No	Números decimales
par_tri	Paridad peso-dolar representadas trimestralmente	Simple monovalorado	Flotante	10	No	Números decimales
ten_par	Tendencias de la paridad	Simple monovalorado	Flotante	10	No	Números decimales
inpc_tri	INPC representadas trimestralmente	Simple monovalorado	Flotante	10	No	Números decimales
ten_inpc	Tendencias del INPC	Simple monovalorado	Flotante	10	No	Números decimales
inf_tri	Inflación representadas trimestralmente	Simple monovalorado	Flotante	10	No	Números decimales
ten_inf	Tendencias de la inflación	Simple monovalorado	Flotante	10	No	Números decimales
pib_tri	PIB representada trimestralmente	Simple monovalorado	Double	15	No	Números decimales
ten_pib	Tendencia del PIB	Simple monovalorado	Double	15	No	Números decimales

Figura 3.1 Diccionario de datos

3.1.3 MODELO ENTIDAD-RELACION

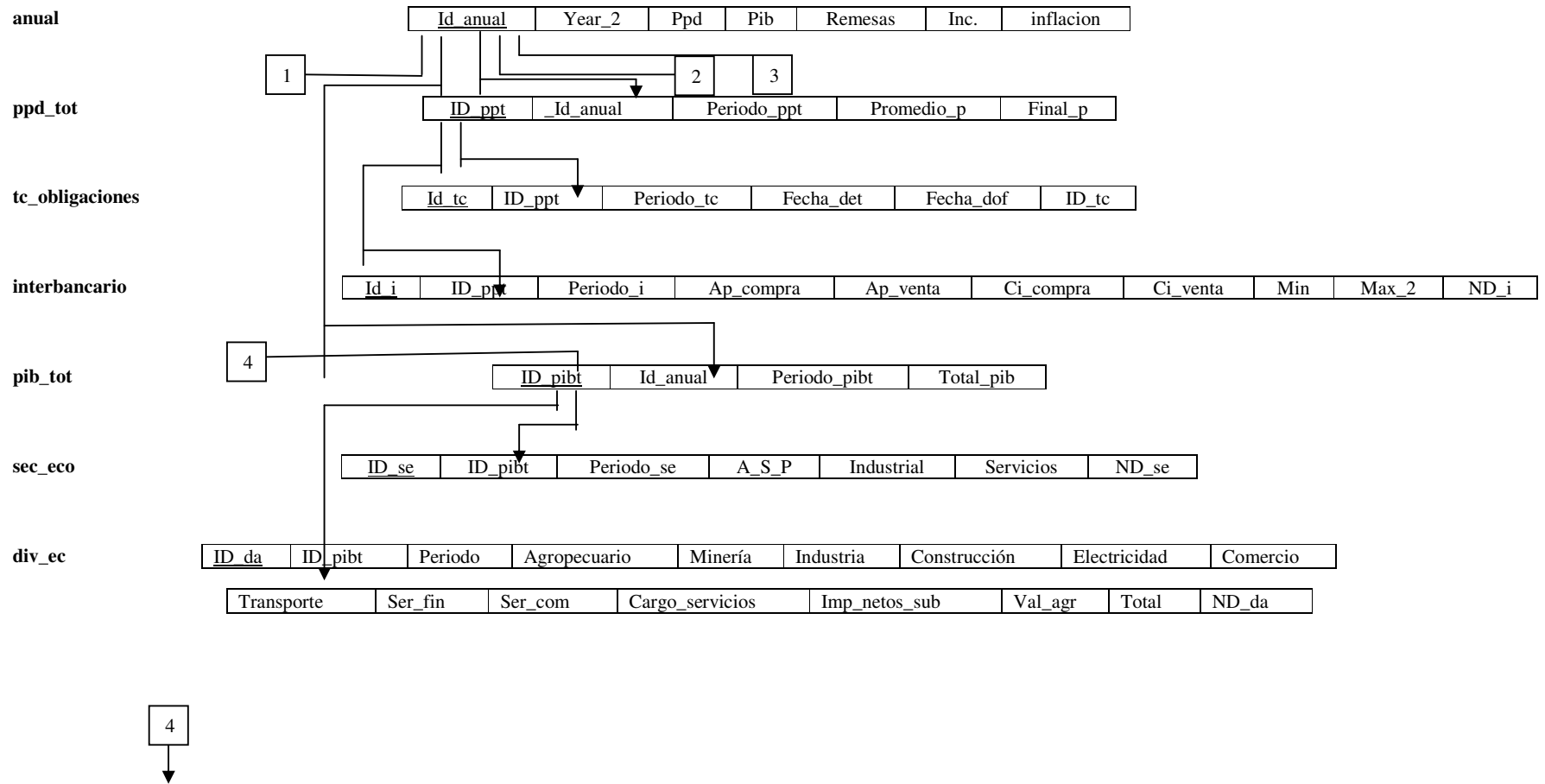
El modelo entidad-relación que se usara es el modelo conceptual más utilizado para el diseño conceptual de bases de datos. Fue introducido por Peter Chen en 1976. El modelo entidad-relación está formado por un conjunto de conceptos que permiten describir la realidad mediante un conjunto de representaciones gráficas como se muestra en la figura 2.8 que después de identificar las entidades, atributos y relaciones se procedió al modelo.





3.1.4 DISEÑO LOGICO

En esta etapa, se transforma el esquema conceptual en un esquema lógico como se muestra en la figura 2.9.



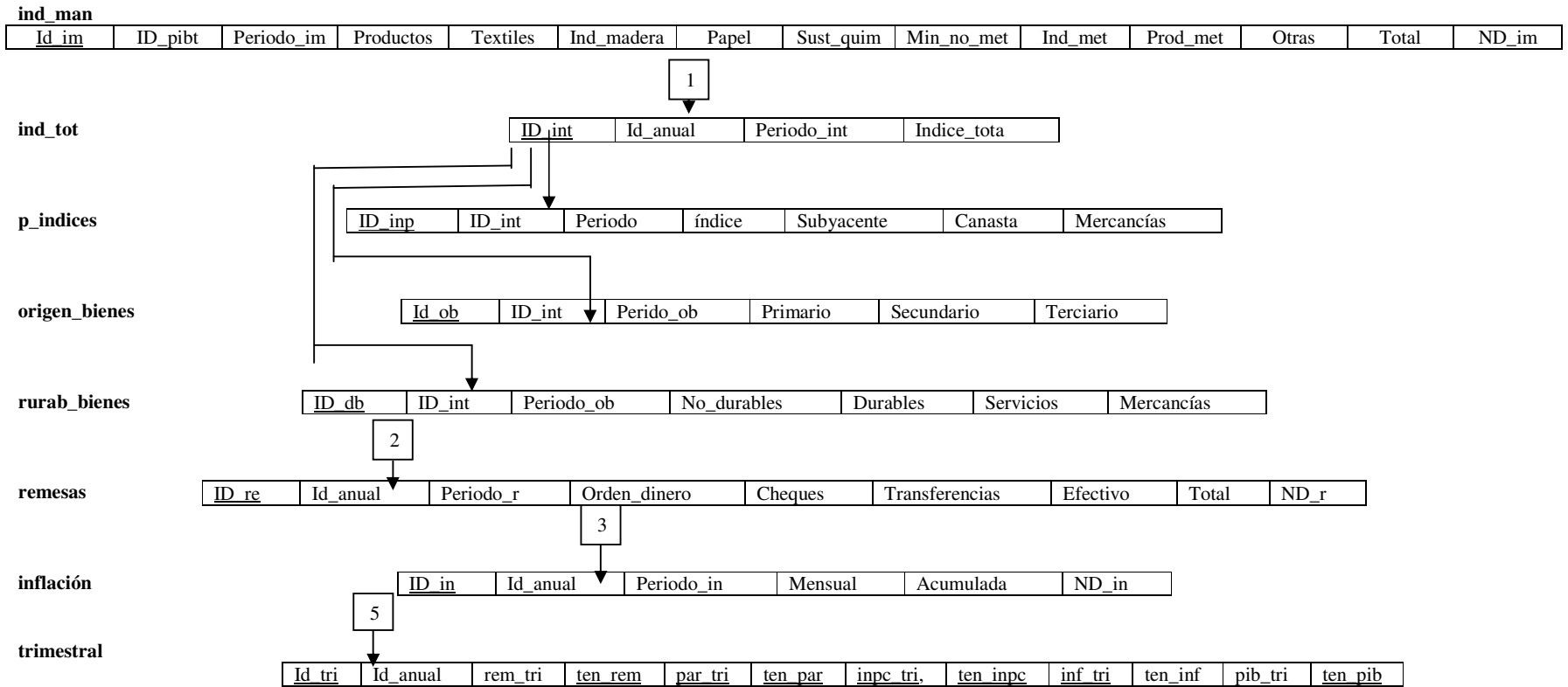


Figura 2.9 Modelo relacional



CAPÍTULO IV

IMPLEMENTACION

Y PRUEBAS



4.1 IMPLEMENTACION

La implementación es la programación del sistema, siguiendo el diseño elaborado anteriormente. Las tablas del modelo relacional se transforman en tablas de la base de datos, y de esta manera se permitirá agregar y manipular los datos.

En este capítulo se presenta la descripción de las principales funciones del sistema, así como el código. Además se muestra la interfaz de cada una de las pantallas principales del sistema.

4.2 HERRAMIENTAS DEL SISTEMA

La implementación del sistema se realizó de la siguiente manera: la base de datos se creó en MySQL, la programación de las interfaces se realizó en PHP utilizando el servidor APACHE, instalando el programa WampServer que contiene lo mencionado anteriormente y de esta manera nos permite crear aplicaciones para visualizar editar y actualizar información de la base de datos.

