

35



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES
"CAMPUS ARAGÓN"

"SISTEMA INTEGRAL DE AGENCIA DE VIAJES"

T E S I S

QUE PARA OBTENER EL TITULO DE :
INGENIERO EN COMPUTACIÓN

P R E S E N T A N :

ARMANDO MÉNDEZ CARRAZA
JOSÉ GUADALUPE RODRIGUEZ MARTÍNEZ

ASESOR: ING. LILIANA HERNÁNDEZ CERVANTES

MÉXICO

TESIS CON
FALLA DE ORIGEN

2002



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

"SISTEMA INTEGRAL DE AGENCIA DE VIAJES"

**TESIS CON
FALLA DE ORIGEN**

*A mis padres y hermanos:
Les agradezco el apoyo que me brindaron todos estos años de altibajos, y gracias a ello logramos
juntos llegar a la meta.*

*A mi esposa y a mis hijos:
Les agradezco el tiempo que no les pude dedicar, el cual gracias a su amor y comprensión
me lo cedieron para poder culminar un gran logro que se fomento desde hace años,
siendo este alcanzado por ustedes y por mí.*

*Quiero Agradecer en primer lugar a Dios por la maravillosa oportunidad que me brinda.
A mis Padres ya que sin ellos jamás hubiera llegado a la meta.
A Xochitl por el apoyo y comprensión que me brindo para que juntos podamos cumplir este sueño.
Por supuesto a esa persona tan importante en mi vida a la que no le he podido dedicar el tiempo
necesario; pero por ti lo intente y lo hemos logrado Pepito.
A ese bebe que muy pronto estará con nosotros y por quien tengo que luchar aun mas.
También se lo dedico a mis hermanos para que siempre luchen por lograr sus objetivos.*

Y por último quiero agradecer a Armando por que gracias a su amistad lo logramos.

¡Muchas Felicidades!

A cada uno de las personas que nos ayudaron a terminar este proyecto solo nos resta decirles gracias.

**TESIS CON
FALLA DE ORIGEN**

INDICE

1. INTRODUCCION	1
1.1 SISTEMA DE INFORMACION	3
1.2 BASES DE DATOS	5
1.2.1. MODELOS DE BASE DE DATOS	6
1.2.2. MODELO JERARQUICO	6
1.2.3. MODELO RELACIONAL	8
1.2.4. ATRIBUTOS LLAVE (K)	12
1.2.5. INTEGRIDAD DE ENTIDADES	14
1.2.6. OPERACIÓN INSERTAR	16
1.2.7. OPERACIÓN ELIMINAR	17
1.2.8. OPERACIÓN MODIFICAR	18
1.2.9. EL ALGEBRA RELACIONAL.	19
1.2.10. OPERACIÓN SELECCIONAR	19
1.2.11. OPERACIÓN PROYECTAR	20
1.2.12. OPERACIONES DE LA TEORIA DE CONJUNTOS	20
1.2.13. UNION	21
1.2.14. INTERSECCION	21
1.2.15. DIFERENCIA	21
1.2.16. REUNION	22
1.2.17. OPERACIONES DE CERRADURA RECURSIVA	24
1.3 UTILIZACION DE LA RED	25
1.3.1. RED DE AREA LOCAL (LAN)	25
1.2.2. RED METROPOLITANA (MAN)	26
1.2.3. RED DE AREA AMPLIA (WAN)	26
1.2.4. TOPOLOGIAS DE REDES	27
1.2.4.1. TOPOLOGIA EN BUS	28
1.2.4.2. TOPOLOGIA EN ANILLO	29
1.2.4.3. TOPOLOGIA EN ESTRELLA	30
1.2.4.4. TOPOLOGIA HIBRIDA	32
1.2.5. PROTOCOLOS DE COMUNICACION	34
1.4 CLIENTE/SERVIDOR	38
1.4.1. SERVIDORES DE ARCHIVOS	39
1.4.2. SERVIDORES DE BASES DE DATOS	40
1.4.3. SERVIDORES DE TRANSACCIONES	40
1.4.4. SERVIDORES DE GROUPWARE	41
1.4.5. SERVIDORES DE OBJETOS	42
1.4.6. SERVIDORES WEB	43
1.4.7. SERVIDORES AMPLIOS Y CLIENTES GRANDES	43
1.4.8. ELEMENTOS DE CONSTRUCCION DE CLIENTE/SERVIDOR	44
2. REQUERIMIENTOS DE LA EMPRESA DE VIAJES TURISTICOS	47
2.1 GIRO DE LA EMPRESA	48
2.2 NECESIDADES	50
2.3 PROBLEMÁTICA ACTUAL Y REQUERIMIENTOS	50
2.4 PROPUESTA DE SOLUCION	53

TESIS CON
FALLA DE ORIGEN

3. ANÁLISIS	55
3.1. HERRAMIENTAS DE DESARROLLO	57
3.1.1. INTEGRACION	59
3.2. ANALISIS DEL SISTEMA DE AGENCIA DE VIAJES TURISTICOS	60
3.3 NORMALIZACION	64
3.3.1. PRIMERA REGLA DE NORMALIZACION	64
3.3.2. SEGUNDA REGLA DE NORMALIZACION	65
3.3.3. TERCERA REGLA DE NORMALIZACION	66
3.3.4. CUARTA REGLA DE NORMALIZACION	69
3.3.5. QUINTA REGLA DE NORMALIZACION	69
4. DESARROLLO	73
4.1. DESCRIPCION DE LOS CAMPOS PRINCIPALES	75
4.2. DESCRIPCION DE LA OPERACION	82
5. CONCLUSION	101
BIBLIOGRAFIA	103

TESIS CON
FALLA DE ORIGEN

CAPITULO I

INTRODUCCIÓN

TESIS CON
FALLA DE ORIGEN

1. INTRODUCCION

Debido a los constantes avances tecnológicos, una empresa que desee mantenerse en activo tiene que ir modernizándose, claro que esto es basándose en su capacidad económica, porque si no lo hace tendera a irse rezagando y por ende desaparecerá del mercado.

Uno de los objetivos de este material es apoyar en la elaboración de un sistema **cliente/servidor**. Se desarrollo un sistema informático para la venta y reservación principalmente de viajes turísticos, como es una empresa relativamente pequeña la cual principalmente se apoya en proveedores de servicios tales como empresas que se dedican a la transportación de pasajeros o a la reservación en pequeños hoteles en ciertos puntos de la República Mexicana, no creía necesario tener un sistema con el cual realizar su operación diaria, pero al momento de querer competir con otras empresas de su tipo se dieron cuenta de la necesidad de implementar un sistema.

Para poder llevar a cabo el desarrollo de dicho sistema se tuvo que iniciar desde entrevistas con el usuario para poder entender sus necesidades hasta verificar su disponibilidad económica para la compra de los elementos necesarios para crear y mantener el sistema que pidieron.

Esperamos que con la lectura de este material podamos apoyar en algo con la interesante tarea de crear un sistema **cliente/servidor**.

Al iniciar este trabajo se penso en dividirlo en dos principales módulos, uno donde se expusiera toda la teoría necesaria para la correcta elaboración de un sistema utilizando un modelo **relacional** para el manejo de la **base de datos**. Y otro donde se desarrollara propiamente el **sistema**.

En el primer capitulo se expone la teoría desde lo que es un sistema hasta los elementos de construcción de sistemas **cliente/servidor**, pasando por el álgebra relacional donde se explican las operaciones básicas para el manejo de **base de datos** respetando las reglas de **integridad relacional** para no afectar la información del **cliente**.

También se comenta un poco de **redes** ya que como es un **sistema cliente/servidor** es importante el manejo de los tipos de **redes** para que exista una buena comunicación.

En el segundo capitulo se expresa lo que realmente es la base del trabajo los requerimientos de la empresa de Agencia de Viajes, dando una explicación de la problemática actual, cuales son sus requerimientos y cuales son las necesidades reales de las personas que operan principalmente en esta empresa. Así también se da una propuesta de solución a los problemas y necesidades actuales de dicha empresa.

El Tercer capitulo se trata de las herramientas con las que se desarrollo y la forma de integrarlas para que en forma conjunta se pueda trabajar y no exista una incompatibilidad al momento de operar. En este capitulo también comentamos la normalización de las **tablas** de la **base de datos** por supuesto se explica ya con **datos** la primera, segunda y tercera **reglas de normalización**, aunque en forma más sencilla también se hace referencia a la cuarta y quinta reglas.

El Cuarto capitulo se trata del desarrollo del **sistema**, explicando las principales **tablas** de la **base de datos** así como los **campos** representativos de cada **tabla**, además se intenta describir la operación del mismo.

Como se ve se intento ordenar la información para que cuando se quiera desarrollar un sistema se pueda apoyar en este documento.

1.1. SISTEMA DE INFORMACION.

Hoy en día para que una empresa tenga una buena organización requiere tener un buen control de la información que maneja. En el caso que nos ocupa - Agencia de Viajes -, es primordial tener un control fiable de la venta y reservación tanto de los boletos como de los viajes que oferta. Por lo tanto decimos que necesita un **sistema** de información adecuado a las reglas de su negocio, esto para manejar la información requerida por los usuarios principales en forma completa y segura.

Para entender mejor el por que de lo anterior, necesitamos explicar el significado de **sistema**. Por lo que un **sistema** es una reunión o conjunto de elementos relacionados que interactuan entre si para lograr un fin común. Un sistema esta compuesto de **entradas, salidas y procesos**; en un diagrama a bloques se puede representar de la siguiente manera como lo muestra la figura 1.1.

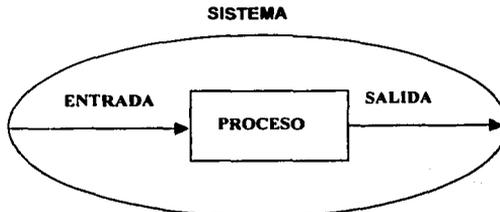


Figura 1.1

**TESIS CON
FALLA DE ORIGEN**

Podemos decir que las **entradas** constituyen la fuerza que se suministra al **sistema**. Estas pueden ser recursos humanos o información que se le provee. El **proceso** es lo que transforma una **entrada** en una **salida**, como proceso podemos nombrar una tarea, una operación, etc., las **salidas** son los resultados que se obtienen de procesar las **entradas**, estos pueden tomar la forma de productos o información.

En nuestro caso podemos definir un **sistema** de información como un conjunto de programas (software), periféricos (hardware), información (**base de datos**), procedimientos (requerimientos del usuario), etc. que tienen como fin primordial ayudar a mantener integridad en la información que maneja la empresa y el punto importante es la **base de datos**.

Basándose en su servicio, el **sistema** de información puede ser clasificado en:

SISTEMA DE PROCESAMIENTO DE DATOS.- Este tipo de **sistema** se desarrolla para soportar una gran cantidad de información. Aquí se pone a disposición de los usuarios toda la información que necesitan para el desempeño de sus funciones.

SISTEMA DE INFORMACIÓN PARA LA GESTIÓN Y LA ADMINISTRACIÓN.- Este tipo de **sistema** proporciona la información necesaria para controlar la evolución de la empresa y el cumplimiento de los objetivos operativos. Se utiliza una **base de datos** compartida para tener acceso a la información específica que necesita cada usuario.

SISTEMA DE APOYO PARA LA TOMA DE DECISIONES.- Este **sistema** se diseña con una orientación hacia el usuario final, es decir aunque depende también de una **base de datos**, la decisión depende del mismo usuario responsable del **sistema**; este tipo de sistema esta destinado mas bien a los usuarios tácticos.

SISTEMA PARA LA DIRECCIÓN.- Este tipo de **sistema** relaciona en la misma **base de datos** toda la información significativa de la organización. Permite acceder a ella tanto en forma jerárquica para la toma de decisiones como para análisis comparativos del desarrollo de la empresa.

Basándonos en los párrafos anteriores llegamos a la conclusión que un **sistema** de información es la parte fundamental de una organización ya que proporciona, usa y distribuye la información.

1.2. BASES DE DATOS.

La parte fundamental de un **sistema** de información es la base de datos. Una base de datos es un conjunto de datos relacionados entre sí, mientras que por datos entendemos a los hechos conocidos que pueden registrarse y tienen un significado implícito, además una base de datos tiene las propiedades siguientes:

- a) Representa algún aspecto del mundo real.
- b) Conjunto de datos con cierto significado inherente.
- c) Se diseña, construye y prueba con datos para un propósito específico.

Con el uso de las bases de datos, se da un giro a la forma de administrar la información ya que con esta se tiene una manera fácil de recuperarla sin mucho gasto y en base a ella se pueden tomar decisiones adecuadas para una empresa. La base de datos debe contar con los siguientes objetivos:

- a) Control de seguridad de datos.- Para proteger los datos es necesario contar con diferentes controles de integridad y seguridad, así como tener la capacidad de recuperación de datos en caso que estos sean perdidos accidentalmente.
- b) Sensibilización de datos.- Esto se refiere a que los datos pueden ser utilizados por las diferentes aplicaciones que necesite el usuario.
- c) Reducción del costo de mejoramiento.- La base de datos debe tener la capacidad de reducir los costos que están asociados al avance de la tecnología, sin tener que realizar grandes cambios en la estructura de la base de datos.

Una característica fundamental del enfoque de bases de datos es que el sistema no solo contiene la base de datos misma, sino también una definición o descripción completa de la base de datos. Esta definición se almacena en el catálogo del sistema, que contiene información como la estructura de cada archivo, el tipo y formato de almacenamiento de cada elemento de información y diversas restricciones que se aplican a los datos. A la información almacenada en el catálogo se le denomina metadatos, y estos describen la estructura de la base de datos primaria.

Con el concepto de estructura de una base de datos nos referimos a los tipos de datos, los vinculos y las restricciones que deben cumplirse para esos datos.

Una herramienta útil para el manejo de bases de datos es el Sistema de Gestión de Bases de Datos (**SGBD**; en inglés Database Management System: DBMS) que es un conjunto de programas que permite a los usuarios crear y mantener una base de datos.

1.2.1. MODELOS DE BASE DE DATOS.

Los modelos de datos de alto nivel utilizan conceptos como **entidades**, **atributos** y **vinculos**. Una **entidad** representa un objeto o concepto del mundo real. Un **atributo** representa alguna propiedad de interés que da una descripción más amplia de una **entidad**. Un **vinculo** describe una interacción entre dos o más **entidades**.

El objeto básico que se representa en el modelo **Entidad - Relación (ER)** es la **entidad**. La cual puede ser un objeto con existencia física o un objeto con existencia conceptual. Cada **entidad** tiene propiedades específicas, llamadas **atributos**. Una **entidad** particular tendrá un valor para cada uno de los **atributos** que describen a cada **entidad** y constituyen una parte decisiva de los datos almacenados en la base de datos.

En el modelo **ER** se manejan varios tipos de **atributos**: simples o compuestos, monovaluados o multivaluados y almacenados o derivados. Los **atributos** compuestos se pueden dividir en componentes más pequeños, que representan **atributos** más básicos con su propio significado independiente.

Existen varios modelos de base de datos, los mas importantes son el modelo jerárquico y el modelo relacional, sobre el cual basaremos nuestro documento.

1.2.2. MODELO JERARQUICO

El primer modelo de base de datos fue el modelo jerárquico. En este modelo se tiene una **tabla** como piedra angular de toda la base de datos. Es decir a partir de esta **tabla** se pueden encontrar todos los datos. Lo anterior se puede apreciar en la figura 1.2.



Figura 1.2.

En este ejemplo para acceder la **tabla** facturas y determinar que factura pertenece a un cliente determinado se tiene que realizar la búsqueda por medio de la clave del cliente. Si lo que queremos es saber que viajes compro el cliente de igual manera la búsqueda sería por medio de la clave del cliente.

Pero si lo que deseamos es saber que viaje compro un cliente con una factura determinada, tendríamos que buscar la factura y posteriormente ir a la **tabla** clientes para obtener la clave del cliente, así con esta llave realizar la búsqueda en la **tabla** viajes, con lo que para obtener una información se tienen que dar dos pasos.

El sistema ideal es que cada **tabla** tenga una llave de cliente, pero si no lo tiene, se accede a partir del código de la **tabla** subordinada.

Este modelo puede crecer sobre la base de las necesidades de acceso, por ejemplo si lo que se requiere es localizar una factura a partir del viaje, se tendrían que regresar a la **tabla** clientes, pero esto se puede evitar creando un puente entre las **tablas** viajes y facturas; Si se necesitara obtener mas información de otras **tablas** subordinadas, se tendrían que generar mas puentes entre cada **tabla**. Cada uno de estos puentes es una **entidad** mas de la base de datos.

De este modo la base de datos iría creciendo tanto en numero de **tablas** como numero de puentes, tejiendo algo parecido a una red. Como se puede apreciar en la figura 1.3.