4.2.1 CONECTIVIDAD A LA BASE DE DATOS

MySQL tiene que conectarse al servidor, para esto, es necesario crear un archivo PHP llamado conexión que se muestra en la figura 4.1

```
<?php
function Conectarse()
{
    if (!($link=mysql_connect("localhost","root","")))
    {
        echo "Error conectando a la base de datos.";
        exit();
    }
    if (!mysql_select_db("financiera",$link))
    {
        echo "Error seleccionando la base de datos.";
        exit();
    }
    return $link;
}
?>
```

Figura 4.1 Archivo conexión.php

Para que MySQL pueda conectarse con el servidor se usa la función `mysql_conect()`:

```
if (!($link=mysql_connect("localhost","root","")))
echo "Error conectando a la base de datos.";
exit();
```

En donde `$link` es un punto de referencia para la nueva conexión que se creó. En lo que se refiere a los parámetros de la función `mysql_connect ()` se refiere a los usuarios y las contraseñas que en este caso no usamos en donde aparecen las comillas, y si no cumple con lo anterior muestra un mensaje de “Error conectando la base de datos” y sale del programa.



```
if (!mysql_select_db("financiera",$link))
{echo "Error seleccionando la base de datos.";
exit();}
```

En caso contrario, es decir que la conexión se haya realizado con éxito, se aplica la función `mysql_select_db ()`, los parámetros que usa es el nombre de la base de datos que se utiliza y la variable del punto de referencia, en caso de que no se encuentre dicha base de datos se muestra un mensaje de error y sale, en caso contrario la conexión es un éxito y devuelve un enlace que corresponde a la conexión abierta asignado a la variable `$link`.

El archivo llamado `conexion.php` es incluido en los archivos que necesiten conectarse a la base de datos y de esta manera acceder a los datos.

4.2.2 ALTAS EN LA BASE DE DATOS

En la figura 4.2 y 4.3 se muestra el código fuente de altas en un indicador que es tipo de cambio para solventar obligaciones, consta de dos archivos, el primer archivo es el formulario que se envía para que el siguiente archivo inserte los datos enviados en la base de datos.

```
<html>
<body>
<h2>INSERTA EN TIPO DE CAMBIO PARA SOLVENTAR OBLIGACIONES </h2>
<FORM ACTION="insertar.php">
<TABLE>
<TR>
<TD>Fecha:</TD>
<TD><INPUT TYPE="text" NAME="Periodo_tc" SIZE="20" MAXLENGTH="30"></TD>
</TR>
<TR>
<TD>Fecha de Determinacion:</TD>
<TD><INPUT TYPE="text" NAME="Fecha_det" SIZE="20" MAXLENGTH="30"></TD>
</TR>
<TR>
<TD>Fecha de DOF :</TD>
<TD><INPUT TYPE="text" NAME="Fecha_dof" SIZE="20" MAXLENGTH="30"></TD>
</TR>
<TR>
<TD>Fecha de DOF :</TD>
<TD><INPUT TYPE="text" NAME="Fecha_dof" SIZE="20" MAXLENGTH="30"></TD>
</TR>
<TR>
<TD>Numero Diferenciador:</TD>
<TD><INPUT TYPE="text" NAME="ND_tc" SIZE="20" MAXLENGTH="30"></TD>
</TR>
</TABLE>
<INPUT TYPE="submit" NAME="accion" VALUE="Grabar">
</FORM>
</body>
</html>
```

Figura 4.2 Archivo inserta.php



```
</HTML>
<BODY>
<p>
<?php
include("conexion.php");
$link=Conectarse();

$fecha=$_GET['Periodo_tc'];
$det=$_GET['Fecha_det'];
$DOF=$_GET['Fecha_dof'];
$ND=$_GET['ND_tc'];

if(($fecha == NULL)||($det == NULL)||($DOF == NULL)||($ND == NULL)) {print (" ALGUN
CAMPO SE ENCUENTRA VACIO <br>\n");}
else
if(ereg("[[:space:]]",$fecha)) {print (" NO ES NECESARIO EL ESPACIO ANTES <br>\n");}
else
if(!ereg("[A-Z][a-z]{2}[[:space:]][0-9]{4}",$fecha)) {print (" EL PERIODO ESTA MAL
ESCRITO.<BR>\n LA PRIMERA LETRA DEL MES ES MAYUSCULA Y LAS SIGUIENTES
MINUSCULAS<br>\n");}
else
if(!is_numeric(ereg_replace(",",".",$det))) { print (" El campo de fecha de determinacion no es
numerico <br>\n"); }
else
if(!is_numeric(ereg_replace(",",".",$DOF))) { print (" El campo de fecha de DOF no es
numerico <br>\n"); }
else
if(!is_numeric(ereg_replace(",",".",$ND))) { print (" El campo de numero diferenciador no es
numerico <br>\n"); }
else
if(!ereg("[1-2]",$ND)) { echo "Este campo numero diferenciador solo acepta los numeros 1 o
2";}
else
{
$result = mysql_query ("select Periodo_tc,ND_tc from tc_obligaciones where Periodo_tc
='$fecha' && ND_tc ='$ND'", $link);
$data = mysql_fetch_array($result);

if (($data['Periodo_tc'] != $fecha)&&($data['ND_tc'] != $ND))

{mysql_query("insert into tc_obligaciones (Periodo_tc, Fecha_det, Fecha_dof,ND_tc) values
('$fecha','$det','$DOF','$ND')", $link);
print ( "Datos insertados<br>\n");
print ("Periodo: $fecha <br>\n");
print ("Fecha de determinacion: $det <br>\n");
print ("Fecha del DOF: $DOF <br>\n");
print ("Numero Diferenciador: $ND <br>\n");}
else
{print ("EL DATO YA FUE INSERTADO Y NO SE PUEDE DUPLICAR");}}
?>
</p>
<p> <a href="inserta.php">INSERTAR MAS DATOS </a></p>
<p> <a href="paridada.php">IR AL MENU DE PARIDAD </a>
</p>
</BODY>
</HTML>
```

Figura 4.3 Archivo insertar.php



4.2.3 BAJAS EN LA BASE DE DATOS

En la figura 4.4 y 4.5 se muestra el código fuente de bajas en un indicador que es tipo de cambio para solventar obligaciones, consta de dos archivos, el primer archivo envía el id del dato que se quiere borrar y el siguiente archivo borra los datos del id enviado en la base de datos.

```
</html>
<body>
<h2>BAJA EN TIPO DE CAMBIO PARA SOLVENTAR OBLIGACIONES </h2>
<?php
include("conexion.php");
$link=Conectarse();
$result=mysql_query("select * from tc_obligaciones",$link);
?>
<TABLE BORDER=1 align="center" CELLPADDING=1 CELLSPACING=1>
<TR><TD>&nbsp;<B>ID</B></TD> <TD>&nbsp;<B>Periodo</B>&nbsp;</TD>
<TD>&nbsp;<B>Eliminar</B>&nbsp;</TD></TR>
<?php

while($row = mysql_fetch_array($result)) {
printf("<tr><td>&nbsp;<td>&nbsp;<td>&nbsp;<td><a
href='borrar.php?ID_tc=%s'>Eliminar</a></td></tr>",
$row["Id_tc"],$row["Periodo_tc"],$row["Id_tc"]);
}
mysql_free_result($result);
mysql_close($link);
?>
</table>
</body>
</html>
```

Figura 4.4 Archivo borra.php

```
</HTML>
<BODY>
<?php
include("conexion.php");
$link=Conectarse();
$id=$_GET['ID_tc'];

mysql_query("delete from tc_obligaciones where ID_tc = '$id',$link);

print ( "SE ELIMINARON LOS DATOS SATISFACTORIAMENTE <br>\n");
?>
<p> <a href="borra.php">ELIMINAR MAS DATOS</a></p>
<p> <a href="paridada.php">IR AL MENU DE PARIDAD </a></p>

</HTML>
</BODY>
```

Figura 4.5 Archivo borrar.php



4.2.4 ACTUALIZACION EN LA BASE DE DATOS

En la figura 4.6, 4.7 y 4.8 se muestra el código fuente de actualización o el reemplazo de un indicador que es tipo de cambio para solventar obligaciones, consta de dos archivos, el primer archivo se divide en dos partes, la primera se envía el período a actualizar, y en la segunda se despliega un formulario con los datos del periodo escrito para que el usuario escriba los datos que va a reemplazar; y el segundo archivo recibe los datos y los actualiza.

```
<HTML>
</head>
<body>
<h2 align="center">ACTUALIZACION</h2>
<form id="FORM1" name="FORM2">
<FORM1 METHOD="POST" ACTION="$PHP_SELF">
<TABLE>
<TR>
<TD>CUAL ES EL PERIODO QUE QUIERES ACTUALIZAR</TD>
<TD><INPUT TYPE="text" NAME="Periodo" SIZE="20" MAXLENGTH="30"></TD>
<TD>INSERTAR EL NUMERO DIFERENCIADOR</TD>
<TD><INPUT TYPE="text" NAME="ND_tc" SIZE="20" MAXLENGTH="30"
VALUE="0"></TD>
</TR>
</TABLE>
<div align="left">
<INPUT TYPE="submit" NAME="accion" VALUE="Enviar">
</div>
</FORM>

<hr>
<?php
include("conexion.php");
$link=Conectarse();

$fecha=$_GET['Periodo'];
$ND=$_GET['ND_tc'];

$result = mysql_query ("select * from tc_obligaciones where Periodo_tc ='$fecha' &&
ND_tc ='$ND'", $link);
$data = mysql_fetch_array($result);

if (NULL == $data['Id_tc'])
{echo "LOS DATOS INSERTADOS NO SON VALIDOS";}
else
{
?>
<form id="FORM2" name="FORM2" METHOD="POST" ACTION="actu.php">

<TABLE>
<TR>
<TD><INPUT NAME="Id_tc" type="hidden" VALUE="<?php echo $data['Id_tc'] ?>">
</TR>
```

Figura 4.6 Archivo act.php (Parte 1 de 2)



```
<TR>
  <TD>Fecha de Determinacion:</TD>
  <TD><INPUT TYPE="text" NAME="Fecha_det" SIZE="20" MAXLENGTH="30"
VALUE="<?php echo $data['Fecha_det'] ?>"></TD>
</TR>
<TR>
  <TD>Fecha de DOF :</TD>
  <TD><INPUT TYPE="text" NAME="Fecha_dof" SIZE="20" MAXLENGTH="30"
VALUE="<?php echo $data['Fecha_dof'] ?>"></TD>
</TR>
</TABLE>

<div align="left">
  <INPUT TYPE="submit" NAME="accion" VALUE="Enviar">

  <input type="reset" name="Submit" value="Limpiar">
</div>
<?php } ?>
</FORM>
```

Figura 4.7 Archivo act.php (Parte 2 de 2)

```
<HTML>
<HEAD>
<TITLE>Actualiza.php</TITLE>
</HEAD>
<BODY>
<?php
  include("conexion.php");
  $link=Conectarse();
  $num=$_POST['Id_tc'];
  $na=$_POST['Fecha_det'];
  $nb=$_POST['Fecha_dof'];

  if(!is_numeric(ereg_replace(",",".", $na))) { print (" EL CAMPO DE FECHA DE
DETERMINACION NO ES NUMERICO<br>\n"); }
  else
    if(!is_numeric(ereg_replace(",",".", $nb))) { print (" EL CAMPO DE FECHA DE DOF NO ES
NUMERICO<br>\n"); }
  else{
    $result = mysql_query ("select * from tc_obligaciones where Id_tc ='$num'", $link);
    $data = mysql_fetch_array($result);

    $a=$data['Fecha_det'];
    $b=$data['Fecha_dof'];

    $sql="UPDATE tc_obligaciones SET Fecha_det='$na', Fecha_dof='$nb' WHERE
Id_tc='$num'";
    mysql_query($sql,$link);

    print ( "LAS ACTUALIZACIONES SE HICIERON CORRECTAMENTE<br>\n<br>\n");
    print("\nSe actualizo $a por $na <br>\n");
    print("Se actualizo $b por $nb <br>\n");}
?>
<p> <a href="act.php">ACTUALIZAR MAS DATOS</a></p>
<p> <a href="paridada.php">IR AL MENU DE PARIDAD </a></p>
```

Figura 4.8 Archivo actu.php



4.2.5 CONSULTAS EN LA BASE DE DATOS

En la figura 4.9 y 4.10 se muestra el código fuente de consultas en un indicador que es tipo de cambio para solventar obligaciones, consta de dos partes, la primera parte consta de los formularios en menú de mes y año, que son las opciones de consultar, la segunda parte es donde se muestran los datos que se consultaron.

```
<html>
<body>
<h4 align="center">TIPO DE CAMBIO PARA SOLVENTAR OBLIGACIONES
EXTRANJERAS</h4>
<h4 align="center"><strong>PROMEDIO MENSUAL</strong></h4>
<h4>Buscar por </h4>
<form name="form1" method="post" action="">
<p>
<label>MES
<select name="select">
<option value="Tod">Todo</option>
<option value="Ene">Enero</option>
<option value="Feb">Febrero</option>
<option value="Mar">Marzo</option>
<option value="Abr">Abril</option>
<option value="May">Mayo</option>
<option value="Jun">Junio</option>
<option value="Jul">Julio</option>
<option value="Ago">Agosto</option>
<option value="Sep">Septiembre</option>
<option value="Oct">Octubre</option>
<option value="Nov">Noviembre</option>
<option value="Dic">Diciembre</option>
</select>
</label>
<label>
<input type="submit" name="Submit" value="Enviar">
</label>
</form>
<form name="form2" method="post" action="">
<label>A&Ntilde;O
<select name="select2">
<option value="Tod">Todo</option>
<option value="92">1992</option>
<option value="93">1993</option>
<option value="94">1994</option>
<option value="95">1995</option>
<option value="96">1996</option>
<option value="97">1997</option>
<option value="98">1998</option>
<option value="99">1999</option>
<option value="00">2000</option>
<option value="01">2001</option>
<option value="02">2002</option>
<option value="03">2003</option>
<option value="04">2004</option>
<option value="05">2005</option>
<option value="06">2006</option>
<option value="07">2007</option>
</select>
```

Figura 4.9 Archivo tpsop.php (Parte 1 de 2)



```
</label>
<label>
<input type="submit" name="Submit2" value="Enviar">
</label>
</form>
<hr>
<p>
<?PHP
  $a = $_REQUEST['select'];
  $b = $_REQUEST['select2'];
?>
<?php
include("conexion.php");
$link=Conectarse();

  if($a=='Tod' || $b=='Tod' || $a==' ')
    $result=mysql_query("select * from tc_obligaciones where ND_tc='1'", $link);
  else

if($a=='Ene' || $a=='Feb' || $a=='Mar' || $a=='Abr' || $a=='May' || $a=='Jun' || $a=='Jul' || $a=='Ago' || $a=='
Sep' || $a=='Oct' || $a=='Nov' || $a=='Dic')
  $result=mysql_query("select * from tc_obligaciones where Periodo_tc like '$a%'&&
ND_tc='1'", $link);
  else
    if($b==' 92' || '93' || '94' || '95' || '96' || '97' || '98' || '99' || '00' || '01' || '02' || '03' || '04' || '05' || '06' || '07')
      $result=mysql_query("select * from tc_obligaciones where Periodo_tc like '%" . $b . "'&&
ND_tc='1'", $link);
?>
</p>
<div align="center">
<TABLE BORDER=1 CELSPACING=1 CELLPADDING=1>
<TR>
<TD>&nbsp;Periodo</TD>
<TD>&nbsp;Fecha de determinacion&nbsp;</TD>
<TD>&nbsp;Fecha del DOF&nbsp;</TD>
</TR>
<?php

while($row = mysql_fetch_array($result)) {
  printf("<tr><td>&nbsp;%s</td><td>&nbsp;%s&nbsp;</td><td>&nbsp;%s&nbsp;</td></tr>", $row["Periodo_tc"], $row["Fecha_det"], $row["Fecha_dof"]);
}
mysql_free_result($result);
mysql_close($link);
?>
</table>
</div>
<blockquote>
  <blockquote>
    <p>&nbsp;</p>
  </blockquote>
</blockquote>
</body>
</html>
```

Figura 4.10 Archivo tcpsop.php (Parte 2 de 2)



4.2.6 GENERACIÓN DE GRÁFICOS

En la figura 4.11 y 4.12 se muestra el código fuente de la generación de los gráficos. Esto se hace por medio de la programación extrayendo los datos de una tabla de la base de datos..

```
<?php
// conectamos a la base de datos
include("conexion.php");
$link=Conectarse();
$result=mysql_query("select ten_pib from trimestral where id_tri<= 24",$link);
$result1=mysql_query("select pib_tri from trimestral where id_tri<= 24",$link);

$data = array(); // creamos un arreglo en blanco, este será el de los datos
$data1 = array();
$line = mysql_fetch_array($result);
$line1 = mysql_fetch_array($result1);

while ($line1 = mysql_fetch_array($result1)) {
    $data1[] = $line1[0];
}
while ($line = mysql_fetch_array($result)) {
    $data[] = $line[0]; // agregamos el dato, suponiendo que este en la primera posición del
arreglo resultante
}
include ("C:/wamp/www/jpgraph-2.2/src/jpgraph.php");
include ("C:/wamp/www/jpgraph-2.2/src/jpgraph_line.php");

// aquí ya tenemos el arreglo en $data, solo le agregamos al código anterior
$datay = $data;
$datay1 = $data1;
// Crea el grafico
$graph = new Graph(570,350,"auto");
$graph->img->SetMargin (25,20,20,30);

//$graph->SetBackgroundImage("tiger_bkg.png",BGIMG_FILLFRAME);

$graph->img->SetAntiAliasing("white");
$graph->SetScale("textlin");
$graph->SetShadow();
$graph->title->Set("PRODUCTO INTERNO BRUTO");

$graph->ygrid->Show(true,false);
$graph->xgrid->Show(true,true);

// Use built in font
$graph->title->SetFont(FF_FONT1,FS_BOLD);

// Slightly adjust the legend from it's default position in the
// top right corner.
$graph->legend->Pos(0.05,0.5,"right","center");

// Create the first line
$p1 = new LinePlot($datay);
$p2 = new LinePlot($datay1);
```

Figura 4.11 Archivo tpib.php (Parte 1 de 2)



```
//$p1->mark->SetType(MARK_FILLEDIRCLE);
//$p1->mark->SetFillColor("red");
//$p1->mark->SetWidth(4);
$p1->SetColor("blue");
$p1->SetCenter();
$p2->SetColor("red");
$p2->SetCenter();
$p1->SetLegend("TENDENCIA");// aqui va el nombre de la señal...
$p2->SetLegend("GRAFICO");
$graph->Add($p1);
$graph->Add($p2);
$graph->xaxis->SetTickLabels($datax);
$graph->xaxis->SetTextTickInterval(1);
// Output line
$graph->Stroke();
?>
```

Figura 4.12 Archivo tpiib.php (Parte 2 de 2)

4.3 MODULOS DEL SISTEMA

1. **ALTA DE INDICADORES.** En la pantalla se muestra el formulario y se agregan los datos a insertar, si todos los campos están llenos y no hay error al insertar los números ni el período del año, se muestra un mensaje en donde dice “DATOS INSERTADOS” y se puede rectificar que los datos que insertamos fueron los correctos al visualizar el mensaje en donde muestra los datos insertados en cada campo, en caso de que exista algún error no permite insertar los datos erróneos y muestra un mensaje que dice “LOS DATOS NO SON NUMERICOS”
2. **BAJA DE INDICADORES.** En la primera pantalla se visualiza el período y el ID del indicador con las ligas Eliminar, en caso de que existan dos meses iguales se muestra el número diferenciador, en el período que se quiera borrar se la da clic en la liga eliminar y en seguida se despliega un mensaje que dice “LOS DATOS FUERON ELIMINADOS CORRECTAMENTE”.
3. **ACTUALIZACIÓN DE INDICADORES.** Este módulo es para reemplazar los datos si es que existen cambios. La primera pantalla se divide en dos partes, lo primero que se visualiza es un mensaje que dice “INSERTE EL PERIODO QUE QUIERE ACTUALIZAR”, al escribir el período, se despliega un formulario con los datos que se quieren reemplazar, al escribir los datos nuevos se envían y si no existe error en éstos, se actualizan y se rectifican al visualizar un mensaje de que los datos fueron actualizados.
4. **CONSULTA DE INDICADORES.** En este módulo, el usuario tiene la opción de seleccionar la búsqueda entre dos opciones, por mes o por año.
5. **CONSULTA DE TENDENCIAS.** Dependiendo del indicador que se quiera consultar, se muestran las tendencias con opción de ver la grafica del indicador o consultarlas en números.
6. **CONSULTA DE PREDICCIONES.** En este módulo se muestra el gráfico de las predicciones del indicador a consultar.