TESIS CON
FALLA DE ORIGEN

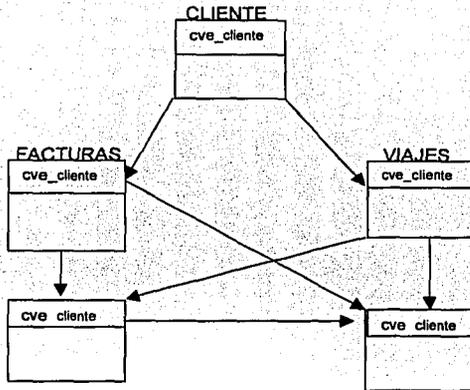


Figura 1.3.

1.2.3. MODELO RELACIONAL

Hacia 1970, E.F. Codd diseñó un modelo de base de datos en el cual no hacen falta puentes, si no que la propia base de datos es suficientemente potente como para buscar por los diferentes **campos** de todas las **tablas**.

Las **tablas** son unidades de almacenamiento de filas y columnas. Las filas son como los **registros**, solo que podemos tener filas provenientes de una consulta con **campos** de varias **tablas**. Las columnas son los **campos**; el modelo relacional puede obtener columnas (campos) ficticios, que puede ser el resultado de cálculos (sumas, multiplicaciones, porcentajes, etc.) de otras columnas.

En el modelo relacional, una fila se denomina **tupla**, una cabecera de columna es un **atributo** y la **tabla** es una **relación**. El tipo de datos que describen los tipos de valores que pueden aparecer en cada columna se llama **dominio**.

Es decir este modelo considera la base de datos como una colección de **relaciones**, donde una **relación** representa una **tabla**, en que cada fila representa una colección de valores que describen una **entidad**. En términos informales, cada **relación** semeja una **tabla** o, hasta cierto punto, un archivo simple.

TESIS CON
FALLA DE ORIGEN

Si visualizamos una **relación** como una **tabla** de valores, cada fila de la **tabla** representa una colección de valores de datos relacionados entre sí. Dichos valores se pueden interpretar como hechos que describen una **entidad** del mundo real. El nombre de la **tabla** y los nombres de las columnas ayudan a interpretar el significado de los valores que están en cada fila de la **tabla**. En las tablas 1.1 y 1.2 intentamos que se aprecie lo mencionado anteriormente.

RELACION ↘

CLIENTE				
Clave	Nombre	Dirección	Teléfono	Estado
004	Alfredo López	Cda. Del Puerto	58756725	México
002	Pedro Martínez	Insurgentes	59352600	Hidalgo
001	Juana García	Churubusco	52369299	Chihuahua
003	Isabel Domínguez	Ignacio Zaragoza	23156890	Monterrey

↑ ATRIBUTO → TUPLA

Tabla 1.1.

RELACION ↘

FACTURA				
Clave	Código	Descripción	Cantidad	Importe
004	000001	Libreta	3	210.00
002	000002	Cartulina	10	500.00
001	000003	Bolígrafo	50	90.00
003	000004	Filtro	100	150.00

↑ ATRIBUTO → TUPLA

Tabla 1.2.

Los componentes básicos de una base de datos relacional son:

TESIS CON
FALLA DE ORIGEN

ESTRUCTURA DE DATOS. Es el esquema de *relación*, este esta constituido por nombre de *relación* y una lista de *atributos*. Cada *atributo* es el nombre de un rol jugado por el *dominio* en el esquema de la *relación*.

INTEGRIDAD DE DATOS. Así se conocen a las reglas semánticas que controlan los comportamientos de la base de datos.

MANIPULACION DE DATOS. Son las operaciones en las relaciones.

DOMINIO. Un dominio D es un conjunto de valores atómicos. (Por atómico queremos decir que cada valor del dominio es indivisible en lo tocante al modelo relacional.) Un método común de especificación de los dominios consiste en especificar un tipo de datos al cual pertenecen los valores que constituyen el dominio.

Las restricciones de *dominio* especifican que el valor de cada *atributo* A debe ser un valor atómico del *dominio* $dom(A)$ para ese *atributo*. Los tipos de datos asociados a los *dominios* por lo regular incluyen los tipos de datos numéricos estándar de los números enteros y reales.

TUPLA. Definimos por *tupla* a una lista ordenada de valores.

RELACIÓN. Una *relación* se define como un conjunto de *tuplas*.

Matemáticamente, los elementos de un conjunto no están ordenados; por tanto, las *tuplas* de una *relación* no tienen un orden específico. En cambio, los *registros* de un archivo se almacenan físicamente en el disco, de modo que siempre existe un orden entre ellos.

El ordenamiento de las *tuplas* no forma parte de la definición de una *relación* porque la *relación* intenta representar los hechos en un nivel lógico o abstracto. Podemos especificar muchos ordenamientos lógicos en una *relación*.

De acuerdo con la definición anterior de *relación*, una *tupla* es una lista ordenada de n valores, así que el orden de los valores de una *tupla* – y por tanto los *atributos* en la definición de un esquema de la *relación* – es importante. No obstante, en un nivel lógico, el orden de los *atributos* y de sus valores en realidad no es importante en tanto se mantenga la correspondencia entre *atributos* y valores.

Cada valor en una *tupla* es un valor atómico; esto es, no es divisible en componentes en lo que respecta al modelo relacional.

Un esquema de *relación* R, denotado por $R(A_1, A_2, \dots, A_n)$, se compone de un nombre de la *relación*, R, y una lista de *atributos*, A_1, A_2, \dots, A_n . Cada *atributo* es el nombre de un papel desempeñado por algún *dominio* D en el esquema R. Se dice que D es el *dominio* de A_i y se denota con $dom(A_i)$. Un esquema de *relación* sirve para describir una *relación*; R es el nombre de la *relación*. El grado de una *relación* es el número de

atributos, n, de su esquema de **relación**. El esquema de una **relación** se puede interpretar como una declaración.

Como ejemplo tenemos las **relaciones Cliente** con los **atributos Nombre, Calle, Ciudad y Numero de cuenta**, así mismo tenemos a **Cuenta** con numero de **Cuenta y Saldo**

Cliente(Nombre, Calle, ciudad, Numero de cuenta).

Cuenta(Numero de cuenta, Saldo)

Cabe señalar que algunas **relaciones** pueden representar hechos acerca de **entidades**, en tanto que otras pueden representar hechos sobre vínculos.

La primera definición de **relación** implica ciertas características que distinguen a una **relación** de un archivo o de una **tabla**.

Las **relaciones** se distinguen de las **tablas** o archivos ordinarios por varias características. La primera es que las **tuplas** de una **relación** no están ordenadas. La segunda consiste en el ordenamiento de los **atributos** en un esquema de **relación** y el ordenamiento correspondiente de los valores dentro de una **tupla**.

Cuando se desea implementar una base de datos relacional para una aplicación compleja, se suele empezar por diseñar con mucho cuidado el esquema de la base de datos. Esto implica decidir cuáles **atributos** deben ir juntos en cada **relación**, elegir nombres apropiados para las **relaciones** y sus **atributos**, especificar los **dominios** y tipos de datos los diversos **atributos**, identificar las **llaves candidatas** y escoger una **llave primaria** para cada **relación**, y especificar todas las **llaves externas**.

La **tabla** de **relación**, consiste de un conjunto de columnas nominadas (que tienen un nombre) y una cantidad arbitraria de filas (**tuplas**). Cada columna esta asociada con un **dominio** que esta especificado con un tipo a la columna.

Cada **tabla** de **relación**, corresponde a un archivo almacenado en el nivel fisico, por lo que podemos decir que una base de datos relacional consiste de un conjunto de **tablas - relaciones** relevantes.

El esquema de **relación**, se usa para describir una **relación** o una instancia de **relación**.

Además de lo expresado, hay que recordar que los enlaces entre las tablas, son una parte fundamental en el modelo relacional, ya que permiten encontrar información de una tabla a partir de la información de otra.

Para poder realizar los enlaces entre tablas, basta con que exista un campo – o campos- común que determine de forma única un registro de la segunda. Esto es lo que denominamos llave.

1.2.4. ATRIBUTOS LLAVE (K).

Una **llave** es un conjunto de campos de una **tabla** que determina de forma única un registro de esta **tabla**. Generalmente se recurre a un campo único, simple y fijo.

La **llave** se determina a partir del significado de los **atributos** en el esquema de la **relación**; por ende, la propiedad no varía con el tiempo; debe seguir siendo válida aunque insertemos **tuplas** nuevas en la **relación**.

Una **relación** se define como un conjunto de **tuplas**. Por definición, todos los elementos de un conjunto son distintos; por tanto, todas las **tuplas** de una **relación** deben ser distintas. Esto significa que no puede haber dos **tuplas** que tengan la misma combinación de valores para todos sus **atributos**. Por lo regular existen otros subconjuntos de **atributos** de un esquema de **relación R** con la propiedad de que no debe haber dos **tuplas** en un mismo ejemplar de **relación r** de **R** con la misma combinación de valores para estos **atributos**.

Se le conoce como **superllave** al conjunto de **atributos** que contienen una **llave**, también se define como **atributo** o combinación de **atributos** que le dan identificación única a cada **entidad** en una **tabla**. Es decir todo conjunto de **atributos** de este tipo es una **superllave** del esquema de **relación R**. Toda **relación** tiene por lo menos una **superllave**. Sin embargo, una **superllave** puede tener **atributos** redundantes, así que un concepto más útil es el de **llave**, que carece de redundancia.

El valor de un **atributo llave** puede servir para identificar de manera única una **tupla** de la **relación**. Ahora bien, cuando un esquema tiene más de una **llave**, cada **llave** es una **llave candidata**. También se dice que es la **superllave** mínima.

LLAVE PRIMARIA. Es usada para identificar **tuplas** en una **relación**. Es la **llave candidata** que es seleccionada para dar identificación única a todos los otros **atributos** en una fila dada. No puede contener entradas nulas.

LLAVE SECUNDARIA. Un **atributo** (o combinación de **atributos**) que se usa estrictamente para propósitos de recuperación. También se le llama **LLAVE ALTERNATIVA**.

En este ejemplo, se puede ver que la **tabla** contiene un **registro** (líneas) y **n - campos** (columnas), y como se vio anteriormente no existe un ordenamiento específico, por lo que para localizar un **x - registro** es necesario apoyarnos en la **llave**, la cual nos dirá en que posición se encuentra el **registro** buscado y este se realiza por medio de direcciones lógicas.

En la tabla 1.3 se requiere el registro del señor Alfredo López y para encontrarlo se busca por la llave que en este caso es la clave del cliente que para nuestro ejemplo

tiene como clave el número 004, para la tabla 1.4 también tenemos que la llave es la clave del cliente como ya sabemos que el cliente es Alfredo López, solo basta buscar en la tabla facturas con la clave del cliente Alfredo López y así sabremos que fue lo que compro este cliente.

CLIENTE

↑ LLAVE

Clave	Nombre	Dirección	Teléfono	Estado
004	Alfredo López	Cda. Del Puerto	58756725	México
002	Pedro Martínez	Insurgentes	59352600	Hidalgo
001	Juana García	Churubusco	52369299	Chihuahua
003	Isabel Domínguez	Ignacio Zaragoza	23156890	Monterrey
005	Alejandro Pérez	Angela Peralta	24217880	Cuernavaca
006	José Luis Hernández	Tecamac	12155590	Puebla
007	Juan González	Moctezuma	15236898	San Luis Potosí
008	Martha Fernández	Toluca	55152813	México
009	Verónica Quintero	Emiliano Zapata	43126871	Morelia
010	Rocío Ruiz	Tlalpan	13456950	Puebla

REGISTRO →

Tabla 1.3.

FACTURA

↑ LLAVE ↑ CAMPO

Clave	Folio	Descripción	Cantidad	Importe
004	000001	Libreta	3	210.00
002	000002	Cartulina	100	300.00
001	000003	Bolígrafo	50	90.00
003	000004	Fieltro	100	150.00
007	000003	Libreta	100	7000.00
010	000004	Bolígrafo	1	1.80
009	000005	Regla	1	100.00
009	000005	Sacapuntas	50	175.00
009	000005	Hojas	1000	45.00
008	000006	C.D.	12	144.00

Tabla 1.4.

Una base de datos relacional suele contener muchas **relaciones** y en éstas las **tuplas** están relacionadas de diversas maneras.

Las reglas que tienen que seguir las *llaves* son:

1.2.5. INTEGRIDAD DE ENTIDADES.

La restricción de integridad de *entidades* establece que ningún valor de la *llave primaria* puede ser nulo. Si el valor de la *llave primaria* tiene valores nulos implica que no podemos identificar algunas *tuplas*.

Para dar una definición más en forma de integridad referencial primero se tiene que definir el concepto de *llave externa*. Las condiciones que debe satisfacer una *llave externa* (dadas a continuación) especifican una restricción de integridad referencial entre los dos esquemas de *relaciones* R1 y R2.

Varios tipos de restricciones pueden considerarse como parte del modelo relacional. Los conceptos de *superllave*, *llave candidata* y *llave primaria* especifican las restricciones de *llave*. La restricción de integridad de *entidades* prohíbe los valores nulos en los *atributos* de *llave primaria*. La restricción de *integridad referencial* entre *relaciones* sirve para mantener la consistencia de las referencias entre *tuplas* de diferentes *relaciones*. En otras palabras las reglas de integridad referencial son las que mantienen la coherencia lógica de la base de datos.

El valor de la *llave externa* puede ser nulo o tiene que ser similar al valor de una *llave primaria* de la *tabla* con la cual se establece la *relación*.

Como se ve en el ejemplo siguiente (tabla 1.5 y tabla 1.6), en una base de datos como la indicada anteriormente de clientes y facturas, lo lógico es que si borramos un cliente, se borren las líneas de la factura que estén asociadas a la misma clave de cliente.

CLIENTE

Clave	Nombre	Dirección	Teléfono	Estado
004	Alfredo López	Cda. Del Puerto	58756725	México
002	Pedro Martínez	Insurgentes	59352600	Hidalgo
001	Juana García	Churubusco	52369299	Chihuahua
003	Isabel Domínguez	Ignacio Zaragoza	23156890	Monterrey
005	Alejandro Pérez	Angela Peralta	24217880	Cuernavaca
006	José Luis Hernández	Tecamac	12155590	Puebla
007	Juan González	Moctezuma	15236898	San Luis Potosí
008	Martha Fernández	Toluca	55152813	México
009	Verónica Quintero	Emiliano Zapata	43126871	Morelia
010	Rocío Ruiz	Tlalpan	13456950	Puebla

Tabla 1.5.

FACTURA

Clave	Folio	Descripción	Cantidad	Importe
004	000001	Libreta	3	210.00
004	000001	Cartulina	100	300.00
004	000001	Boligrafo	50	90.00
003	000002	Fieltro	100	150.00
007	000003	Libreta	100	7000.00
010	000004	Boligrafo	1	1.80
009	000005	Regla	1	100.00
009	000005	Sacapuntas	50	175.00
009	000005	Hojas	1000	45.00
008	000006	C.D.	12	144.00

Tabla 1.6.

El uso de reglas de integridad referencial también prohíbe a los usuarios que añadan **registros** a una **tabla** que no tenga índice principal, o modificar valores de una **tabla** muestra que repercutan en **tablas** subordinadas.

En una base de datos con muchas **relaciones**, suele haber muchas restricciones de integridad referencial. Para especificar dichas restricciones es preciso, primero, comprender con claridad el significado o papel que cada uno de los conjuntos de **atributos** desempeña en los diversos esquemas de relaciones de la base de datos.

Las restricciones de integridad referencial casi siempre surgen de los vínculos entre las entidades representadas por los esquemas de **relaciones**.

Cabe señalar que una **llave externa** puede hacer referencia a su propia **relación**.

Se puede representar en un diagrama las **relaciones** de integridad referencial trazando un arco dirigido de cada **llave externa** a la **relación** a la cual hace referencia. Para mayor claridad, la punta de la flecha puede apuntar a la **llave primaria** de la **relación** referida.

Se tiene que especificar todas las restricciones de integridad en el esquema de la base de datos relacional si es que se quiere mantener dichas restricciones en todos los ejemplares de la base de datos.

Dentro del modelo relacional es posible realizar operaciones y estas se pueden clasificar en obtenciones y actualizaciones.

Las operaciones del álgebra relacional, con los que podemos especificar obtenciones, son tres las operaciones de actualización básicas que se efectúan con **relaciones**:

Insertar.- Esta operación sirve para insertar una o más **tuplas** nuevas en una **relación**

Eliminar.- Sirve para eliminar **tuplas**

Modificar.- Sirve par alterar los valores de algunos **atributos**.

Cabe aclarar que siempre que se apliquen operaciones de actualización se debe cuidar de no violar las restricciones de integridad especificadas en el esquema de la base de datos relacional.