4.4 INTERFACES

La interfaz inicial se realizó mediante tres marcos que a continuación se describen:

1. Marco superior. Es el que contiene el título “ Sistema financiero.”
2. Marco principal. En donde como primera interfaz se visualiza un signo de pesos simbolizando que el sistema se relaciona con la economía y abajo un formulario muy pequeño en donde solicita el usuario y la contraseña del administrador, si son correctos los datos, accedemos a una página igual que la de usuarios pero en vez de que nos muestren las consultas o los datos , nos muestran las opciones de altas y bajas según la conveniencia del usuario. En caso de que los datos sean incorrectos despliega un mensaje de Error.
3. Marco izquierdo otro en donde se visualizan los botones de los principales indicadores, cada indicador se divide en otros indicadores y es por esto que al ir a un botón, en la página principal se despliegan las ligas a los datos de los indicadores.

En la figura 4.13 se muestra el archivo del código de la interfaz inicial del Sistema Financiero y en la figura 4.14 la interfaz del Sistema

De la figura 4.15 a la 4.19 se muestran las interfaces de los botones de los indicadores.

En la figura 4.20 se muestra la interfaz del botón tendencias, la liga datos se refiere a los tendencias mostradas en números, la liga gráfico en la gráfica de tendencias y la liga predicción en la gráfica de predicciones, y al darle clic en todas las gráficas se despliegan las graficas de tendencias de todos los indicadores.

```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>SISTEMA FINANCIERO</title>
</head>

<frameset rows="" cols="185,*" frameborder="yes" border="0" framespacing="0">
<frame src="4.php" name="leftFrame" scrolling="No" noresize="noresize"
id="leftFrame" />
<frameset rows="58,*" cols="" framespacing="0" frameborder="yes" border="0">
<frame src="3.php" name="topFrame" scrolling="No" noresize="noresize"
id="topFrame" />
<frame src="2.php" name="mainFrame" id="mainFrame" />
</frameset>
</frameset>
<noframes><body>
</body>
</noframes></html>
```

Figura 4.13 Archivo 1.php



Figura 4.14 Interfaz inicial

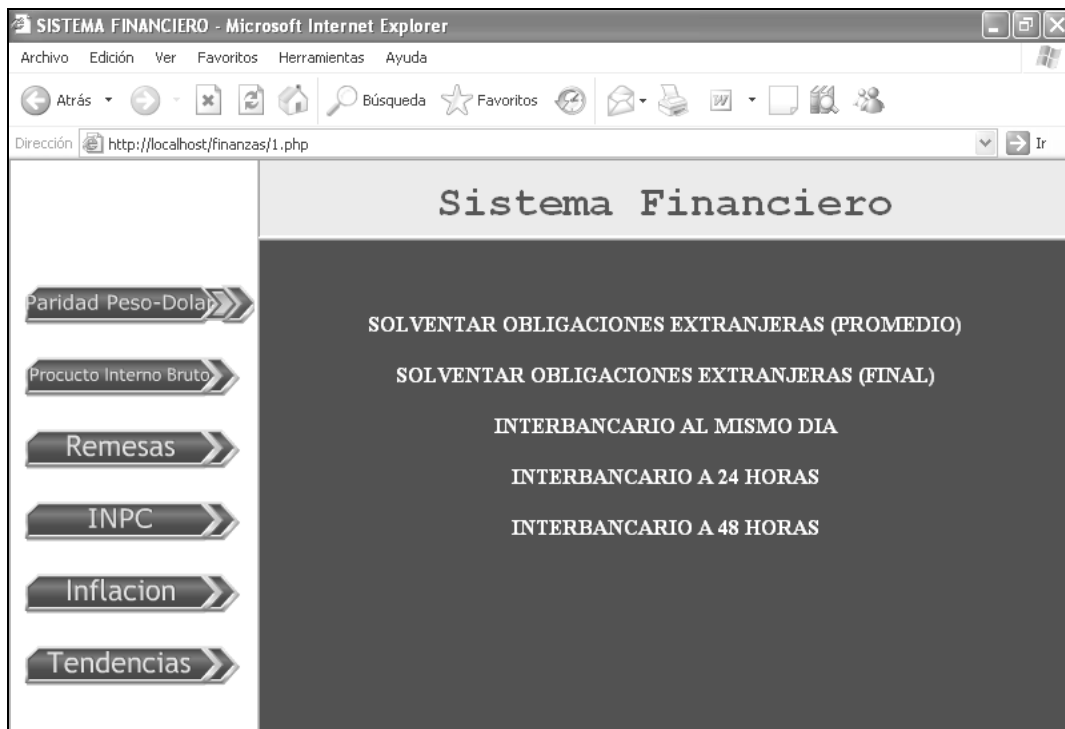


Figura 4.15 Interfaz de paridad peso dólar.

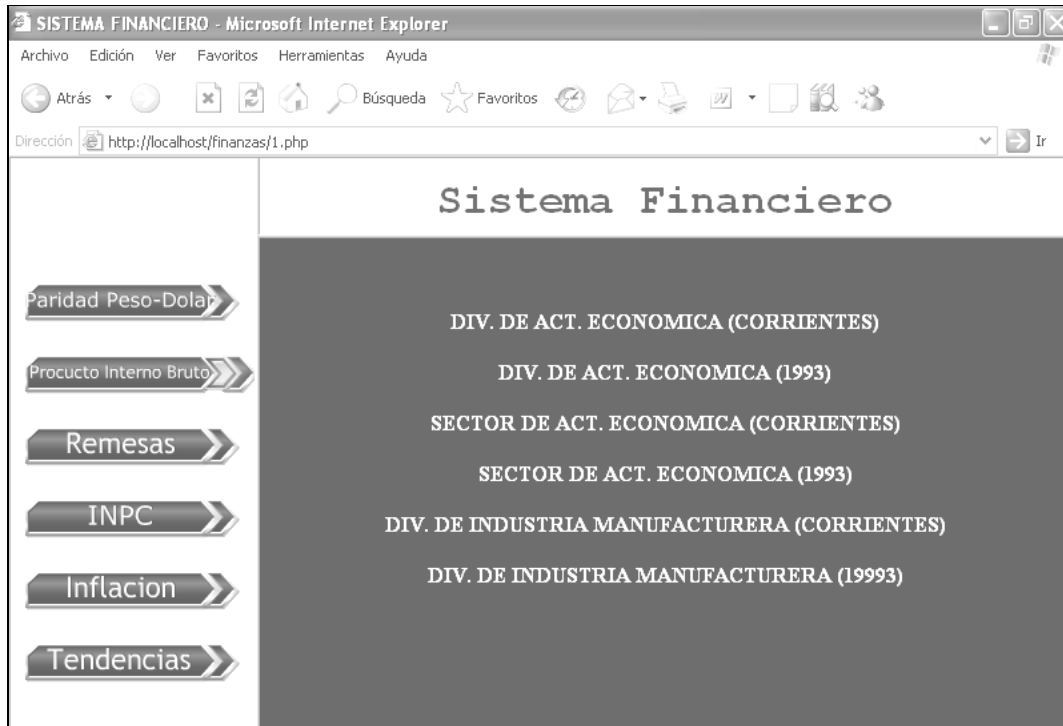


Figura 4.16 Interfaz de producto interno bruto

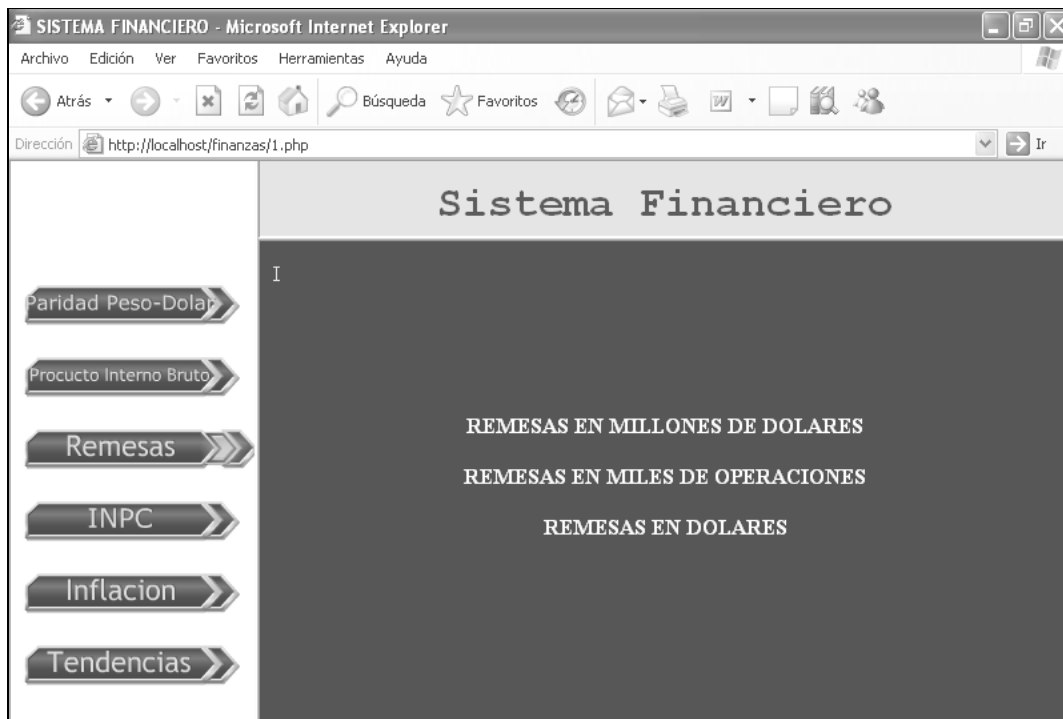


Figura 4.17 Interfaz de remesas

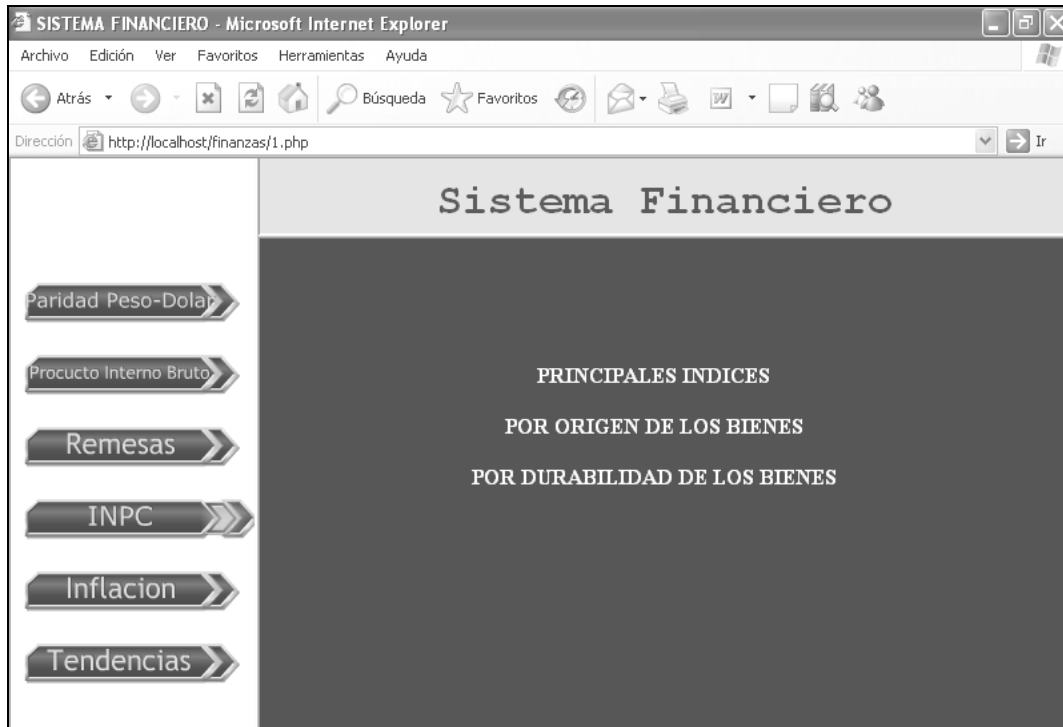


Figura 4.18 Interfaz del INPC



Figura 4.19 Interfaz de Inflación



Figura 4.20 Interfaz de Tendencias

De la figura 4.21 a la figura 4.26 se muestran las interfaces que usarán los administradores cuando quieran actualizar la base de datos.

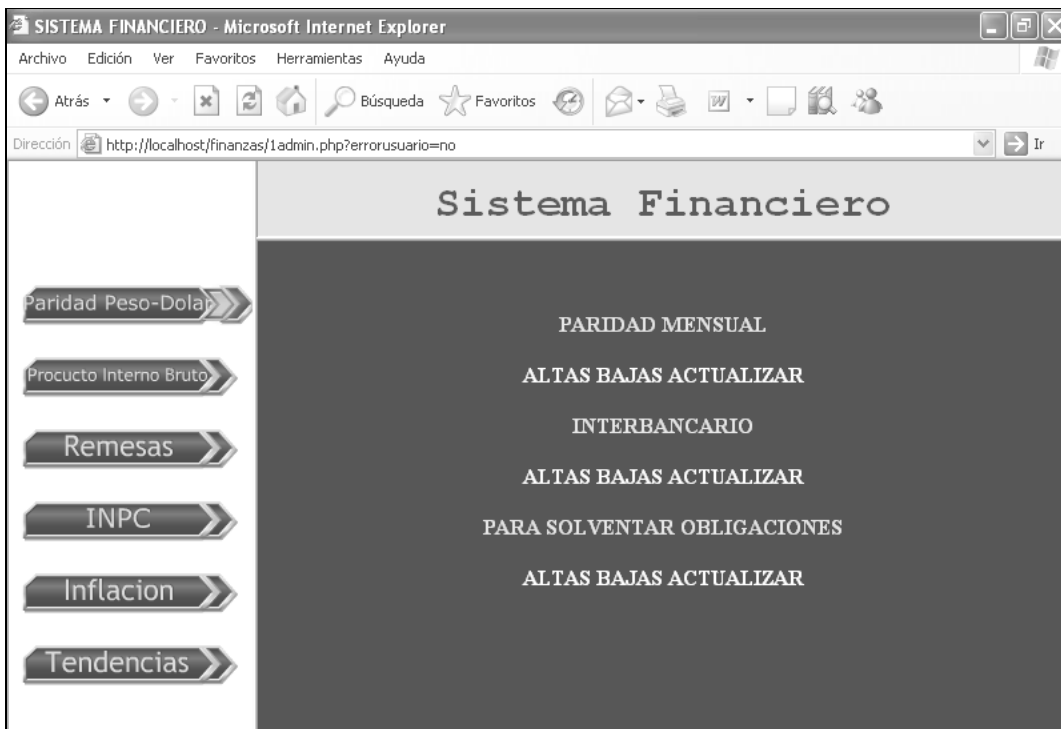




Figura 4.21 Interfaz de modificación de Paridad Peso-Dólar



Figura 4.22 Interfaz de modificación de Producto Interno Bruto



Figura 4.23 Interfaz de modificación de Remesas

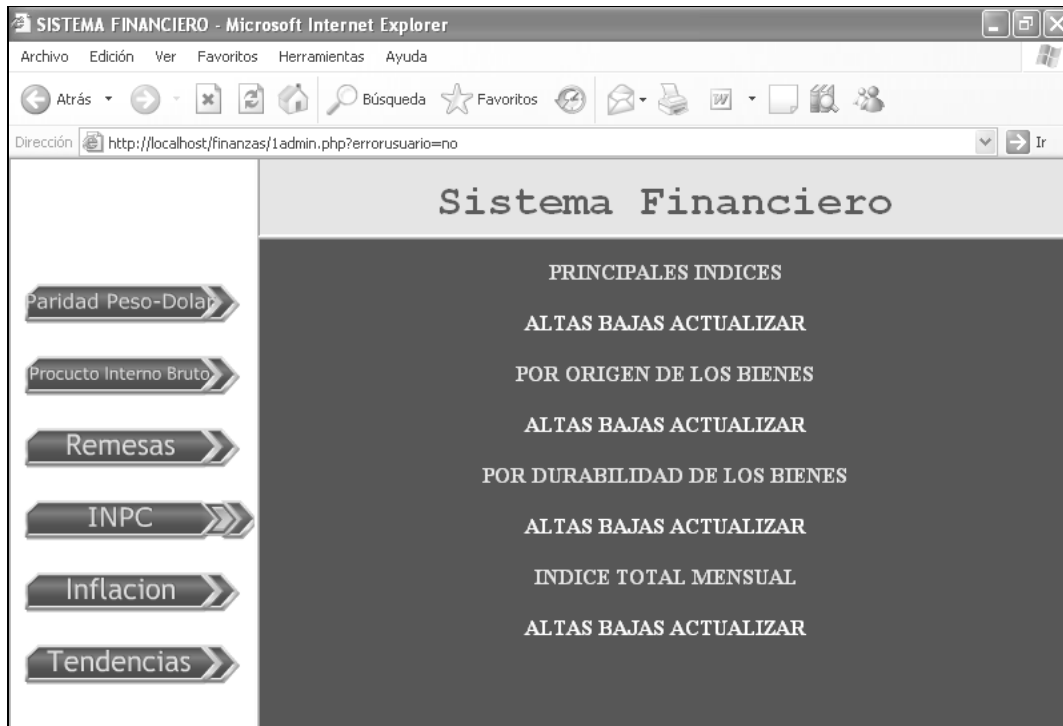


Figura 4.24 Interfaz de modificación de Índice Nacional de Precios al Consumidor



Figura 4.25 Interfaz de modificación de Inflación



Figura 4.26 Interfaz de modificación de Inflación

4.5 PRUEBAS

Después de implementar el sistema con los requerimientos planteados es necesario validarlo para tener la seguridad de que el Sistema funciona correctamente; para ello es necesario hacer las pruebas. A continuación se muestran las pruebas que se realizaron para verificar el funcionamiento.