Para que quede mas claro la forma de utilizar las operaciones del álgebra relacional, definiremos las tres operaciones básicas de la siguiente manera:

1.2.6. INSERTAR.

La operación insertar proporciona una lista de valores de **atributos** para una nueva **tupla** t que se ha de insertar en una **relación** R . La inserción puede violar cualquiera de los cuatro tipos de restricciones siguientes:

- Las restricciones de **dominio** pueden violarse si se proporciona un valor de **atributo** que no aparezca en el **dominio** correspondiente.
- Las restricciones de **llave** pueden violarse si un valor **llave** de la nueva **tupla** t ya existe en otra **tupla** de la **relación** $r(R)$.
- La integridad de **entidades** puede violarse si la **llave primaria** de la nueva **tupla** t es nula.
- La integridad referencial puede violarse si el valor de cualquier **llave externa** de t hace referencia a una **tupla** que no existe en la **relación** referenciada.

Si una inserción viola una o más restricciones, se dispone de dos opciones. La primera es rechazar a inserción, en cuyo caso sería útil que el **SGBD** explicara al usuario por qué fue rechazada. La segunda es intentar corregir la razón por la que se rechazó la inserción.

Como ejemplo podemos señalar el siguiente(tabla 1.7), donde se intenta insertar un registro que ya existe (tabla 1.8), también si ponemos la clave en blanco e intentamos insertarlo el sistema nos indicara que esta violando alguna restricción.

CLIENTE

Clave	Nombre	Dirección	Teléfono	Estado
004	Alfredo López	Cda. Del Puerto	58756725	México
002	Pedro Martínez	Insurgentes	59352600	Hidalgo
001	Juana García	Churubusco	52369299	Chihuahua
003	Isabel Domínguez	Ignacio Zaragoza	23156890	Monterrey
005	Alejandro Pérez	Angela Peralta	24217880	Cuernavaca
006	José Luis Hernández	Tecamac	12155590	Puebla
007	Juan González	Moctezuma	15236898	San Luis Potosí
008	Martha Fernández	Toluca	55152813	México
009	Verónica Quintero	Emiliano Zapata	43126871	Morelia
010	Rocio Ruiz	Tlalpan	13456950	Puebla

Tabla 1.7.

004	Alfredo López	Cda. Del Puerto	58756725	México
-----	---------------	-----------------	----------	--------

Tabla 1.8.

1.2.7. ELIMINAR.

La operación eliminar sólo puede violar la integridad referencial, si las **llaves externas** de otras **tuplas** de la base de datos hacen referencia a la **tupla** que se ha de eliminar. Para especificar la eliminación, una condición expresada en términos de los **atributos** de la **relación** selecciona la **tupla** (o **tuplas**) por eliminar.

Si una operación de eliminación provoca una violación se dispone de tres opciones. La primera es rechazar la eliminación. La segunda es tratar de propagar la eliminación eliminando las **tuplas** que hacen referencia a la **tupla** que se desea eliminar. Una tercera opción es modificar los valores del **atributo** de referencia que provocan la violación; todos esos valores se pondrían en nulo o se modificarían de modo que hicieran referencia a otra **tupla** válida. Nótese que, si un **atributo** de referencia que origina una violación forma parte de la **llave primaria**, no se puede cambiar a nulo, pues si se hiciera se violaría la integridad de **entidades**.

Las siguientes tablas (1.9 y 1.10) se trata de ejemplificar que si se intentara borrar el cliente 004 y no los registros de la factura que hacen referencia al cliente en cuestión, el sistema tendrían que indicar que se esta tratando de violar la integridad referencial.

CLIENTE

Clave	Nombre	Dirección	Teléfono	Estado
004	Arredó Sobez	Cda. Del Puerto	54736725	Morelia
002	Pedro Martínez	Insurgentes	59352600	Hidalgo
001	Juana García	Churubusco	52369299	Chihuahua
003	Isabel Domínguez	Ignacio Zaragoza	23156890	Monterrey
005	Alejandro Pérez	Angela Peralta	24217880	Cuernavaca
006	José Luis Hernández	Tecamac	12155590	Puebla
007	Juan González	Moctezuma	15236898	San Luis Potosí
008	Martha Fernández	Toluca	55152813	México
009	Verónica Quintero	Emiliano Zapata	43126871	Morelia
010	Rocío Ruiz	Tlalpan	13456950	Puebla

Tabla 1.9.

FACTURA

Clave	Folio	Descripción	Cantidad	Importe
004	000001	Libreta	3	210.00
004	000002	Cartulina	100	300.00
004	000003	Bolígrafo	50	90.00
003	000004	Filtro	100	150.00
007	000003	Libreta	100	7000.00
010	000004	Bolígrafo	1	1.80
009	000005	Regla	1	100.00
009	000005	Sacapuntas	50	175.00
009	000005	Hojas	1000	45.00
008	000006	C.D.	12	144.00

Tabla 1.10.

1.2.8. MODIFICAR.

La operación modificar sirve para cambiar los valores de uno o más **atributos** en una **tupla** (o **tuplas**) de una **relación** R. Es necesario especificar una condición para los **atributos** de R a fin de seleccionar la **tupla** (o **tuplas**) que se modificarán.

La modificación de un **atributo** que no es **llave primaria** ni **llave externa**. Casi nunca causa problemas; basta con que el **SGBD** constate que el nuevo valor sea del tipo de

TESIS CON
FALLA DE ORIGEN

datos correctos y esté en el dominio. Modificar un valor de *llave primaria* es similar a eliminar una *tupla* e insertar otra en su lugar, porque usamos la *llave primaria* para identificar las *tuplas*.

1.2.9. EL ALGEBRA RELACIONAL.

Para realizar consultas y actualizar la información en una **base de datos** en una forma mas eficiente tenemos que hacer uso de las operaciones del **álgebra relacional**, estas operaciones pueden ser entre otras seleccionar, proyectar, unión, diferencia, etc., todas estas operaciones y algunas otras se encuentran en el **álgebra relacional** y en sus **operaciones de conjuntos** los que veremos a continuación.

Las operaciones del álgebra relacional suelen clasificarse en dos grupos. Uno contiene las operaciones corrientes de la teoría matemática de conjuntos; es posible aplicarlas por que las **relaciones** se definen como conjuntos de **tuplas**. Entre las operaciones de conjuntos están la **UNIÓN**, la **INTERSECCIÓN**, la **DIFERENCIA** y el **PRODUCTO CARTESIANO**. El otro grupo consiste en operaciones creadas específicamente para bases de datos relacionales; incluyen **SELECCIONAR**, **PROYECTAR** y **REUNIÓN**, entre otras.

1.2.10. SELECCIONAR.

La operación SELECCIONAR sirve para seleccionar un subconjunto de las **tuplas** de una relación que satisfacen una condición de selección.

Como ejemplo si se tienen las dos tablas anteriores con las que hemos estado trabajando, solo se necesita indicar una condición para que se extraiga la información correspondiente. Es decir si se quisiera seleccionar el registro del cliente que tenga la clave **004** el resultado sería el siguiente (tabla 1.11):

Clave	Nombre	Dirección	Teléfono	Estado
004	Alfredo López	Cda. Del Puerto	58756725	México

Tabla 1.11.

TESIS CON
FALLA DE ORIGEN

datos correctos y esté en el dominio. Modificar un valor de *llave primaria* es similar a eliminar una *tupla* e insertar otra en su lugar, porque usamos la *llave primaria* para identificar las *tuplas*.

1.2.9. EL ALGEBRA RELACIONAL.

Para realizar consultas y actualizar la información en una base de datos en una forma mas eficiente tenemos que hacer uso de las operaciones del **álgebra relacional**, estas operaciones pueden ser entre otras seleccionar, proyectar, unión, diferencia, etc., todas estas operaciones y algunas otras se encuentran en el **álgebra relacional** y en sus **operaciones de conjuntos** los que veremos a continuación.

Las operaciones del álgebra relacional suelen clasificarse en dos grupos. Uno contiene las operaciones corrientes de la teoría matemática de conjuntos; es posible aplicarlas por que las **relaciones** se definen como conjuntos de **tuplas**. Entre las operaciones de conjuntos están la **UNIÓN**, la **INTERSECCIÓN**, la **DIFERENCIA** y el **PRODUCTO CARTESIANO**. El otro grupo consiste en operaciones creadas específicamente para bases de datos relacionales; incluyen **SELECCIONAR**, **PROYECTAR** y **REUNIÓN**, entre otras.

1.2.10. SELECCIONAR.

La operación SELECCIONAR sirve para seleccionar un subconjunto de las **tuplas** de una relación que satisfacen una condición de selección.

Como ejemplo si se tienen las dos tablas anteriores con las que hemos estado trabajando, solo se necesita indicar una condición para que se extraiga la información correspondiente. Es decir si se quisiera seleccionar el registro del cliente que tenga la clave 004 el resultado sería el siguiente (tabla 1.11):

Clave	Nombre	Dirección	Teléfono	Estado
004	Alfredo López	Cda. Del Puerto	58756725	México

Tabla 1.11.

TESIS CON
FALLA DE ORIGEN

1.2.11. PROYECTAR.

Si visualizamos una relación como una tabla, la operación **SELECCIONAR** selecciona algunas filas de la tabla y desecha otras. La operación **PROYECTAR**, en cambio, selecciona ciertas columnas de la tabla y desecha las demás. Si sólo nos interesa ciertos **atributos** de una relación "proyectamos" la relación sobre esos **atributos** con la operación **PROYECTAR**.

Si se quiere trabajar con esta operación se tendría que decir que columnas se requieren, ejemplo seleccionar las columnas de Nombre, dirección y teléfono para la misma condición del ejemplo anterior, el resultado sería el siguiente:

Nombre	Dirección	Teléfono
Alfredo López	Cda. Del Puerto	58756725

Tabla 1.12.

1.2.12. OPERACIONES DE LA TEORIA DE CONJUNTOS.

El siguiente grupo de operaciones del álgebra relacional son las operaciones matemáticas normales de conjuntos. Se aplican al modelo relacional porque las relaciones se definen como conjuntos de **tuplas** de dos relaciones como conjunto.

Se utilizan varias operaciones de la teoría para combinar de diversas maneras los elementos de dos conjuntos, entre ellas, **UNIÓN**, **INTERSECCIÓN** y **DIFERENCIA**. Estas operaciones son binarias; es decir, se aplican a dos conjuntos. Al adaptar estas operaciones a las bases de datos relacionales, debemos asegurarnos que se puedan aplicar a dos relaciones para que el resultado también sea una relación válida. Para ello, las dos relaciones a las que se aplique cualquiera de las tres operaciones anteriores deberán tener el mismo tipo de **tuplas**; esta condición se denomina compatibilidad de unión. Se dice que dos relaciones $R(A_1, A_2, \dots, A_n)$ y $S(B_1, B_2, \dots, B_n)$ son compatibles con la unión si tienen el mismo grado n y si $\text{dom}(A_i) = \text{dom}(B_i)$ para $1 \leq i \leq n$. Esto significa que las dos relaciones tienen el mismo número de **atributos** y que cada par de **atributos** correspondientes tienen el mismo dominio.

Podemos definir las tres operaciones **UNIÓN**, **INTERSECCIÓN** y **DIFERENCIA** para dos relaciones compatibles con la unión, R y S , como sigue:

1.2.13. UNIÓN

El resultado de esta operación denotado por $R \cup S$, es una relación que incluye todas las **tuplas** que están en R o en S o en ambas. Las **tuplas** repetidas se eliminan.

Siguiendo con el mismo ejemplo, si se quiere la **unión** de las tablas *Cientes* y *Facturas* para el cliente con clave 003 el resultado sería el siguiente:

Folio	Nombre	Dirección	Teléfono	Estado	Descripción	Cantidad	Importe
000004	Isabel Dominguez	Ignacio Zaragoza	23156890	Monterrey	Filtro	100	150.00

Tabla 1.13.

1.2.14. INTERSECCION.

El resultado de esta operación, denotado por $R \cap S$, es una relación que incluye todas las **tuplas** que están tanto en R como S.

Si queremos una **intersección** el resultado sería el siguiente:

Clave
003

Tabla 1.14.

1.2.15. DIFERENCIA.

El resultado de esta operación, denotado por $R - S$, es una relación que incluye todas las **tuplas** que están en R pero no en S.

Al solicitar una diferencia de *Cientes* - *Facturas* el resultado sería el siguiente:

Nombre	Dirección	Teléfono	Estado
Isabel Dominguez	Ignacio Zaragoza	23156890	Monterrey

Tabla 1.15.

El **PRODUCTO CARTESIANO** crea **tuplas** con los **atributos** combinados de dos relaciones. Después podremos seleccionar sólo las **tuplas** relacionadas de las dos relaciones especificando una condición de selección apropiada.

Como esta secuencia de **PRODUCTO CARTESIANO** seguido de **seleccionar** se utiliza con mucha frecuencia para identificar y seleccionar **tuplas** relacionadas de dos relaciones, se creó una operación especial, llamada **REUNIÓN**, con el fin de especificar esta secuencia con una sola operación. El **PRODUCTO CARTESIANO** casi nunca se usa como operación significativa por sí sola.

1.2.16. REUNIÓN.

La operación **REUNIÓN**, denotada por **&**, sirve para combinar **tuplas** relacionada de dos relaciones en una sola **tupla**. Esta operación es muy importante en cualquier base de datos relacional que comprenda más de una relación, porque permite procesar los vínculos entre las relaciones.

La **REUNIÓN** más común implica condiciones de reunión con comparaciones, de igualdad exclusivamente. Una **REUNIÓN** así, en la que el único operador de comparación empleado es **=**. Se llama **EQUIRREUNIÓN**. Observe que en el resultado de una **EQUIRREUNIÓN** siempre tenemos uno o más pares de **atributos** con valores idénticos en todas las **tuplas**.

Si tenemos las siguientes tablas: (1.16 y 1.17)

CLIENTE

Clave	Nombre	Dirección	Teléfono	Estado
004	Alfredo López	Cda. Del Puerto	58756725	México
002	Pedro Martínez	Insurgentes	59352600	Hidalgo
001	Juana García	Churubusco	52369299	Chihuahua
003	Isabel Domínguez	Ignacio Zaragoza	23156890	Monterrey
005	Alejandro Pérez	Angela Peralta	24217880	Cuernavaca
006	José Luis Hernández	Tecamac	12155590	Puebla
007	Juan González	Moctezuma	15236898	San Luis Potosí
008	Martha Fernández	Toluca	55152813	México
009	Verónica Quintero	Emiliano Zapata	43126871	Morelia
010	Rocío Ruiz	Tlalpan	13456950	Puebla

Tabla 1.16.

FACTURA

Clave	Folio	Descripción	Cantidad	Importe
004	000001	Libreta	3	210.00
004	000002	Cartulina	100	300.00
004	000003	Bolígrafo	50	90.00
003	000004	Fieltro	100	150.00
007	000003	Libreta	100	7000.00
010	000004	Bolígrafo	1	1.80
009	000005	Regla	1	100.00
009	000005	Sacapuntas	50	175.00
009	000005	Hojas	1000	45.00
008	000006	C.D.	12	144.00

Tabla 1.17.

Si queremos la *reunión* de las dos tablas con la condición de que la clave del cliente sea 004, entonces el resultado será el mostrado en la tabla 1.18:

Clave	Nombre	Dirección	Teléfono	Estado	Folio	Descripción	Cantidad	Importe
004	Alfredo López	Cda. Del Puerto	58756725	México	000001	Libreta	3	210.00
004	Alfredo López	Cda. Del Puerto	58756725	México	000002	Cartulina	100	300.00
004	Alfredo López	Cda. Del Puerto	58756725	México	000003	Bolígrafo	50	90.00

Tabla 1.18.

Cabe señalar que, si ninguna combinación de *tuplas* satisface la condición de reunión, el resultado de una *REUNIÓN* es una relación vacía con *cero tuplas*.

Se ha demostrado que el conjunto de operaciones del álgebra relacional $\{\sigma, \pi, \cup, -, \times\}$ es un conjunto completo es decir, cualquiera de las operaciones del álgebra relacional se puede expresar como una secuencia de operaciones de este conjunto. Por ejemplo, la operación *INTERSECCION* se puede expresar empleando *UNIÓN* y *DIFERENCIA*.

Aunque, en términos estrictos, la *INTERSECCIÓN* no es indispensable, resulta poco cómodo especificar esta expresión compleja cada vez que deseamos especificar una intersección.

De manera similar, una *REUNIÓN* natural se puede especificar como un *PRODUCTO CARTESIANO* seguido de operaciones *SELECCIONAR* y *PROYECTAR*. Así pues, las diversas operaciones de *REUNION* tampoco son estrictamente necesarias para el poder expresivo del álgebra relacional; sin embargo, son muy importantes porque son cómodas y se emplean con mucha recurrencia en las aplicaciones de bases de datos.

Algunas solicitudes que se hacen comúnmente a las bases de datos no se pueden atender con las operaciones estándar del álgebra relacional. Por lo que se tienen que utilizar **Funciones agregadas**.