En la figura 4.27 se muestra una prueba realizada para dar de alta los datos de un indicador. Al escribir datos que no sean números se visualiza un error como se muestra en la figura 4.28, al igual que si los espacios están vacíos como se muestra en la figura 4.29. Además si el usuario no escribe correctamente el período se visualiza un mensaje que ayuda a que el usuario corrija la forma de escribirlo como se muestra en la figura 4.30. Una vez que el administrador captura los datos correctamente le da clic en el botón grabar, y en seguida se muestra un mensaje de que los datos fueron insertados y también los datos insertados como se muestra en la figura 4.31.

Si el administrador quiere borrar todos los datos, le tiene que dar clic en limpiar y los datos que se escribieron se borran



Sistema Financiero

ALTA EN TIPO DE CAMBIO INTERBANCARIO

Periodo:	Jun 2008
Apertura a la compra:	12.36
Apertura a la venta:	12.38
Cierre a la compra:	11.23
Cierre a la venta:	11.23
Maximo:	12.365
Minimo:	11.14
Numero Diferenciador:	1

Figura 4.27 Alta de los indicadores (formulario)

Sistema Financiero

EL CAMPO DE APERTURA A LA COMPRA NO ES NUMERICO

[INSERTAR MAS DATOS](#)

[IR AL MENU DE PARIDAD](#)

Figura 4.28 Mensaje de error en los datos

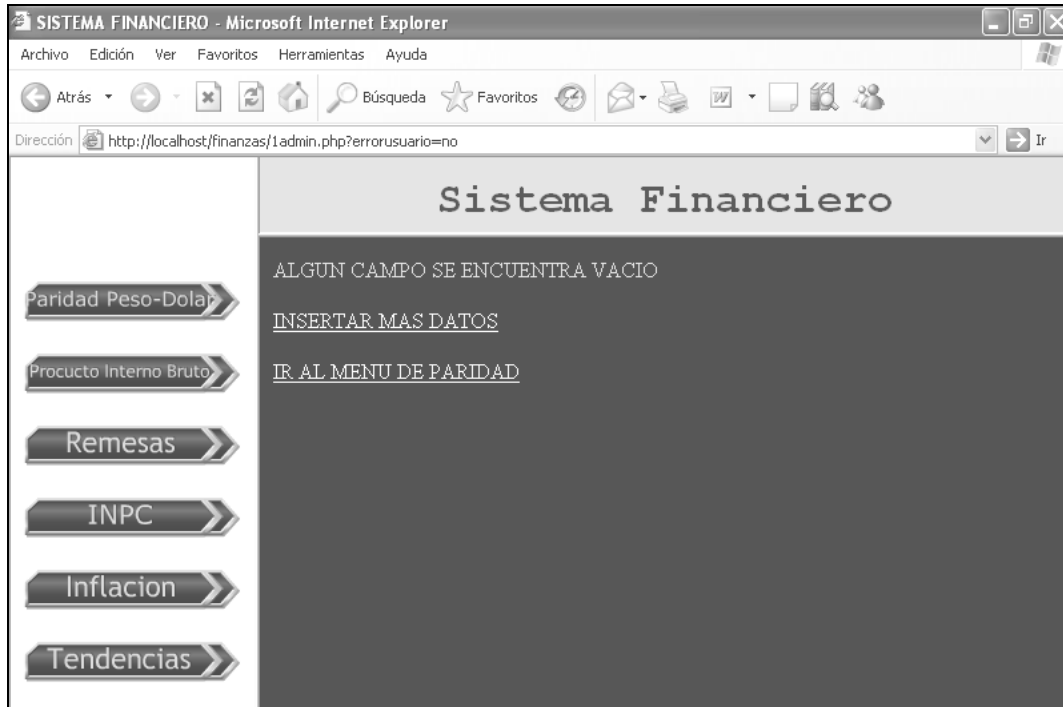


Figura 4.29 Mensaje de error de campo vacío

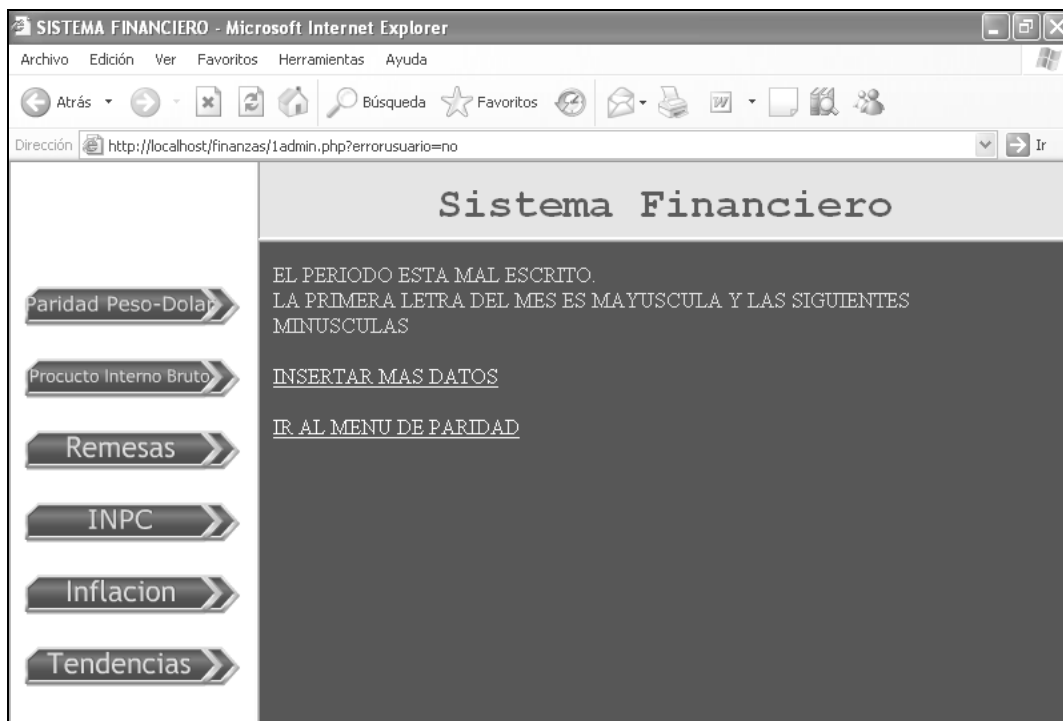


Figura 4.30 Mensaje de error en el campo período

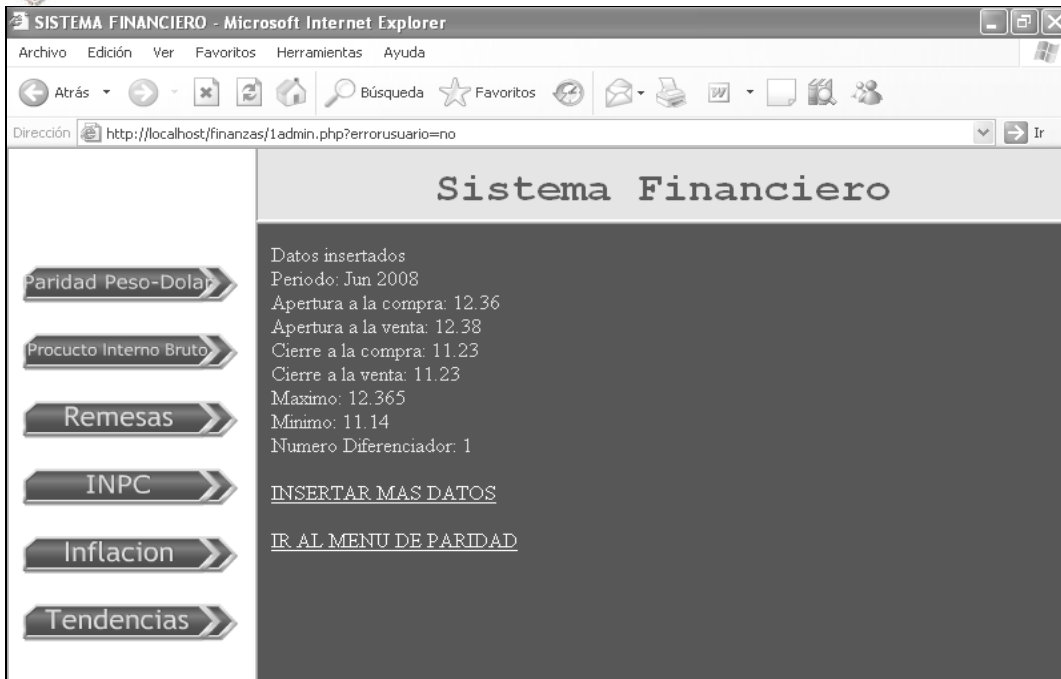


Figura 4.31 Alta de los indicadores (dato insertado)

En la figura 4.32 se muestra la prueba realizada para dar de baja los datos de algún mes, para dar de baja se tiene que dar clic en Eliminar en la fila en donde aparezca el mes que se necesita dar de baja, en caso de que existan dos meses iguales también se muestra el número diferenciador, para que así el usuario no se confunda como se muestra en la figura 4.33, en la figura 4.34 se muestra la interfaz donde se visualiza un mensaje de que los datos fueron eliminados correctamente.