El primer tipo de solicitud que no se puede expresar en el álgebra relacional es la especificación de funciones agregadas matemáticas sobre colecciones de valores de la base de datos. Otro tipo de solicitud que se hace con frecuencia implica agrupar las **tuplas** de una relación según el valor de algunos de sus **atributos** y después aplicar una función agregada independientemente a cada grupo. Vale la pena subrayar que el resultado de aplicar una función agregada es una relación, no un número escalar, aunque tenga un solo valor.

1.2.17. OPERACIONES DE CERRADURA RECURSIVA.

Otro tipo de operación que, en general, no puede especificarse en el álgebra relacional es la cerradura recursiva (a veces llamada cierre recursivo). Esta operación se aplica a un vínculo recursivo entre **tuplas** del mismo tipo, como el vínculo entre un empleado y un supervisor.

REUNIÓN EXTERNA Y UNIÓN EXTERNA.

Las operaciones de **REUNIÓN** antes descritas seleccionan **tuplas** que satisfacen la condición de reunión. Por ejemplo, con una operación de **REUNIÓN NATURAL** $R * S$, sólo las **tuplas** de R que tienen **tuplas** coincidentes en S – y viceversa – aparecen en el resultado. Por tanto, las **tuplas** sin una “**tupla** relacionada” se eliminan del resultado, y lo mismo sucede con las **tuplas** que tiene nulo en los **atributos** de **reunión**. Podemos usar un conjunto de operaciones, llamadas **REUNIONES EXTERNAS**, cuando queremos conservar en el resultado todas las **tuplas** que están en R o en S , o en ambas, ya sea que tengan o no **tuplas** coincidentes en la otra relación.

La operación de **UNIÓN EXTERNA** se creó para efectuar la **unión** de **tuplas** de dos relaciones que no son compatibles con la **unión**. Esta operación efectuará la **UNIÓN** de **tuplas** de dos **relaciones** que sean parcialmente compatibles, lo que significa que sólo algunos de sus **atributos** son compatibles con la **unión**. Otra capacidad que tiene casi todos los lenguajes comerciales es la de especificar operaciones con los valores después de extraerlos de la base de datos.

1.3. UTILIZACION DE LA RED.

Para que funcione un sistema de información remotamente, tiene que contar con una forma de conexión entre cada equipo que va a estar en operación con este sistema, por lo que es necesario conocer los conceptos básicos de redes.

Una **red** es la interconexión de computadoras con el objeto de compartir recursos, como son: programas y componentes de hardware.

Los tres tipos de redes mas conocidas son **LAN**, **MAN** y **WAN**, los que describiremos a continuación.

1.3.1. RED DE AREA LOCAL (LAN)

Una red de área local (**LAN**: Local Area Network) es la interconexión de dispositivos de cómputo que pueden comunicarse entre sí y compartir un grupo de recursos comunes, como impresoras, discos, etc. Normalmente, están limitadas en distancia (5 km), por lo que pueden abarcar desde un departamento hasta un edificio, o todo un campus universitario. En general, el hecho de trabajar dentro de una red de área local es sencillo y garantiza accesos seguros a quienes se encuentran interconectados.

Las primeras redes que se instalaron en algunas compañías, incluyendo IBM, Honeywell y Digital Equipment Corporation (DEC), tenían sus propios estándares que definían la forma de conectar las computadoras entre sí. Estos estándares instauraron los mecanismos necesarios para poder transmitir información de una computadora a otra. Estas primeras especificaciones no fueron enteramente compatibles entre ellas; por ejemplo las redes conectadas a la arquitectura IBM denominada SNA (Systems Network Architecture).

Posteriormente, las organizaciones dedicadas a la creación de estándares, incluyendo la Organización de Estándares Internacionales (ISO: International Standards Organization) y el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE: Institute of Electrical and Electronic Engineers), desarrollaron modelos que fueron reconocidos y aceptados internacionalmente como estándares para el diseño de cualquier red de computadoras.

Utilidad de las redes de área local.

Las redes de área local son muy importantes dentro de las diferentes organizaciones ya que forman parte indispensable de la productividad de las personas. Las redes de área local han evolucionado ya que se tiene contacto con ellas todos los días, cada vez

1.3. UTILIZACION DE LA RED.

Para que funcione un sistema de información remotamente, tiene que contar con una forma de conexión entre cada equipo que va a estar en operación con este sistema, por lo que es necesario conocer los conceptos básicos de redes.

Una **red** es la interconexión de computadoras con el objeto de compartir recursos, como son: programas y componentes de hardware.

Los tres tipos de redes mas conocidas son **LAN**, **MAN** y **WAN**, los que describiremos a continuación.

1.3.1. RED DE AREA LOCAL (LAN)

Una red de área local (**LAN**: Local Area Network) es la interconexión de dispositivos de cómputo que pueden comunicarse entre sí y compartir un grupo de recursos comunes, como impresoras, discos, etc. Normalmente, están limitadas en distancia (5 km), por lo que pueden abarcar desde un departamento hasta un edificio, o todo un campus universitario. En general, el hecho de trabajar dentro de una red de área local es sencillo y garantiza accesos seguros a quienes se encuentran interconectados.

Las primeras redes que se instalaron en algunas compañías, incluyendo IBM, Honeywell y Digital Equipment Corporation (DEC), tenían sus propios estándares que definían la forma de conectar las computadoras entre sí. Estos estándares instauraron los mecanismos necesarios para poder transmitir información de una computadora a otra. Estas primeras especificaciones no fueron enteramente compatibles entre ellas; por ejemplo las redes conectadas a la arquitectura IBM denominada SNA (Systems Network Architecture).

Posteriormente, las organizaciones dedicadas a la creación de estándares, incluyendo la Organización de Estándares Internacionales (ISO: International Standards Organization) y el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE: Institute of Electrical and Electronic Engineers), desarrollaron modelos que fueron reconocidos y aceptados internacionalmente como estándares para el diseño de cualquier red de computadoras.

Utilidad de las redes de área local.

Las redes de área local son muy importantes dentro de las diferentes organizaciones ya que forman parte indispensable de la productividad de las personas. Las redes de área local han evolucionado ya que se tiene contacto con ellas todos los días, cada vez

con más frecuencia, por ejemplo, las redes de los supermercados, donde todas las cajas están organizadas dentro de una red de área local y gracias a esta tecnología pueden mantenerse bases de datos centralizadas de precios e inventarios; así, al momento de digitar la clave del producto o bien de leer el código de barras, se extrae el precio de la base de datos central y se descuenta la mercancía del almacén, lo que ayuda enormemente al manejo de los inventarios de dichas cadenas de negocios a través del uso de bases de datos centralizadas.

Las redes de área local no son para uso exclusivo de las empresas de tecnología; también se utilizan en agencias de viajes, bancos, casas de bolsa, aerolíneas y muchos otro tipos de negocios. Con el uso de las redes, la productividad de la gente se incrementa, entre otras cosas porque los servicios que utilizan todos los días (facturación, inventarios, etc.) tienen una localización central, lo que garantiza que dichos servicios estén disponibles en el momento que se requieran.

El hecho de compartir recursos dentro de la red se evitan gastos que de otro modo se emplearían en la compra de equipo como impresoras, faxes, unidades de almacenamiento y unidades de CD-ROM, el cual sería para un solo usuario. Por todo lo anterior, el uso de los recursos se optimiza y se adquiere solamente lo que el negocio necesita y no lo que los usuarios requieran de forma individual.

Para utilizar un solo programa para una aplicación específica en toda la organización y ayudar a la estandarización del software, el único camino disponible es el uso de las redes de área local. De esta manera, la paquetería que se encuentra en la red, y no la que los usuarios escojan, es la única que podrá ser usada, lo que ayuda a las empresas en el ahorro que se genera por la estandarización recursos de capacitación y a contar con un pequeño grupo de gente cuya función principal sea dar soporte en dicha paquetería a toda la organización.

1.3.2. RED METROPOLITANA (MAN)

La tecnología **MAN** es la interconexión de redes locales que están distribuidas en una área específica, como un campus un polígono industrial o una ciudad.

1.3.3. RED DE AREA AMPLIA (WAN)

La tecnología **WAN** a veces llamada Long Haul Network (Redes de Gran Alcance), proporciona comunicación que cubre gran distancia. Este tipo de redes posee organizaciones distribuidas que utilizan diversos mecanismos de transmisión de datos contando con elementos de control que permiten integrar a todos los elementos y

direccionar entre nodo y nodo de la red. Muchas tecnologías **WAN** no tienen límite de distancia de recorrido; una **WAN** puede permitir que dos puntos inmediatamente lejanos se comuniquen. Por ejemplo una **WAN** puede recorrer un continente o unir computadoras a través de un océano.

Por lo común las **WAN** operan más lentamente que las **LAN** y tienen tiempo de retraso mucho mayores entre las conexiones. La velocidad normal para una **WAN** llega a un rango que va de los 56 Kbps a 155 Mbps (Millones de Bits Por Segundo). Los retrasos para una **WAN** pueden variar de unos cuantos milisegundos a varias decenas de segundos.

1.3.4. TOPOLOGIAS DE REDES.

A la forma en que se conectan las computadoras en una red se llama topología. Actualmente existe una gran variedad de topologías, como son la topología en bus, en estrella, en anillo; y en el caso de redes complejas, topologías mixtas o híbridas, dependiendo de la flexibilidad y/o complejidad que se quiera dar al diseño.

TIPOS DE CONEXIÓN.

Existen dos tipos de conexión a una red: la conexión punto a punto y la conexión multipunto.

Punto a punto. Una conexión punto a punto es una conexión de dos dispositivos. Por ejemplo, una conexión de dos computadoras mediante fibra óptica o par trenzado (twisted pair).

Multipunto. Una conexión multipunto utiliza un solo cable para conectar más de dos dispositivos. Por ejemplo, un cable que tiene varios dispositivos conectados al mismo medio de transmisión. En la figura 1.4 se puede apreciar la conexión multipunto.



Figura 1.4 Conexión multipunto.

TESIS CON
FALLA DE ORIGEN

1.3.4.1. TOPOLOGÍA EN BUS

La topología en BUS es una topología de red multipunto, en la cual los dispositivos se conectan a un mismo cable, uno tras otro.

En la topología en BUS, todos los dispositivos comparten el mismo medio, los mensajes que se transmiten a través de este medio son atendidos por todos los demás dispositivos que lo comparten.

La topología en BUS se considera como una carretera por la que transitan todos los vehiculos (paquetes o tramas) y que está limitada en distancia, dependiendo del tipo de cable y los conectores que se utilicen. Los conectores son resistencias que sirven para mantener constantes la impedancia del cable para poder transmitir la información.

En la figura 1.5 se puede apreciar lo indicado en párrafos anteriores:

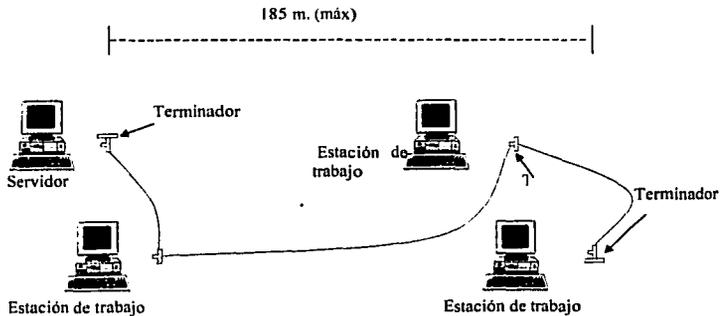


Figura 1.5 Topología en BUS con cable coaxial delgado (thin coax)

En la siguiente tabla (1.19) se indican las ventajas y las desventajas de la topología en BUS.

Ventajas	Desventajas
La falla en una computadora no afecta a la red	Frágil. Si el cable se desconecta o se troza, la red deja de funcionar en su totalidad por pérdida de impedancia
Las conexiones a la red son sencillas y flexibles.	Difícil de aislar cuando hay problemas de cableado.
Es una topología barata en cuestión de cable, conectores T y terminadores.	Degradación del desempeño de la red con el crecimiento de dispositivos.

Tabla 1.19. Ventajas y desventajas de la topología BUS.

1.3.4.2. TOPOLOGÍA EN ANILLO

La topología en anillo es una red punto a punto donde los dispositivos se conectan en un círculo irrompible formado por un concentrador, que es el encargado de formar eléctricamente el anillo en la medida en que se insertan los dispositivos. En la topología en anillo, el mensaje viaja en una sola dirección y es leído por cada una de las computadoras individualmente y retransmitido al anillo en caso de no ser el destinatario final del mensaje.

Esta topología se usa generalmente Token Ring y Token Passing, en donde token (testigo) da a cada estación la oportunidad de transmitir, cuando el token es liberado, pasa a la siguiente computadora que desee transmitir, y así sucesivamente.

No se sabe que haya un número máximo de dispositivos conectados en este tipo de topología debido a que no se comparte el medio como en el caso de la topología en BUS.

La topología en anillo se puede apreciar mejor en el siguiente diagrama mostrado en la figura 1.6.

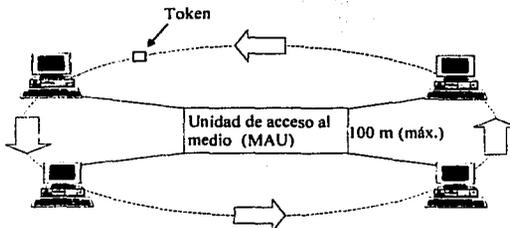


Figura 1.6: Topología en anillo

En la tabla 1.20 se pueden comparar las ventajas y desventajas de la topología en anillo.

Ventajas	Desventajas
Si el cable de un dispositivo falla, no afecta la integridad del anillo.	Un alto costo en el cableado y las conexiones, así como en el concentrador.
Igualdad de acceso a todos los dispositivos.	Si el concentrador falla, el anillo se rompe.
El desempeño de la red está garantizado.	

Tabla 1.20 Ventajas y desventajas de la topología en anillo.

1.3.4.3. TOPOLOGÍA EN ESTRELLA.

La topología en estrella es una topología en red punto a punto, ya que los dispositivos se encuentran conectados a un concentrador. Generalmente se le denomina topología de concentradores.

La topología en estrella concentra a todos los dispositivos en una estación centralizada que enruta el tráfico al lugar apropiado. Tradicionalmente, esta topología es un acercamiento a la interconexión de dispositivos en la que cada dispositivo se conecta por un circuito separado a través del concentrador.

TESIS CON
FALLA DE ORIGEN

Esta topología es similar a la red de teléfonos, en donde existe un conmutador (PBX) y cada llamada que se hace tiene que pasar por el PBX para poder llegar a su destino.

La figura 1.7 nos muestra mas explícitamente esta topología.

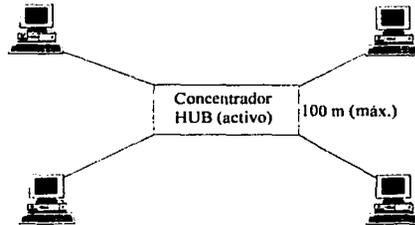


Figura 1.7: Topología en estrella

No existe un número máximo de conexiones debido a que los concentradores son cada vez más poderosos y soportan mayor número de dispositivos con un nivel de servicio muy alto. En general, el número de estaciones que se pueden conectare al concentrador depende del tráfico que se genere entre ellas, y cuando éste es excesivo la red se divide mediante un dispositivo adicional cuya función es aislar el tráfico de un segmento al otro.

Tipos de acceso

La topología en estrella y anillo físicamente tienen forma de estrella, pero dependiendo del concentrador que se instale permanecen con esta forma o se genera un anillo. En este caso existen dos formas de comunicar los dispositivos con el concentrador o estación controladora de la topología: poleo y contención

El tipo de **acceso de poleo** consiste en contar con una estación, la cual es la encargada de asignar permisos a cada dispositivo dentro del segmento; es decir, si el dispositivo tiene permiso de enviar su información, éste comienza su transferencia a su destinatario, de lo contrario tiene que esperar su turno. Cada dispositivo tiene una cantidad de tiempo igual a los demás, por lo que existe igualdad de acceso al medio. En este tipo de acceso no se puede enviar información si no se tiene el permiso para hacerlo.

En el tipo de **acceso de contención** cada dispositivo envía su información sólo cuando nadie en la red está enviando información; es decir, *sólo un dispositivo a la vez puede enviar información*, y el concentrador es el encargado de administrar el tráfico y enrutarlo de la mejor manera posible. Este tipo de acceso permite un mayor número de paquetes y mejor rendimiento en l red.

En la tabla 1.21 se aprecian las ventajas y desventajas de fia topología en estrella.

Ventajas	Desventajas
Si el cable de un dispositivo falla no afecta la integridad de la red.	Alto costo en el cableado, las conexiones y el concentrador
Facilidad para añadir nuevos dispositivos.	Si el concentrador falla, la red entera deja de funcionar.
Administración y monitoreo centralizado.	

Tabla 1.21. Ventajas y desventajas de la topología en estrella.

1.3.4.4. TOPOLOGÍA HÍBRIDA.

La topología híbrida es el conjunto de todas las anteriores. Su implementación se debe a la complejidad de la solución de red, o bien al aumento en el número de dispositivos, lo que hace necesaria establecer una topología de este tipo. Las topologías híbridas tienen un costo elevado debido a su administración y mantenimiento, ya que cuentan con segmentos de diferentes tipos, lo que obliga a invertir en equipo adicional para lograr la conectividad deseada.

En las siguientes figuras se intenta demostrar como se tendría que conectar para que fuera una topología híbrida.

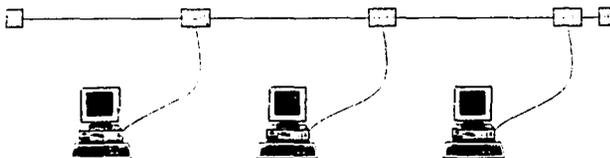


Figura 1.8. (Conexión multipunto)



Figura 1.9. (Topología en BUS)

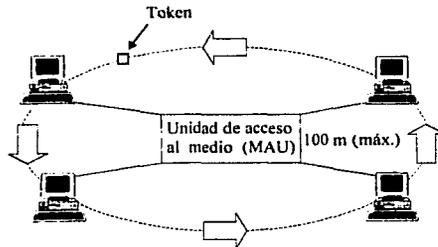


Figura 1.10. (Topología en anillo).



Figura 1.11. (Topología en estrella).

TESIS CON FALLA DE ORIGEN

1.3.5. PROTOCOLOS DE COMUNICACIÓN.

Un protocolo es el conjunto de normas o procedimiento necesario para iniciar y mantener una comunicación.

En este caso de las redes de computadora, un protocolo es el conjunto de normas que permiten que dos o más computadoras puedan comunicarse. El protocolo consta de una sintaxis, una semántica y un tiempo. La sintaxis en un protocolo define los conjuntos de bits (una serie de unos o ceros) divididos en campos. Por ejemplo, los primeros 48 bits son la dirección fuente y los siguientes 48 son la dirección destino. La semántica define el significado exacto de los bits dentro del campo. Por ejemplo, una dirección de 48 bits iguales (unos), significa que es una dirección broadcast; es decir, que puede ser leída por todos los computadores de la red. El tiempo define la relación entre el rango de los bits dentro de los campos y las pausas entre reconocimientos de los mismos.

Para el desarrollo de nuestro sistema nos basaremos en el protocolo TCP/IP.

TCP/IP

TCP/IP (Transmission Control Protocol/Internet Protocol) es probablemente uno de los protocolos de comunicaciones más antiguos en los estándares de interredes (redes interconectadas) TCP/IP se desarrolló por la Agencia de Proyectos de Investigación Avanzada de la Defensa de los Estados Unidos de América (DARPA: Defense's Advanced Research Project Agency), DARPA convierte todos los viejos sistemas ARPANET para correr o ejecutar los protocolos de Internet.

El protocolo de comunicaciones es flexible y permite la transmisión de tramas sin errores entre diferentes sistemas, ha estado funcionando desde nace más de 15 años. Debido a que este protocolo de transferencia de información, puede enviar grandes volúmenes de información a través de redes no confiables, garantizando que ésta será recibida sin error al momento de alcanzar su destino final.

Cuando se utiliza TCP/IP, la información viaja en segmentos creado por TCP entre emisor y receptor para poder acceder a alguna aplicación. Los segmentos creados por TCP son encapsulados en IP, a esta encapsulación se llama datagrama IP. El datagrama IP permite que los segmentos TCP que fueron creados por alguna aplicación puedan ser transmitidos o enrutados en la red de área local o en la red de área extendida.

Las redes TCP/IP permiten que la información puede enviarse de un sistema a otro, sin que éstos tengan que ser del mismo fabricante; por ejemplo, una estación con Windows NT, de Microsoft, puede intercambiar información con una computadora con Pathworks, de Digital, siempre y cuando utilicen el mismo protocolo de comunicaciones, que en este caso es TCP/IP.

MICROSOFT WINDOWS NT.

Windows NT es el último eslabón en una cadena de evolución de Microsoft orientados a los sistemas operativos de red. En él se encuentran aspectos de varios de ellos resumidas en un sistema operativo robusto y rico en características potentes.

HISTORIA.

La influencia más obvia es de la línea de productos de Windows, particularmente de Windows para trabajo en Grupo 3.11, Windows 95 y últimamente Windows 98 y después de rato descubrir que no se trataba de Windows sino de alguna de las versiones tradicionales Windows se adaptan fácilmente al entorno NT.

El modelo de redes de Windows NT ha evolucionado desde dos líneas de productos: LAN Manager y Windows par Trabajo en Grupo. De LAN Manager heredó la capacidad de administrar redes con varios servidores usando el concepto de Dominio. De Windows para Trabajo en Grupo se extrajo la característica de la comparación de recursos sobre los que los usuarios pueden viajar para conectarse.

Adicionalmente se unen en Windows NT todas las experiencias de Microsoft en el desarrollo de otros productos como OS/2 (desarrollo en el cual participaron para la primeras dos versiones en equipo con IBM) y la versión UNIX denominada XENIX en los años 80 para computadoras basadas en el 80286.

Windows NT Workstation y Windows NT Server

Existen dos versiones de Windows NT: Windows NT Workstation versión 4.0 (actualmente se consiguen las versiones beta 5.0). Ambas versiones son idénticas excepto por el hecho de que Windows NT Server incluye características que le permite comportarse como un sistema operativo versátil de red.

La unión de las dos versiones en realidad produce efectos sinérgicos. Inclusive Windows NT Workstation puede jugar papeles importantes en las redes al convertirse por ejemplo en servidor de impresión o como un enrutador TCP/IP. Por supuesto las mejores computadoras cliente para una red basada en Windows NT Server serán aquellas que estén corriendo en Windows NT.

Características del Sistema Operativo Windows NT

Portabilidad. Debido a que el núcleo central de Windows NT está escrito en lenguaje C, este sistema operativo se convierte en uno de los más fácilmente transportables los sistemas disponibles actualmente. Significa esto que Windows NT se puede instalar en máquinas diferentes: computadoras basadas en Intel x86, computadoras de arquitectura MIPS RISC e inclusive máquinas Digital Alpha AXP RISC.

Gran Capacidad de Almacenamiento.

Este sistema se diseñó para sacar máximo provecho de los procesadores 80386 y posteriores de Intel. Es por ello que Windows NT soporta hasta 4 GB de memoria RAM.

Multitarea y Multithreading.

La multitarea es un proceso simulado. Cuando una computadora pareciera que está ejecutando dos o tres trabajos simultáneamente en realidad no lo está haciendo. Todo lo que ocurre es que el sistema operativo se encarga de repartir el tiempo entre los trabajos a ejecutar: un primer momento para ejecutar la tarea A, luego otro momento para que la B haga otro poco y un tercero momento para la tarea C antes de volverle a dar tiempo a la tarea A.

Si se ejecuta rápidamente este proceso será transparente y dará la sensación al usuario de verdadera multitarea. Incluso en DOS que no fue creado originalmente para ello, se podría simular este efecto. El problema radica en el hecho de que si alguna de las tareas llega a fallar podría interrumpir todo el proceso de la máquina y por ende perjudicaría a las demás tareas.

Windows NT no tiene este problema. Windows NT puede aislar las tareas de tal suerte que ninguna interfiera a las demás. Incluso si alguna se bloqueara, el usuario siempre tendrá la opción de matarla (kill) para que las demás continúen normalmente.

Multitareas es la ilusión de hacer varias cosas al mismo tiempo. Multithreading se aprovecha de las ventajas de algunos procesadores que pueden ejecutar dos o más tareas (llamadas threads) en áreas separadas de la CPU. Muchos procesadores RISC y el procesador Pentium de Intel tienen esta facultad.

Windows NT permite que una aplicación pueda ejecutar sus propios threads. De esta manera un proceso complejo se podría separar en varias subtareas que se ejecutarían simultáneamente y luego se reunirían para producir un resultado.

Soporte de multiprocesadores.

Cuando un sistema operativo soporta Multithreading entonces tiene el potencial de usar más de un procesador. La capacidad que se genera para una computadora se denomina escalabilidad. Significa que si se tiene un servidor de red con varios procesadores, la red estará en capacidad de crecer de acuerdo a las necesidades.

Al principio un procesador podrá llevar toda la carga del trabajo pero más adelante, con el ingreso de más usuarios a la red, el servidor podría disminuir su rendimiento y requerir en ese momento de otra CPU que apoye.

Existen dos tipos de multiprocesamiento: multiprocesamiento asimétrico se asignan diferentes tareas a diferentes procesadores. Esta es una función relativamente fácil de implementar pero podría provocar que algunos procesadores estuviesen más

ocupados que otros. En el procesamiento simétrico cualquier procesador. Es un poco más difícil de lograr pero se obtiene el máximo rendimiento de todos los procesadores.

Sistema de archivos de NT (NTFS) Windows NT trajo consigo un nuevo sistema de archivos llamando NTFS (NT File System), que mejora la eficiencia de las redes así como su confiabilidad y seguridad. En DOS existía un mecanismo llamado FAT así como en OS/2 existe HPFS. Estas fórmulas no estaban diseñadas para redes por dos razones : primero, no estaban diseñadas para resolver en forma dinámica cualquier problema del disco y segundo, no podían impedir que un intruso accediera a los datos guardados en el disco.

El método NTFS compensa las fallas del disco redirigiendo los datos de los sectores malos del disco a un sector substituto. Adicionalmente, NTFS impide que un intruso, por ejemplo, acceda al disco arrancado la máquina desde disquete con algún sistema operativo. El sistema NTFS obliga a que los archivos se puedan leer o modificar únicamente obteniendo un acceso usando un nombre y una contraseña.

Seguridad. La seguridad en Windows NT Workstation está basada en subdirectorios. Normalmente se le asigna a una estructura de directorios el mismo conjunto de restricciones de acceso.

En Windows NT Server, por el contrario, las restricciones pueden ser mucho más detalladas. El administrador de la red puede asignarle a cada usuario los recursos a los que le está permitiendo acceder y se le puede restringir inclusive hasta el nivel de archivo.

Windows NT tiene otras facultades de seguridad interesantes como el hecho de permitirle a un usuario especial poder sacar un backup de toda la información de un servidor sin necesidad de darle atributos de administrador. De esta forma alguien diferente del administrador podrá encargarse de la tarea de sacar backups de soporte sin requerir de un acceso completo al sistema.

Cuando se trata de ingresar a un sistema NT, el acceso se logra presionando primero la combinación de teclas CTRL-ALT-DEL. Este mecanismo fue diseñado para evitar que programas falsos simulen la pantalla de ingreso tratando de robarse el nombre de los usuarios y sus contraseñas. Por supuesto ninguna aplicación ni ningún usuario podrán impedir que la combinación de teclas CTRL-ALT-DEL produzcan el efecto para el que fueron programadas.

Windows NT incluye los siguientes protocolos

- TC/IP
- NEWLink IPX/SPX
- NetBEUI

Los protocolos se pueden añadir, borrar o configurar usando la red en el panel de control.

1.4. CLIENTE/SERVIDOR

La plataforma que encontramos mas apta para un sistema de información como el que nos ocupa (Agencia de Viajes) es el de Cliente/Servidor. Y entendemos por **Cliente/Servidor** a las entidades lógicas independientes que operan en conjunto a través de una red para realizar una tarea.

Las características que diferencian a **cliente servidor** de otras plataformas son las siguientes:

- **Servicios:** **Cliente/Servidor** es una relación entre procesos ejecutados en aparatos distintos. El proceso del **servidor** hace de este un proveedor de servicios y el **cliente** es un consumidor de servicios.
- **Recursos compartidos:** Un **servidor** puede atender a muchos **clientes** al mismo tiempo y a regular recursos compartidos.
- **Protocolos asimétricos:** Entre el **cliente** y el **servidor** se establece una relación de "Muchos a Uno", esto quiere decir que para muchos **clientes** un **servidor**.
- **Transparencia de ubicación:** El **servidor** es un proceso que puede residir en el mismo aparato que el **cliente** o en un aparato distinto en una red. Un programa puede ser un **cliente**, un **servidor** o ambos.
- **Mezcla e igualdad:** El software del **cliente/servidor** es independiente al hardware o de las plataformas del sistema operativo, se puede mezclar e igualar plataformas de **cliente/servidor**.
- **Intercambios basados en mensajes:** El **cliente/servidor** son sistemas que interactúan a través de un mecanismo de transmisión de mensajes. El mensaje es el medio de entrega para las solicitudes y respuestas de servicio.
- **Facilidad de escalabilidad:** Los sistemas **cliente/servidor** pueden escalarse horizontal o verticalmente. La escalabilidad horizontal consiste en la adición o eliminación de estaciones de trabajo de **clientes**. La escalabilidad verticales migrar un aparato **servidor** más grande y más veloz o a **servidores** múltiples

- **Integridad:** El código del *servidor* y los datos del *servidor* se encuentran **centralmente**, a esto se le puede dar un mantenimiento de menor costo y en la protección de la integridad de los datos compartidos.

Los modelos de *cliente/servidor* pueden distinguirse uno de otros por el servicio que prestan:

1.4.1. SERVIDORES DE ARCHIVOS.

Se trata de una modalidad que requiere numerosos intercambios de mensajes a través de la red para obtener los datos solicitados (Figura 1.12). Son útiles para compartir archivo por medio de una red. Son indispensables para la creación de depósitos compartidos de documentos, imágenes, planos de ingeniería y otros objetos de datos de grandes dimensiones.

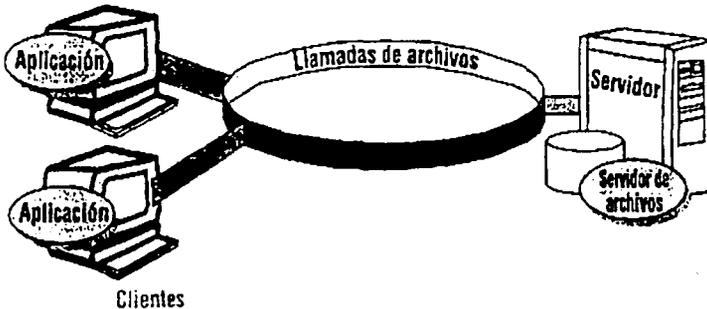


Figura 1.12. 1

1.4.2. SERVIDORES DE BASES DE DATOS.

El *cliente* envía solicitudes de SQL en calidad de mensajes al *servidor* de base de datos (Figura 1.13). El código que procesa la solicitud de SQL y los datos encuentran en la misma máquina. El *servidor* hace uso de su propia capacidad de procesamiento para encontrar los datos solicitados y envía los datos requeridos por el *cliente* (a diferencia del *servidor* de archivo al cual se le envían todos los datos para que éste encuentre sus propios datos).



Figura 1.13.¹

1.4.3. SERVIDORES DE TRANSACCIONES.

Con un *servidor de transacciones*, el *cliente* invoca procedimientos remotos que residen en el *servidor* con un mecanismo de base de datos de SQL (Figura 1.14). El intercambio por la red consiste en un solo mensaje de solicitud/respuesta (a diferencia de lo que ocurre con el *servidor* de bases de datos, en cuyo caso es necesario un mensaje de solicitud/respuesta para cada instrucción de SQL en una transacción). Los enunciados de SQL aciertan o fallan todos como una sola unidad. A estas instrucciones de SQL agrupadas se les conoce como transacciones.

¹ Visual Basic Programación Cliente/Servidor. Alfonso Gonzalez. Alfaomega Ra-Ma.

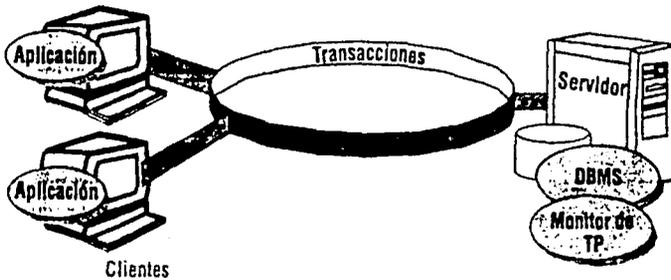


Figura 1.14. 1

Como *servidor de transacciones*, uno mismo crea la aplicación *cliente/servidor* generando el código tanto del *cliente* como de *servidor*. El *cliente* suele incluir una interfaz gráfica de usuario (GUI: graphical user interface). El *servidor* consiste por lo general en transacciones de SQL contra una base de datos, a esas aplicaciones se les llama procesamiento de transacciones en línea (OLTP: online transaction processing). Las aplicaciones OLTP requieren de controles de seguridad e integridad de la base de datos.

1.4.4. SERVIDORES DE GROUPWARE.

El groupware dirige la administración de información semiestructurada como texto, imagen, correo, tableros de avisos y flujo de trabajo, estableciendo un contacto directo entre personas.