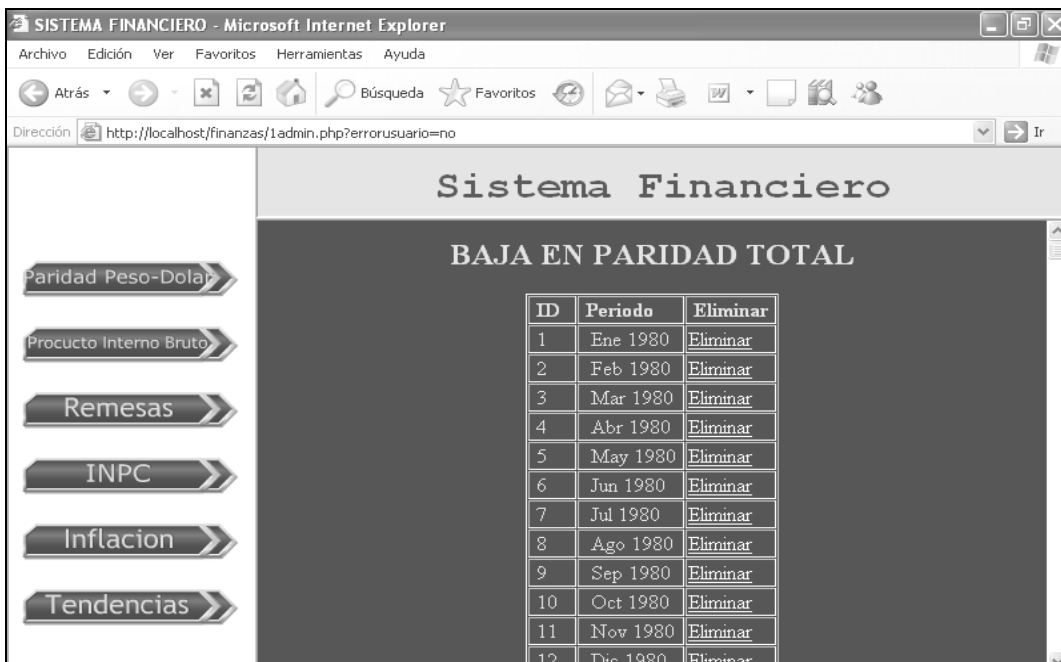


Figura 4.32 Baja de datos cuando no existan dos meses iguales



SISTEMA FINANCIERO - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Dirección: http://localhost/finanzas/1admin.php?errorusuario=no

Sistema Financiero

542	Abr 2006	3	Eliminar
543	May 2006	3	Eliminar
544	Jun 2006	3	Eliminar
545	Jul 2006	3	Eliminar
546	Ago 2006	3	Eliminar
547	Sep 2006	3	Eliminar
548	Oct 2006	3	Eliminar
549	Nov 2006	3	Eliminar
550	Dic 2006	3	Eliminar
551	Ene 2007	3	Eliminar
552	Feb 2007	3	Eliminar
553	Mar 2007	3	Eliminar
554	Abr 2007	3	Eliminar
556	Jun 2007	1	Eliminar
557	Jun 2008	1	Eliminar

Paridad Peso-Dolar

Producto Interno Bruto

Remesas

INPC

Inflacion

Tendencias

Figura 4.33 Baja de datos cuando existan meses iguales

SISTEMA FINANCIERO - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Dirección: http://localhost/finanzas/1admin.php?errorusuario=no

Sistema Financiero

SE ELIMINARON LOS DATOS SATISFACTORIAMENTE

[ELIMINAR MAS DATOS](#)

[IR AL MENU DE PARIDAD](#)

Paridad Peso-Dolar

Producto Interno Bruto

Remesas

INPC

Inflacion

Tendencias

Figura 4.34 Mensaje de que los datos fueron eliminados



En la figura 4.35 se muestra la pantalla para actualizar los datos, lo primero que se visualiza en un cuadro para insertar el período que se quiera actualizar; en la figura 4.36 se muestra la pantalla para el caso de que existan dos meses iguales, en esta se solicita el número diferenciador. Al enviar los datos pedidos se visualiza el formulario con los datos que contiene ese período como se muestra en la figura 4.37. En este formulario se escriben los datos que se deseen reemplazar como se muestra en la figura 4.38 y se le da clic en enviar, en caso de que los datos sean erróneos, se visualiza el mensaje de error, y en caso de que los datos sean correctos en seguida se visualiza el mensaje de que los datos fueron insertados correctamente como se muestra en la figura 4.39.

En caso de que el administrador tenga un error en el formulario y decida no actualizar los datos escritos tendrá que dar clic en reestablecer.



Figura 4.35 Actualizar datos cuando no existan dos meses iguales



SISTEMA FINANCIERO - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Búsqueda Favoritos

Dirección: http://localhost/finanzas/1admin.php?errorusuario=no Ir

Sistema Financiero

ACTUALIZACION

INSERTAR EL PERIODO QUE QUIERES ACTUALIZAR (INSERTA UN ESPACIO ANTES) INSERTAR EL NUMERO DIFERENCIADOR

LOS DATOS INSERTADOS NO SON VALIDOS

Paridad Peso-Dolar

Producto Interno Bruto

Remesas

INPC

Inflacion

Tendencias

Figura 4.36 Actualizar datos cuando existen dos meses iguales

SISTEMA FINANCIERO - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Búsqueda Favoritos

Dirección: http://localhost/finanzas/1admin.php?errorusuario=no Ir

Sistema Financiero

INSERTAR EL PERIODO QUE QUIERES ACTUALIZAR (INSERTA UN ESPACIO ANTES) INSERTAR EL NUMERO DIFERENCIADOR

Apertura a la compra:	<input type="text" value="12.36"/>
Apertura a la venta:	<input type="text" value="12.38"/>
Cierre a la compra:	<input type="text" value="11.23"/>
Cierre a la venta:	<input type="text" value="11.23"/>
Maximo:	<input type="text" value="11.14"/>
Minimo:	<input type="text" value="12.365"/>

Paridad Peso-Dolar

Producto Interno Bruto

Remesas

INPC

Inflacion

Tendencias

Figura 4.37 Actualizar datos mostrando el formulario anterior



SISTEMA FINANCIERO - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Búsqueda Favoritos

Dirección: http://localhost/finanzas/1admin.php?errorusuario=no Ir

Sistema Financiero

INSERTAR EL PERIODO QUE QUIERES ACTUALIZAR (INSERTA UN ESPACIO ANTES) INSERTAR EL NUMERO DIFERENCIADOR

Apertura a la compra:	<input type="text" value="11.36"/>
Apertura a la venta:	<input type="text" value="14.26"/>
Cierre a la compra:	<input type="text" value="13.25"/>
Cierre a la venta:	<input type="text" value="12.11"/>
Maximo:	<input type="text" value="11.25"/>
Minimo:	<input type="text" value="16.35"/>

Figura 4.38 Actualizar datos mostrando los datos actuales

SISTEMA FINANCIERO - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Búsqueda Favoritos

Dirección: http://localhost/finanzas/1admin.php?errorusuario=no Ir

Sistema Financiero

LAS ACTUALIZACIONES SE HICIERON CORRECTAMENTE

Se actualizo 12.36 por 11.36
Se actualizo 12.38 por 14.26
Se actualizo 11.23 por 13.25
Se actualizo 11.23 por 12.11
Se actualizo 11.14 por 11.25
Se actualizo 12.365 por 16.35

[ACTUALIZAR MAS DATOS](#)
[IR AL MENU DE PARIDAD](#)

Figura 4.39 Datos actualizados.



Si el usuario quiere consultar los datos de los indicadores se visualiza una pantalla con todos los datos como se muestra en la figura 4.40. Existen dos formas de buscar los datos necesarios, la primera es la búsqueda por mes en donde se visualizan el mes indicado de todos los años insertados como se muestra en la figura 4.41, y la segunda forma es por año, para ello se da clic en enviar y visualiza los datos del año elegido como se muestra en la figura 4.42.

Periodo	Apertura a la compra	Apertura a la venta	Cierre a la compra	Cierre a la venta	Maximo	Minimo
Ene 1992	3.0644	3.0655	3.0631	3.0642	3.0627	3.0658
Feb 1992	3.0596	3.0606	3.0593	3.0603	3.0588	3.0611
Mar 1992	3.0656	3.0666	3.0657	3.0667	3.0637	3.0685
May 1992	3.0932	3.0942	3.0991	3.1004	3.0968	3.1026
Jun 1992	3.1164	3.1174	3.1116	3.1117	3.1151	3.1181
Jul 1992	3.1127	3.1137	3.1115	3.1125	3.111	3.1141

Figura 4.40 Consulta de todos los datos

Periodo	Apertura a la compra	Apertura a la venta	Cierre a la compra	Cierre a la venta	Maximo	Minimo
Feb 1992	3.0596	3.0606	3.0593	3.0603	3.0588	3.0611
Feb 1993	3.0934	3.0954	3.0923	3.1101	3.0916	3.0965
Feb 1994	3.114	3.1189	3.1166	3.1228	3.1154	3.126
Feb 1995	5.6065	5.659	5.592	5.667	5.6015	5.7214

Figura 4.41 Consulta de los datos del mes de febrero

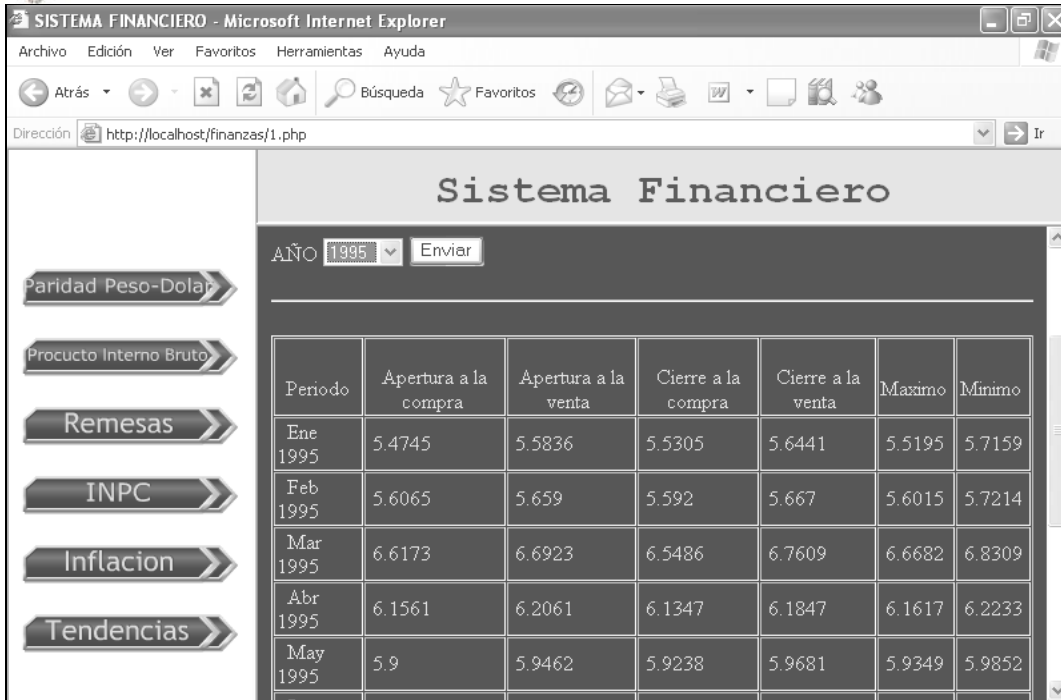


Figura 4.42 Consulta de los datos del año 1995

En la figura 4.43 se muestra el gráfico del producto interno bruto, se visualizan dos líneas porque una línea es el gráfico de los indicador total y la otra es la tendencias en la figura 4.44 se muestra el grafico de las predicciones y en la figura 4.45 se muestran las tendencias de los todos los indicadores.

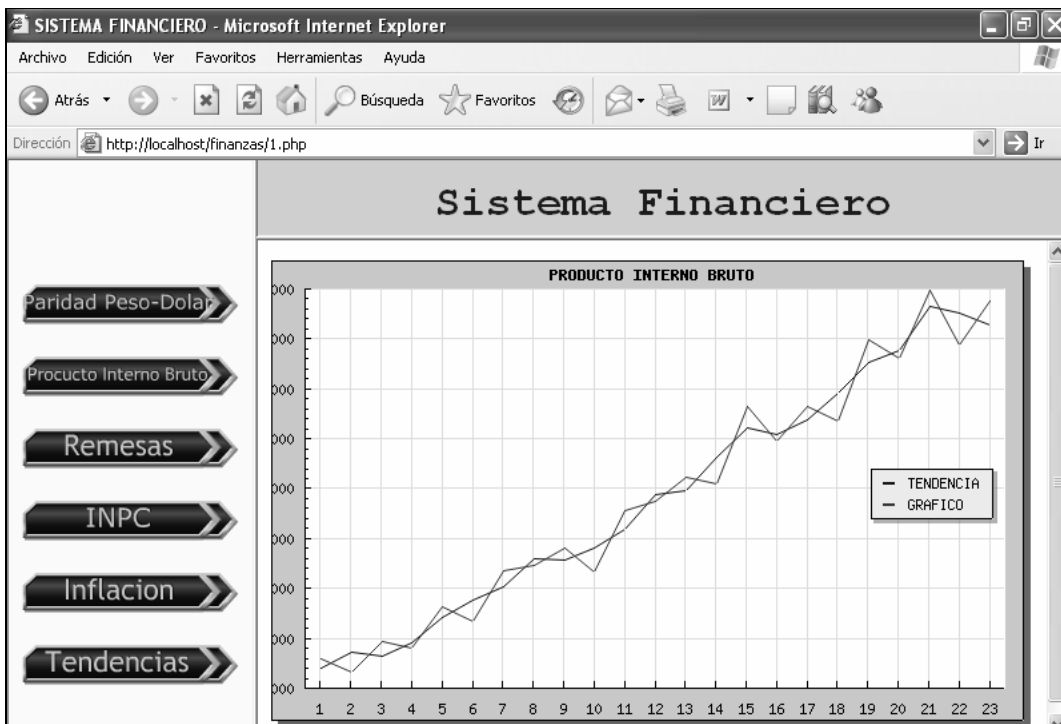


Figura 4.43 Gráfico de tendencias



Figura 4.44 Gráfico de predicciones

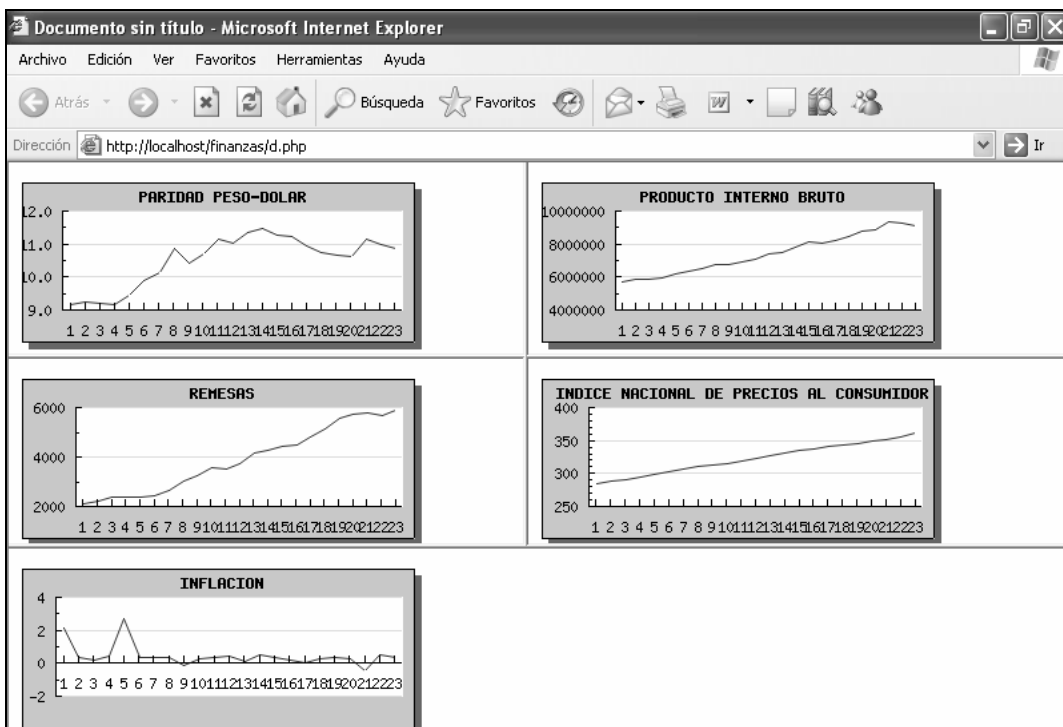


Figura 4.45 Tendencias de los indicadores principales



CONCLUSION

Se realizó el análisis de requisitos y se describió detalladamente, llevando una concordancia para obtener los resultados esperados.

El sistema cubrió las necesidades y las expectativas del cliente al realizar un sistema capaz de proporcionar servicios de búsqueda y consulta sobre los principales indicadores económicos de nuestro país.

El sistema tal como se planteó se realizó mediante una página Web en donde se reconoció y complació los requerimientos del cliente. Para el desarrollo de éste, el usuario quedó satisfecho al visualizar los dos tipos de búsqueda que se pueden realizar.

El sistema aparte de hacer la búsqueda, se pueden consultar los índices; es decir, se muestra el comportamiento de los indicadores mediante los números.

En la página Web se genera y visualizan los gráficos de las tendencias y las predicciones, de manera amigable y eficiente tal como el cliente lo requirió.

Los resultados fueron los esperados ya que el sistema contiene los indicadores de Paridad Peso-Dólar, Producto Interno Bruto, Inflación, Índice Nacional de Precios al Consumidor, Remesas. En donde se visualizan algunas divisiones de cada indicador, para extender la información y de esta manera ser más eficaz.

El sistema que se realizó beneficia a los estudiantes de economía y economistas, inversionistas, así como a las personas que tengan curiosidad sobre el comportamiento de los indicadores, tendencias y principalmente las predicciones de estos, ya que en la Web no existe alguna pagina en la que se pueda orientar acerca de las tendencias y lo mas interesante es el exponer como se comportarán los indicadores en los próximos años. Y así la gente anteriormente mencionada podrá saber lo que les esperará en el sentido financiero.

TRABAJOS A FUTURO

1. Aumentar el número de los indicadores para hacer una página extensa.
2. Aumentar el número de histogramas de las tendencias y predicciones dependiendo de los indicadores
3. Realizar los análisis de las tendencias de los indicadores.



BIBLIOGRAFIA

- [1] Universidad Tecnológica Nacional de Córdoba, Argentina
www.profesores.frc.utn.edu.ar/sistemas/analisisdesistemas/INGENI1.HTM
- [2] Ingeniería de Software
www.wikilearning.com/ingenieria_del_software
- [3] Ingeniería de Software. Un enfoque practico. 5ª ed. Roger S. Pressman Mc Graw Hill.
- [4] Fundamentos de Bases de Datos 5ª ed. McGraw Hill. Silbershatz,Kort, Sudarshan Mc Graw Hill.
- [5] Introducción a los Sistemas de Bases de Datos 7ª ed. C. J. Date. Prentice Hall.
- [6] Diseño de Bases de Datos Adoración de Miguel, Paloma Martínez, Elena Castro. Alfaomega Ra-Ma.
- [7] Desarrollo Web con PHP, Apache y MySQL. Glass, Le Scouarnec, Naramore, Mailer, Stolz, Gerner. Anaya Multimedia.
- [8] Procesamiento de Base de Datos. Fundamentos, diseño e instrumentación. Favid M. Kroenke ed. Prentice Hall.
- [9] Carlo Batini, Stefano Ceri, Shamkant B. Navathe, "Diseño Conceptual de Bases de Datos: Un enfoque de entidades-interrelaciones", Addison-Wesley/Díaz de Santos, 1994.
- [10] Conceptos de uso común en economía
<http://simonfigueroa.tripod.com/Conceptos.htm>
- [11] Análisis Econométrico con Eviews. Carrascal, Gonzáles, Rodríguez. Ra-Ma.