Software de groupware especializado puede integrarse a paquetes de API de *cliente/servidor* de un proveedor, en la mayoría de los casos, se crean aplicaciones usando un lenguaje de creación de scripts e interfaces basadas en formas ofrecidos por proveedores. El middleware de comunicación entre el *cliente* y el *servidor* es propio de cada distribuidor (Figura 1.15).

TESIS CON
FALLA DE ORIGEN



Figura 1.15. 1

1.4.5. SERVIDORES DE OBJETOS.

La aplicación *cliente/servidor* se genera como un conjunto de objetos de comunicación (Figura 1.16). Los objetos del *cliente* se comunican con los objetos del *servidor* mediante un corredor de solicitudes de objetos (ORB: object request broker). El *cliente* invoca un método de un objeto remoto. El ORB localiza una instancia de esa clase de *servidor* de objetos, invoca el método solicitado y envía los resultados al objeto del *cliente*. Los objetos del *servidor* deben ofrecer soporte de concurrencia y participación.



Figura 1.16. 1

TESIS CON
FALLA DE ORIGEN

1.4.6. SERVIDOR WEB.

Este nuevo modelo de *cliente/servidor* esta integrado por *clientes* "universales", compactos, portátiles, en comunicación con *servidores* superamplios. Un *servidor* Web envía documentos cuando los *clientes* los solicitan por nombre por su nombre (Figura 1.17), *clientes* y *servidores* se comunican mediante protocolos semejantes a RPC denominado HTTP. Este protocolo define un conjunto simple de órdenes; los parámetros se transmiten en cadenas, sin estipulaciones de datos tecleados.

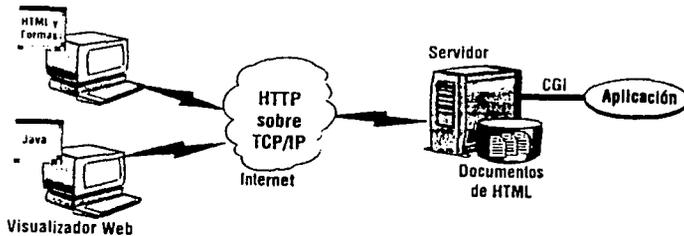


Figura 1.17. ¹

1.4.7. SERVIDORES AMPLIOS Y CLIENTES GRANDES.

Las aplicaciones *cliente/servidor* pueden distinguirse unos de otros por la forma en que la aplicación distribuida se divide entre el *cliente* y el *servidor* (Figura 1.18). El modelo del *servidor* amplio se le atribuyen más funciones al *servidor* y en el modelo de *cliente* grande se procede a la inversa. Los *servidores* de groupware, transacciones y *servidores* Web son ejemplos de *servidores* amplios; los *servidores* de base de datos y archivo son ejemplos de *clientes* grandes. Los objetos distribuidos pueden ser de ambas clases.

TESIS CON
FALLA DE ORIGEN

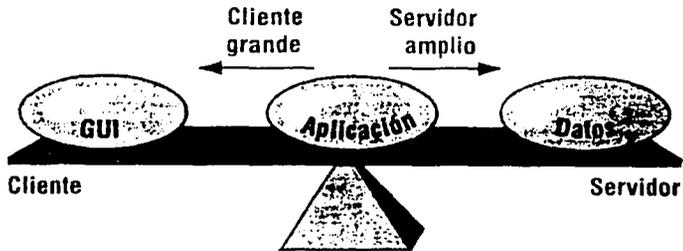


Figura 1.18. 1

Los **clientes** grandes son la modalidad más común de **cliente/servidor**. La mayor parte de la aplicación corre en el lado de la ecuación que corresponde al **cliente**. Los **clientes** grandes se emplean en software de apoyo de decisiones y personal. Ofrece flexibilidad y oportunidades para crear herramientas frontales que les permitan a los usuarios finales generar sus propias aplicaciones.

Las aplicaciones de **servidores** amplios son más fáciles de administrar y desplegar en red, porque la mayor parte de código corre en los **servidores**. Los **servidores** amplios pretenden reducir al mínimo los intercambios en red mediante la creación de niveles de servicio más abstractos. En el modelo de **servidor** amplio, el **cliente** aporta la GUI e interactúa con el **servidor** a través de llamadas de procesamientos remotos (o invocaciones de métodos).

1.4.8. ELEMENTOS DE CONSTRUCCIÓN DE CLIENTE/SERVIDOR.

Los tres elementos de construcción **cliente/servidor** son el **cliente** el **servidor** y el middleware que los entaza (Figura 1.19).

TESIS CON
FALLA DE ORIGEN

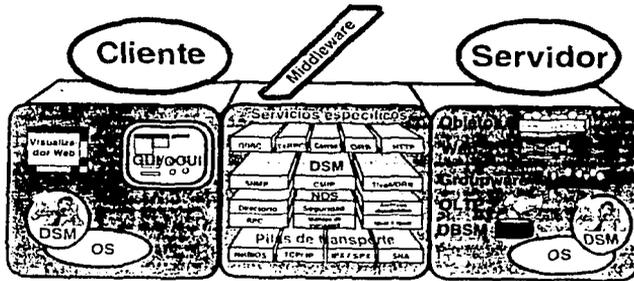


Figura 1.19. 1

El elemento del *cliente* ejecuta la parte del *cliente* de la aplicación. Corre en un sistema operativo(OS: operating system) que ofrece una interfaz gráfica de usuario (GUI: graphical user interface) o una interfaz de usuario orientada a objetos (OOUI: object oriented user interface) y puede acceder a servicios distribuidos, dondequiera que se encuentren. El sistema operativo descarga en el elemento de middleware la responsabilidad de manejar los servicios no locales, ejecuta un componente del factor de administración de sistemas distribuidos (DSM: distributed system management).

El elemento del *servidor* ejecuta la parte del *servidor* de la aplicación, la aplicación del *servidor* suele correr sobre un paquete comercial de software del *servidor*. Las plataformas del *servidor* son los servicios de bases de datos de SQL, los monitores TP, los *servidores* de groupware, los *servidores* de objetos y el Web.

El elemento de *middleware* se divide este elemento en tres categorías: pilas de transporte, sistemas operativos de redes(NOS: network operating system) y middleware de servicios específicos. El middleware es el sistema nervioso de la infraestructura de *cliente/servidor* y además cuenta con un componente del software de DSM.

Una estación de trabajo de administración recoge información de todos sus agentes en la red y la despliega gráficamente al igual que puede instruir a sus agentes que realicen acciones en su lugar. Se entiende a la administración como la conducción de una "red autónoma dentro de una red".

TESIS CON
FALLA DE ORIGEN

OBSERVACIONES.

En el presente capitulo se vio la parte teorica que se necesita para el analisis y desarrollo de una aplicaci3n. Empezando por los conceptos generales del modelo de **datos**, donde se explican las ventajas de utilizar un modelo relacional, ya que se describe lo que es una **tabla**, **campo**, **registro**, **llave**, etc. Lo que se intenta es conocer los principales elementos que conforman una **base de datos** y cual es la utilidad para guardar y obtener informaci3n.

En el siguiente capitulo lo que se quiere es conocer las necesidades y problemas de la empresa (Agencia de Viajes), con los puntos que se vieron en este capitulo se llega a una propuesta de soluci3n de acorde a la informaci3n recabada basandonos en entrevistas e informaci3n obtenida directamente de la empresa.

CAPITULO II

REQUERIMIENTOS DE LA EMPRESA DE VIAJES TURÍSTICOS.

2. REQUERIMIENTOS DE LA EMPRESA DE VIAJES TURÍSTICOS.

INTRODUCCION.

En este capitulo el objetivo primordial es conocer las necesidades especificas de la empresa; es muy importante para el diseño conocer el giro de la empresa, así como sus problemas actuales y a que esto será el detonante de nuestro desarrollo y solo conociendo al cliente podremos dar una propuesta de solución eficaz.

Por lo que al terminar este capitulo se tendrá necesariamente una visión mas amplia del movimiento del cliente, esto para que cuando se este diseñando la nueva aplicación sepamos si algo de lo que tiene actualmente la empresa puede ser reciclado y utilizado para la solución que proponemos.

2.1 GIRO DE LA EMPRESA.

La República Mexicana, es un País que por sus características geográficas cuenta con una infinidad de puntos turísticos para todo tipo de personas.

Pero debido a que durante los últimos años se han encarecido los servicios turísticos, menos familias pueden realizar sus viajes de descanso dentro del País. Esto fue la principal causa de que se creara la empresa "VIAJES OLIMPO", como una alternativa para las familias de niveles económicos medios, esta empresa tomo como base los destinos tanto de playa como ciudades coloniales mas visitados y que sus costos de hospedaje son relativamente cómodos, además esta empresa también organiza y realiza viajes turísticos que se adaptan a los gustos y necesidades de las personas.

Se penso que el medio de transporte ideal es el autobús, esto para que los costos sean bajos, debido a que su manutención es económica.

Los destinos con los cuales trabaja son:

Playas:

Acapulco.
Huatulco.
Puerto Escondido.
Puerto Vallarta,
Zihuatanejo,
Veracruz.
Mazatlán.

Ciudades:

Aguascalientes.
Guadalajara.
Guanajuato.
Oaxaca.
Puebla,
Querétaro
Taxco Guerrero.
Zacatecas.

Se manejan los siguientes recorridos fijos:

Acapulco.- Saliendo del D.F.
Acapulco.- Saliendo del D.F. y pasando por la Ciudad de Taxco Guerrero.
Aguascalientes.- Saliendo del D.F. con escala en la Ciudad de Querétaro.
Puerto Escondido.- Saliendo del D.F. con escala en las Ciudades de Puebla y Oaxaca.
Huatulco.- Saliendo del D.F. pasando por las Ciudades de Puebla y Oaxaca.
Zihuatanejo.- Saliendo del D.F.
Puerto Vallarta.- Saliendo del D.F. y pasando por la Ciudad de Guadalajara.
Oaxaca.- Saliendo del D.F. y pasando por la Ciudad de Puebla.
Guanajuato.- Saliendo del D.F. pasando por la Ciudad de Querétaro.
Puebla.- Saliendo del D.F.
Zacatecas.- Saliendo del D.F. pasando por las Ciudades de Querétaro y Aguascalientes.
Veracruz.- Saliendo del D.F.

Como se ha mencionado esta empresa no solo tiene rutas ya específicas, si no que puede organizar rutas según el gusto y necesidades del usuario, como pueden ser los viajes a balnearios o a zonas arqueológicas. El único requisito que pone la empresa es que se complete el cupo del autobús.

Esta empresa la matriz se encuentra en la Ciudad de México y cuenta con sucursales en Acapulco, Guadalajara, Guanajuato, Oaxaca, Puebla y Querétaro.

2.2. NECESIDADES

Como se ha mencionado esta empresa se dedica a realizar viajes turísticos, por lo que necesita controlar y dar seguimiento a la reservación de boletos, así como la programación de los viajes por realizar.

Además se necesita contar con la información en tiempo real de los asientos que se ocupan en las terminales intermedias, ya que el área administrativa requiere de ese informe para determinar si el viaje es costeable y así determinar si se puede realizar sin tener pérdidas, de lo contrario cancelar o suspender el viaje con suficiente tiempo de anticipación.

Por lo que la necesidad primordial de esta empresa es la de contar con un sistema informático con el cual pueden reservar los boletos de los diferentes recorridos y fechas en tiempo real para no vender boletos duplicados así como contar con suficiente información también para el área administrativa.

Todo lo anterior, es para cuidarlos problemas propios de las redes telefónicas tanto locales como de larga distancia y así estar en línea la matriz con todas las sucursales.

Lo anterior para evitar los problemas propios de las redes telefónicas tanto locales como de larga distancia y así estar en línea la matriz con todas sus sucursales.

Es importante mencionar que esta empresa como es pequeña tiene que hacer uso del servicio de algunas líneas de autobuses externas, con las que ya tiene un convenio que hace que el precio sea económico, por lo que también necesita tener un inventario de las líneas de autobuses con las que puede contar, además así se podrá saber de que línea es el autobús que realizara el viaje, esto también le sirve para ir consolidando los convenios con las empresas que le dan mejor servicio.

2.3. PROBLEMÁTICA ACTUAL Y REQUERIMIENTOS.

En la actualidad el sistema que opera esta empresa es muy simple, existe una matriz que es la que controla la venta de boletos, cada sucursal cuando vende algún boleto lo tiene que hacer es llamar por teléfono a la matriz, para que le informen si todavía hay boletos disponibles y cuales son estos boletos, así se puede dar el caso de que al mismo tiempo se tenga una llamada y el sistema no sea actualizado correctamente.

Además esto se traduce en mucho tiempo sobre todo en épocas vacacionales o "puentes", y es que tienen que estar llamando por cada venta y confirmando telefónicamente a la matriz, y como esta no tiene las suficientes líneas telefónicas, por lo que muchas veces suena ocupado y tienen que estar intentando en detrimento del tiempo del consumidor.

Este sistema se puede representar de la siguiente manera:

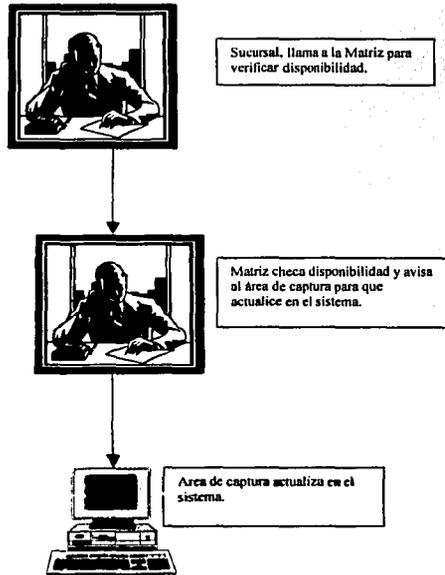


Figura 2.1.

El problema principal se debe a que es una empresa pequeña y de reciente creación, por lo que no cuenta con suficientes líneas telefónicas para tener comunicación con todas las sucursales al mismo tiempo y además las llamadas de larga distancia son costosas, y estos egresos se tienen que considerar en el valor de los boletos redundando en precios altos.

TESIS CON
FALLA DE ORIGEN

Además la gran mayoría de las sucursales no cuentan con una infraestructura robusta, solo basta con mencionar que solo tienen una línea telefónica que hace las veces de fax, lo que da una dimensión del problema que tienen para comunicarse con la matriz.

La empresa requiere de un sistema de venta de boletos en el cual se pueda consultar en el momento deseado la información real de los asientos reservados, así mismo se solicita que se realice el estimado de ventas para cada viaje en los diferentes puntos del recorrido.

Se requiere que este sistema sea totalmente amigable, esto para que con una sencilla capacitación el personal de ventas pueda operarlo sin problemas. El sistema tiene que mostrar visualmente los lugares disponibles para cada autobús; al mismo tiempo que se realice una venta se tiene que actualizar la base para que cualquier persona que lo consulte tenga la información real de cada viaje. Al mismo tiempo se requiere que se impriman los boletos en los que se tiene que incluir tanto los datos del viaje como los del usuario. También se requiere una consulta de tipo gerencial para que en la matriz el área administrativa tenga la información necesaria para tener en orden fiscalmente a la empresa.

Se requiere que la consulta despliegue los datos del viaje, indicando que persona ocupa cada lugar y además en donde abordaron el autobús, esta información también se requiere se imprima para que el operador y el encargado del viaje tengan un listado para organizar mejor este.

También se requiere que las sucursales tengan la hora exacta de la partida del autobús, ya que en estos momentos si el autobús tiene retrasos propios de la temporada o por algún inconveniente ajeno a la empresa, la única que tiene ese dato es la matriz y las sucursales tienen que llamar cuando se dan cuenta del retraso, y esto es por que la gente de la matriz en ocasiones no tiene el tiempo necesario para avisar a las sucursales. Como originalmente la matriz tenía asociación con diferentes empresas turísticas del interior de la República que hacían las veces de sucursal, existían múltiples problemas de comunicación y siempre se echaban la culpa uno al otro, es por ello que la empresa "VIAJES OLIMPO" se anima a participar sola creando pequeñas sucursales en diferentes puntos del País, que es cuando se empiezan a generar problemas con el sistema que manejan, ya que este es muy simple y no existe una base de datos real.

Una de las notas primordiales es que el sistema después de realizar el primer recorrido a cada destino, tendrá que generar datos estimados para que en lo sucesivo se pueda reservar mas exactamente los boletos para cada ciudad. Claro tomando en cuenta que estos datos variaran dependiendo de la temporada, pero como el área administrativa necesita datos estadísticos para determinar su mercadeo en cada viaje se estarán actualizando los datos.

Además la empresa requiere tener un listado completo de las personas que realizaran el viaje, esto para tener un control mas preciso de los pasajeros; además quieren que se realice una actualización exacta de la hora en que se realiza la partida, para que en las sucursales del interior se pueda tener la hora en que estará llegando el autobús a recoger a la gente de ellos. Por lo que esta fecha también se tendrá que tomar en cuenta para los reportes estadísticos y así corregir los problemas de impuntualidad.

2.4. PROPUESTA DE SOLUCION.

Las principales características del sistema propuesto son:

Como no todos los usuarios del sistema tienen que poder acceder a la información del sistema **SIAMO I**, principalmente debe existir una seguridad, esto para que el acceso sea basándose en el nivel que se le asigne.

También se ofrece un control de origen – destino, así como de las líneas de autobuses que prestaran servicio a rutas ya especificadas. Así mismo se debe tener el control de los viajes, ya que no necesariamente se realizan itinerarios en los cuales se pueda utilizar una ruta de autobuses comerciales. Además se tiene contemplado dividir el servicio por zonas geográficas del país, esto para tener un control más exacto de las sucursales.

Además se contará con las herramientas necesarias para poder verificar la productividad tanto de la matriz como de las sucursales, ya que las principales actividades tendrán objetivos, para que así se pueda contar con estadísticas fiables del funcionamiento.

El sistema además de todo lo anterior se podrá acceder desde cualquier sucursal de la agencia, lo que redundará en un mejor servicio al poder contar con información en línea en cualquier instante. Así no se vendrán lugares que ya fueron vendidos además se podrá contar con la venta de viajes redondos.

El área de contabilidad podrá verificar la información necesaria para que en cualquier instante se pueda realizar una auditoría y así mantener la economía sana de la empresa.

Se podrán generar los boletos en papel al mismo instante de la venta, además se tendrá en papel las estadísticas de la productividad tanto de cada sucursal como de la empresa en general.

Como el borrado de los elementos que trabajan para la empresa, dentro de los cuales podemos citar a las sucursales, líneas de autobuses y empleados por citar algunos no es física si no lógica se podrá tener información histórica en cualquier momento, además se está sugiriendo una opción de respaldo en dispositivos magnéticos, para que en cualquier momento que sea requerido se pueda utilizar.

Además de todo lo indicado anteriormente, el sistema será desarrollado con la posibilidad de que pueda ser actualizado y perfeccionado, sin tener que cambiar totalmente el esquema de funcionamiento, con lo que se ahorra una importante suma en este rubro.

Como se ve todos al mismo tiempo tendrán la misma información, lo que permitirá evitar la duplicidad de ventas, ya que en el momento que se reserve un boleto este aparecerá como vendido para todos los demás vendedores, si por cualquier razón este boleto no se vende se cambiara el status como libre, además el sistema tendrá que permitir la cancelación de los boletos con tiempo anticipado.

Gráficamente el sistema puede ser representado como lo muestra la figura 2.2:

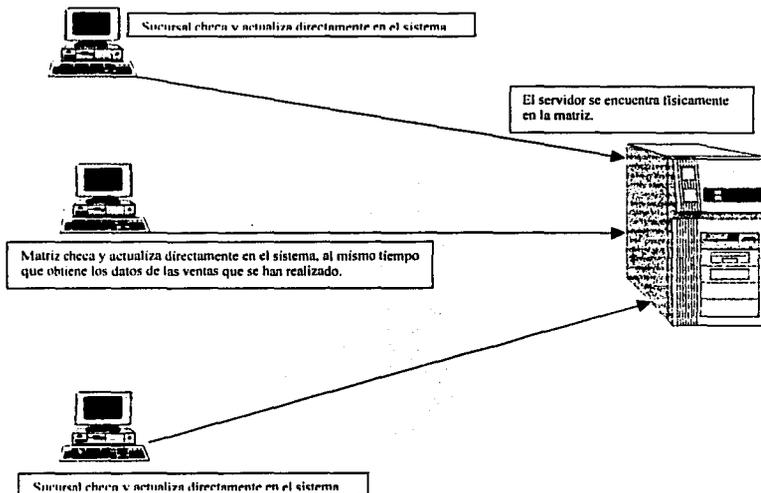


Figura 2.2.

OBSERVACIONES.

Como se vio el sistema que actualmente opera la empresa cuenta con una deficiencia muy grande principalmente y es no contar con las actualizaciones en línea para que los usuarios tengan un buen servicio ya que como utilizan el teléfono para hacer reservaciones y ventas dependen de los recursos humanos principalmente y como tal hay un gran porcentaje de que existan errores y se puedan duplicar ventas o que no se registre la venta de un boleto redundando en un servicio incapaz de generar satisfacciones al cliente.

En el siguiente capítulo lo que se intenta es que tomando como base la propuesta de solución que se da se pueda ir desarrollando una aplicación mas robusta.

TESIS CON
FALLA DE ORIGEN

CAPITULO III

ANALISIS

3. ANÁLISIS

El objetivo principal de este capítulo es realizar el análisis de la propuesta de solución y demostrarle al cliente de la necesidad de robustecer su operación para que pueda crecer en base al servicio que pueda prestarle al cliente.

Para realizar el análisis del sistema, primero se concreto una entrevista con el cliente, donde nos mostró parte del sistema con el que actualmente trabajan.

El cliente explico la necesidad de tener un sistema mas seguro, en línea y sobre todo de fácil manejo. De la misma manera vimos la necesidad de utilizar la mayoría de los recursos con los que cuenta, esto para que no se gastara demasiado.

Al realizar el análisis determinamos que las herramientas a utilizar tendrían que ser en modo gráfico al mismo tiempo de un bajo costo. La propuesta del tiempo que nos tardaremos en la elaboración del sistema se muestra en el siguiente diagrama de Gantt.

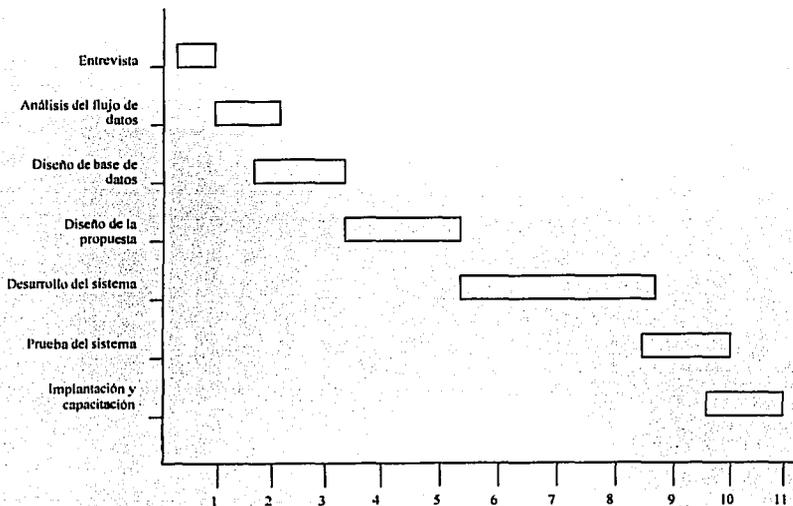


Figura 3.1.

TESIS CON
FALLA DE ORIGEN

En la entrevista además de obtener información para conocer perfectamente la forma de operar de la empresa, se intento conocer propuestas de cómo les gustaría a los empleados que fuese el sistema para que les ayudara a realizar su trabajo de la forma mas fácil y efectiva.

Aquí principalmente dentro de las solicitudes fue que el sistema fuera en un ambiente totalmente gráfico y sobre todo muy amigable, para que de esta forma el servicio fuera efectiva y el cliente quedara satisfecho por la atención y el tiempo de esta.

Además se tenia que tomar en cuenta que la instalación y capacitación no podía ser en la llamada temporada alta ya que si se daba en ese periodo sería en detrimento del servicio y sobretodo del cliente.

Se determino que como toda la información del sistema ya estaría grabada sería factible que el sistema emitiera los reportes necesarios tanto para manejar estadísticas de la productividad de la empresa, así como los reportes administrativos para poder controlar fácilmente los viajes que se tengan que realizar y los itinerarios de los pasajeros.

Como la empresa por el momento se puede considerar pequeña pero con expectativas de crecer, el sistema en cuestión tiene que ser capaz de poder ir acoplándose fácilmente y de una manera económica a las necesidades de la empresa. Además se tiene que ser robusto en todos los sentidos, ya que además de ser rápido y eficaz también tiene que ser muy seguro y no cualquiera podrá modificar información ni consultarla, para lo que requieren un esquema de seguridad por niveles ya que como todo sistema de ventas se tendrá información confidencial.

Por lo que se dice que se espera un sistema robusto en todos los sentidos.

3.1. HERRAMIENTAS DE DESARROLLO.

Las herramientas de desarrollo con las que se pensó trabajar el sistema son **ERWIN**, **VISUAL BASIC** y **SQL SERVER**.

Las cuales se describen a continuación:

ERWIN :

Erwin es una herramienta para el diseño de base de datos. Contiene propiedades expandibles y modelado dimensional para el diseño y modelado de DataWareHouse. También incluye otras aplicaciones basadas en bases de datos relacionales en las cuales se requiere un diseño iterativo entre sus niveles físicos y lógicos.

Podemos crear disparadores, proceso almacenados, diagramas físicos del modelado de la base de datos con sus respectivas tablas y genera el código el cual permite crear la base en el manejador de base de datos que se va a utilizar.

VISUAL BASIC

'La palabra "Visual" hace referencia al método que se utiliza para crear la interfaz gráfica de usuario (GUI).

En la actualidad ya no es necesaria escribir código para crear la apariencia y la ubicación de los elementos de la interfaz, lo único que hay que hacer es agregar objetos que ya están diseñados y colocarlos en el lugar que sean necesarios.

'La palabra "Basic" hace referencia al lenguaje BASIC (Beginners All-Purpose Symbolic Instruction Code).

Visual Basic ha evolucionado a partir del lenguaje BASIC en la actualidad contienen centenares de instrucciones, funciones y palabras clave, las cuales en su mayoría están relacionadas con la interfaz gráfica de Windows. Con esta tecnología es fácil crear aplicaciones con sólo aprender pocas palabras clave, y en el caso de personas que ya tienen conocimientos en versiones anteriores satisfacer cualquier objetivo que puedan alcanzar mediante este lenguaje de programación.

El lenguaje de programación *Visual Basic* está incluido en Microsoft Excel, Microsoft Access y muchas aplicaciones Windows.

SQL SERVER :

SQL Server es una base de datos relacional. SQL lenguaje estructurado de consulta (Structured Query Language) es utilizado para definir, modificar y gestionar datos nos permite administrar los cambios en la base de datos usando tablas, índices, llaves, filas y columnas para almacenar la información.

²*SQL Server 7* tiene una gestión de direcciones de 64 bits para permitir la ejecución de aplicaciones que necesiten más de los 2GB de memoria permitidos por un software de 32 bits. Windows NT y *SQL Server* llegarán a usar ocho procesadores y podrán emplear hasta 3 GB de memoria de aplicaciones.

² Visual Basic 6.0 Manual de programador, Microsoft Corporation

² SQL Server 7 Gayle Coffman

3.1.1. INTEGRACION

En la actualidad el tiempo de diseño es importante ya que cuando utilizamos un manejador de base de datos para el diseño y el modelado de la misma, es muy lento ya que es necesario crear la base de datos las tablas que la constituyen mas todos sus parámetros, todo lo anterior se logra mediante instrucciones del mismo manejador lo cual es tedioso, tardado y repetitivo y es que para crear las tablas se utilizan las mismas instrucciones solo se diferenciaran por los campos que contengan cada una de ellas.

Por lo cual se utilizo como herramienta del modelado de la base de datos de la agencia de viajes; **Erwin** por su flexibilidad en el modelado y diseño por medios gráficos ya que solo se dibuja el diagrama con sus respectivos campos y se marca un campo el cual será la llave primaria y se unen por medio de líneas, terminado este proceso se selecciona el manejador de datos que será utilizado para contener la base y por ultimo se genera el código.

Visual Basic es el lenguaje más comercial por ser sencillo, flexible y tener una compatibilidad con todos los equipos donde este instalado el sistema operativo Windows, a pesar de ser sencillo se pueden realizar aplicaciones robustas con interfaces gráficas muy aceptadas.

Con este lenguaje es fácil realizar la programación de cualquier aplicación, ya que aquí se utilizan los objetos, cada objeto ya tiene predeterminadas ciertas instrucciones por lo que se facilita el código; **Visual Basic** como su nombre lo indica es un lenguaje orientado a una forma visual, por lo que la presentación es mas sencilla y por los elementos que contiene cada objeto se ahorra código.

Microsoft ha diseñado su manejador de base de datos que es **SQL SERVER** siendo utilizado por medianas y grandes empresas, ya que su costo es relativamente económico.

Como se pensó en la utilización de **Visual Basic** se determino que el mejor manejador para la base sería **SQL SERVER** por ser compatible y realizado por la misma empresa (MICROSOFT). A lo anterior se llevo después de haber considerado ORACLE, pero se encontró que este es sumamente caro.

Por ser herramientas de fácil manejo, así como tener una amplia aceptación en el medio informático y un costo no muy fuerte, se escogieron como herramientas básicas para la generación de esta aplicación que en general se solicito por parte de la empresa de viajes que su costo de mantenimiento y actualización no fuera mucho, pero sobre todo que tuviese una escalabilidad amplia y que la misma no fuese mas complicada que la realización de este proyecto.

Ya que de lo contrario cada que se quisiera modificar o agregar algún modulo a este se tendría que solicitar un nuevo análisis tanto de sistema como de la base de datos, lo que implicaría un costo elevado para la empresa.

3.2. ANÁLISIS DEL SISTEMA DE AGENCIA DE VIAJES TURÍSTICOS

En el primer contacto que se tuvo para el análisis del sistema siavo fue con personal de la agencia quien expuso sus necesidades y la forma de trabajar manualmente.

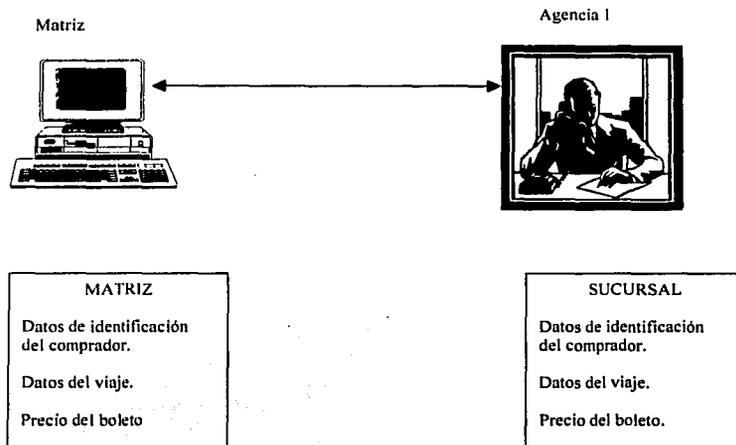


Figura 3.2.

Cada agencia se comunica con la matriz para conocer la disponibilidad de los boletos para cada viaje, en la agencia avisa y recibe vía telefónica los datos necesarios, la sucursal genera el boleto.

Como se ve en las tablas anteriores, en cada equipo se captura la información necesaria, por lo que se requiere doble captura y doble espacio, ya que el operador captura y graba en su PC la información, el administrador en la matriz también realiza lo propio y posteriormente vía dispositivos magnéticos se realiza un cruce para

TESIS CON
FALLA DE ORIGEN

encontrar posibles diferencias entre cada una de las capturas, como se ve con este pequeño ejemplo se trata de doble trabajo con los errores consecuentes. Además es importante especificar que la atención a cada usuario es muy lenta, por que como no se tienen suficientes líneas telefónicas y en lo que el operador de la matriz checa y registra cada movimiento se pierde tiempo.

Después de analizar las opciones que se tienen determinamos que el siguiente esquema sería el mejor para que la operación sea mas ágil, redundando en una mejor atención para con el usuario.

En el esquema que proponemos solo se necesita capturar una única vez la información, la cual esta en línea y puede ser accesada en cualquier momento desde cualquier lugar.

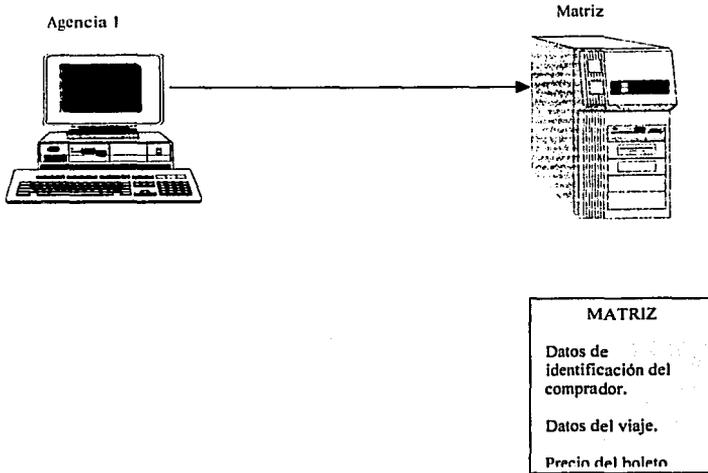


Figura 3.3.

Como se vera solo se captura en la sucursal, actualizando en ese mismo instante la base de datos la cual se encuentra en la matriz y puede ser actualizada en cualquier momento desde cualquier sucursal, asi la base siempre esta en línea y cuenta con datos reales, además la atención al usuario se realiza en unos cuantos segundos, y se realiza la impresión del boleto en el momento que se requiera.

TESIS CON
FALLA DE ORIGEN

Nos dimos cuenta que es importante para la empresa conocer en que lugar fue comprado el boleto, por lo que se necesita saber que agencia fue la expendedora del boleto; así como es una empresa pequeña no cuenta con los autobuses necesarios, por lo que trabaja con las líneas comerciales, por lo mismo se necesita saber que línea realizara el viaje, además nos explicaron que hay diferentes descuentos, los cuales pueden ser entre otros para niños, maestros, insen, etc., obviamente también se necesitan tanto el origen como el destino del viaje.

Por lo anterior se penso en manejar catálogos, ya que se cuentan con listas predeterminadas, pensándose en crear los catálogos siguientes:

- Agencias.
- Líneas de autobuses.
- Orígenes.
- Destinos.
- Tipo de Descuento.

Lo anterior se determino, par que no existieran errores al momento de grabar la información y así mismo se siguiera una misma forma, ya sea de abreviar o de poner nombres y así no generar confusión al momento de querer interpretar la información.

Además nos proporcionaron la suficiente información como para poder entender en una forma mas amplia la operación del sistema, ya que como se pretende que se reutilice alguna de la información con la que cuentan actualmente se requiere comprender la operación actual y al mismo tiempo diseñar la posible utilización de la información existente.

Al mismo tiempo nos proporcionaron los siguientes datos los cuales son los necesarios para el buen funcionamiento de la agencia. Hay que recordar que además de los catálogos que se indicaron anteriormente se tiene que guardar la información principal en tablas, las cuales tienen que describir en forma concreta el tipo de información que tendrán, por lo que el nombre de la misma tiene que hacer referencia lo mejor posible a la información.

Como es una empresa que cuenta con algunas sucursales en el interior del país lo primero que se necesita saber es el catalogo de las agencias con las que se cuenta, en este catalogo se tiene que poder ir registrando cada movimiento que se realice, es decir dar de baja o alta sucursales e indicar si es posible una clave con la que se conocerá esta para no tener que realizar una captura excesiva por parte del operador.

Por lo que la información obtenida después de realizar un pequeño análisis quedo registrada en la tabla Agencias, en la cual se inicio poniendo la totalidad de la información que se considera necesaria para poder llevar un manejo perfecto de la agencia de viajes.

Cabe hacer la aclaración que descubrimos que hay información que se duplica pero esto será objeto de un análisis mas explícito en paginas posteriores. En la tabla siguiente se intento poner la información que actualmente se ocupa y la que se requiere.

TABLA AGENCIAS

Agencia 1	Líneas	Numero viaje	Fecha de viaje	Hora de viaje
Ciudad México	TNT	306	13/08/2001	15:30 p.m.
Zacatecas	Flecha roja	500	15/09/2001	22:10 p.m.
Zacatecas	TNT	280	25/08/2001	07:00 a.m.

Hora de salida	Kilometro viaje	Total pasajeros	Vendidos	Importe
15:35 p.m.	150	35	30	7,500.00
22:20 p.m.,	300	45	45	14,400.00
07:04 a.m.	300	35	29	8,700.00

Num asiento	Tipo_vta1	Tipo_vta2	Tipo_vta3	Tipo_Vta4
32	Menor			
01		Insen		
35			Estudiante	

Origen paso	Destino paso	Tarifa
Puebla	Oaxaca	250.00
Querétaro	Aguascalientes	320.00
		300.00

Tabla 3.1.

Al tener los datos empleados, necesitamos un buen diseño de bases de datos ya que si no se hace incorrectamente causara grandes problemas cuando se tengan que realizar consultas muy complicadas.

Para esto tendremos que utilizar las reglas de Normalización, esto consiste en eliminar redundancias e inconsistencias de dependencias en el diseño de tablas. Las reglas son cinco.

3.3. NORMALIZACION

3.3.1. PRIMERA REGLA DE NORMALIZACIÓN.

Es muy importante eliminar los grupos repetitivos de las tablas individuales, crear una tabla separada por cada grupo de datos relacionados e identificar cada grupo de datos relacionados con una clave primaria.

Como vemos desde el momento de estar repitiendo el campo Tipo_vta ya no se cumple la primera regla además tenemos que colocar un campo que no se repita para cada campo de agencias, esté será la llave (clave) primaria.

TABLA AGENCIAS

Cve_Agen	Agencia 1	Líneas	Numero viaje	Fecha de viaje
DF	Ciudad México	TNT	306	13/08/2001
ZAC	Zacatecas	Flecha roja	500	15/09/2001
ZAC	Zacatecas	TNT	280	25/08/2001

Hora de viaje	Hora de salida	Kilometro viaje	Total pasajeros	Vendidos
15:30 p.m.	15:35 p.m.	150	35	30
22:10 p.m.	22:20 p.m.	300	45	45
07:00 a.m.	07:04 a.m.	300	35	29

Importe	Num asiento	Tipo_vta1	Origen paso	Destino paso
7,500.00	32	Menor	Puebla	Oaxaca
14,400.00	01	Insen	Querétaro	Aguascalientes
8,700.00	35	Estudiante		Estudiante

Tarifa
250.00
320.00
300.00

Tabla 3.2.

De esta forma resulta nuestra nueva tabla cumpliendo la primera regla, al eliminar el campo que se estaba repitiendo (Tipo_vta).

3.3.2. LA SEGUNDA REGLA DE NORMALIZACIÓN.

Consiste en crear tablas separadas para aquellos grupos que se aplican a varios registros y relacionar estas tablas mediante claves externas.

Para esto se tendrá que separar el campo Tipo_vta en otra tabla quedando de la siguiente manera.

TABLA AGENCIAS

Cve_Agen	Agencia 1	Líneas	Numero viaje	Fecha de viaje
DF	Ciudad México	TNT	306	13/08/2001
ZAC	Zacatecas	Flecha roja	500	15/09/2001
ZAC	Zacatecas	TNT	280	25/08/2001

Hora de viaje	Hora de salida	Kilometro viaje	Total pasajeros	Vendidos
15:30 p.m.	15:35 p.m.	150	35	30
22:10 p.m.	22:20 p.m.,	300	45	45
07:00 a.m.	07:04 a.m.	300	35	29

Importe	Num asiento	Origen paso	Destino paso	Tarifa
7,500.00	32	Puebla	Oaxaca	250.00
14,400.00	01	Querétaro	Aguascalientes	320.00
8,700.00	35	Estudiante		300.00

Tipo_vta
1
2
3

Tabla 3.3.

TABLA TIPO_VTA

Tipo_vta	Descripción	Descuento	Status
1	Menor	0	A
2	Insen	50	A
3	Estudiante	45	A

Tabla 3.4.

Se han creado las tablas separadas en la tabla **agencias**, esta relacionada con la clave externa en la tabla **Tipo_vta** , **Tipo_venta**.

¿Qué pasaría si tuviéramos que añadir otro Número de viaje o 10 más se tendríamos que teclear la líneas y su status.

3.3.3. TERCERA REGLA DE NORMALIZACIÓN.

Habrà necesidad de eliminar aquellos campos que no dependan de la clave.

TABLA AGENCIAS

Cve_Agen	Nombre	Status
DF	Ciudad México	A
ZAC	Zacatecas	A
ZAC	Zacatecas	A

Tabla 3.5.

TABLA LINEAS

Cve_lin	Cva_Agen	Nombre	Status
001	DF	TNT	A
002	ZAC	Flecha Roja	A
003	PUE	AU	A

Tabla 3.6.

TABLA DESTINOS

Cve_des	Descripcion	Status
OAX	OAXACA	A
PUE	PUEBLA	A
GTO	GUANAJUATO	A

Tabla 3.7.

TABLA ORIGEN

Cve_origen	Descripcion	Status
001	DF	TNT
002	ZAC	ZACATECAS
003	QRO	QUERETARO

Tabla 3.8.

TABLA MERCADOS

Cve_merc	Cve_lin	Cve_agen	Cve_dest	Cve_origen
01	003	PUE	OX	DF
02	001	TNT	PUE	ZAC
03	002	ZAC	GTO	QRO

Status
15:30 p.m.
22:10 p.m.
07:00 a.m.

Tabla 3.9.

ANALISIS

TABLA VIAJES

Num_viaje	Fecha_viaje	Hora_viaje	Cve_merc	Cve_agen
306	13/08/2001	15:30 p.m.	01	PUE
500	15/09/2001	22:10 p.m.	02	TNT
280	25/08/2001	07:00 a.m.	03	ZAC

Cve_dest	Cve_origen	Hora_salida	Km_viaje	Total_pasajero
OAX	DF	15:30 p.m.	150	35
PUE	ZAC	22:10 p.m.	300	45
GTO	QRO	07:00 a.m.	300	35

Vendidos	Importe	Status
30	7,500.00	A
45	14,400.00	A
29	8,700.00	A

Tabla 3.10.

TABLA ITINERARIOS

Num_viaje	Fecha_viaje	Hora_viaje	Cve_merc	Cve_agen
306	13/08/2001	15:30 p.m.	01	PUE
500	15/09/2001	22:10 p.m.	02	TNT
280	25/08/2001	07:00 a.m.	03	ZAC

Cve_dest	Cve_origen	Origen_paso	Destino_paso	Tarifa
OAX	DF	PUE	OAX	35
PUE	ZAC	DF	GUA	45
GTO	QRO	DF		35

Status
A
A
A

Tabla 3.11.

TABLA ASIENTOS

Num asiento	Num viaje	Fecha viaje	Hora viaje	Cve merc
25	306	13/08/2001	15:30 p.m.	01
35	500	15/09/2001	22:10 p.m.	02
01	280	25/08/2001	07:00 a.m.	03

Cve agen	Cve dest	Cve origen	Origen paso	Destino paso
PUE	OAX	DF	PUE	OAX
TNT	PUE	ZAC	DF	GUA
ZAC	GTO	QRO	DF	

Tipo Vta	Nombre	Ape pat	Ape mat
Menor	JOSE	RODRIGUEZ	HERNANDEZ
Insen	ARMANDO	MENDEZ	CARRANZA
Estudiante	RODOLFO	ROMERO	LOPEZ

Tabla 3.12.

3.3.4. CUARTA REGLA DE NORMALIZACIÓN.

Las relaciones varios con varios, entidades independientes no pueden ser almacenadas en la misma tabla.

En nuestro caso no existen relaciones de varios a varios.

3.3.5. QUINTA REGLA DE NORMALIZACIÓN

La tabla original debe ser reconstruida desde las tablas resultantes en las cuales a sido modificada.

Después de analizar la forma en que se fueron integrando las tablas se determino que en este caso se podrían utilizar algunos catálogos para facilitar el manejo del sistema, ya que la gran mayoría de los datos se van formando a raíz de la generación del viaje, y en este caso el número de viaje sería nuestra llave principal.

ANALISIS

También se determinó que este sistema tiene que tener una seguridad para que una persona sea la que lo administre. Por lo que se pensó en generar niveles de operación, para así mantener el sistema seguro; estableciendo la utilización de módulos dentro del sistema, en los que solo podrán operar las personas que se les asigne dicho permiso.

Así se utilizarán por ejemplo módulos de administración, módulos de venta, etc. Es importante señalar que este sistema además de manejar viajes manejará los ingresos obtenidos.

En estos momentos el sistema cuando realiza reservaciones de hoteles no tiene participación directa, lo único que se hace es hablar a los hoteles que estén en la zona, de acuerdo a la categoría que solicite el cliente, se hace la reservación y el cliente se arregla directamente con la empresa hotelera directamente al llegar al destino.

Ahora se requiere realizar un módulo en el que se le cobrará al cliente y se le realizará la reservación, entregándole directamente la factura, y la empresa se hará responsable de todo lo relacionado con el hotel, obviamente este módulo solo se podrá operar en lugares donde se tiene convenio y la sucursal de la agencia se encuentra directamente conectada con el hotel, es decir la sucursal ocupa una instalación del mismo hotel.

De lo contrario sería muy difícil llevar a cabo la actualización al sistema del hotel.

Además de ser muy caro y tendríamos que buscar una forma de enlazar la tecnología de ambas empresas.

Los principales datos que solicitan los hoteles son:

TABLA: HOTELES

Cve_hotel	Cve_dest	Nombre	Categoría	Cant_habitaciones	Status
PUE	PUE	PUEBLA	****	360	A
JAL	GUA	JALISCO	***	120	B
REA	OAX	REAL	*****	60	A

Tabla 3.13.

TABLA: HABITACIONES

Num_habitacion	Cve_hotel	Cve_dest	Num_factura	Tipo_habitacion
1	PUE	PUE	100001	Sencilla
1	JAL	GUA	100002	Triple
1	REA	OAX	100000	Doble

Tarifa	Fecha_entrada	Fecha_salida	Nombre	Ape_pat	Ape_mat
600	20/03/2001	25/03/2001	José	Guillen	Mora
400	06/04/2001	20/04/2001	María	Salazar	Sánchez
1000	12/01/2001	20/01/2001	Juan Carlos	Pérez	López

Tabla 3.14.

TABLA: CUARTOS

Num_habitacion	Cve_hotel	Cve_dest	Num_factura	Fecha
1	PUE	PUE	100001	01/03/2001
1	JAL	GUA	100002	06/04/2001
1	REA	OAX	100000	12/01/2001

Tabla 3.15.

El modulo de reservaciones para los hoteles será realmente pequeño, ya que la rama principal de la agencia es la venta de boletos para viajes.

OBSERVACIONES.

Como se vio en este capítulo originalmente se tenían tablas muy grandes y para poder obtener información se necesitaba que varios campos fueran llaves para que el acceso fuera ágil. Después de analizar la problemática se llegó a la conclusión de que estaban mal elaboradas las tablas por que tenían mucha información duplicada.

Se procedió a la normalización generando catálogos y tablas auxiliares lo que redundó en una mejor accesibilidad. También se verificó y si existen tablas que pueden ser reutilizadas solo con unos pequeñas adecuaciones.

Tomando como base la propuesta que se dio en este capítulo y en el anterior, en el siguiente se empieza con el desarrollo de la aplicación, diseñando la **base de datos** con sus **tablas** aplicando ya la normalización y llevando un modelado **relacional** para que las tablas tengan una "comunicación" perfecta y se puedan generar consultas y actualizaciones mas rápidas.

CAPITULO IV

DESARROLLO

4. DESARROLLO

El presente capítulo tiene el objetivo primordial de generar el desarrollo del sistema desde su **base** con cada una de sus **tablas** hasta la aplicación en sí. En este capítulo se detalla más a fondo la descripción de las tablas y se comenta aunque sea de un modo muy general la operación del sistema.

Después de haber detectado las necesidades de la Agencia de Viajes Turísticos, y de haber realizado la normalización de la información que ellos requieren se llegó a la conclusión que las tablas quedarían de la siguiente manera.

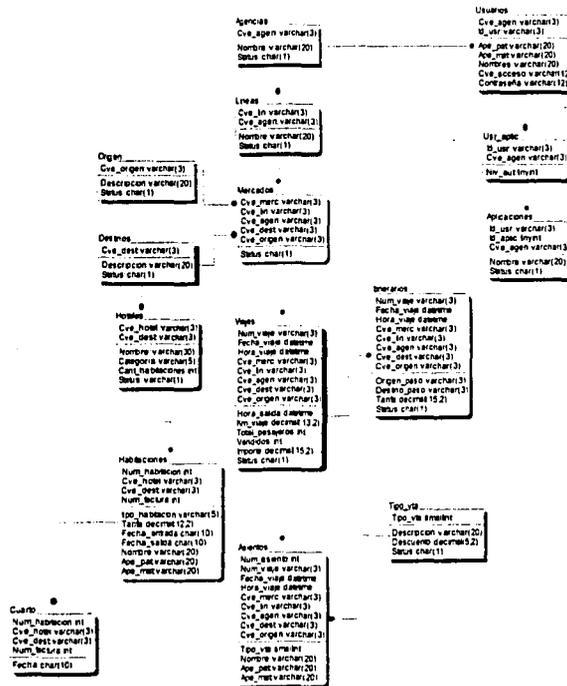


Figura 4.1.

4.1. DESCRIPCIÓN DE LOS CAMPOS PRINCIPALES

El campo status de las tablas que lo tienen es para borrar los registros de una forma lógica, de esta forma si el campo esta marcado con una "A" significa que este registro esta activo y si no es así este no aparecerá en la información real. Pero si los podremos conservar para un informe histórico. A continuación se describen los campos mas importantes de este sistema:

Tabla: Agencias:

La Agencia tiene varias sucursales distribuidas en diferentes partes de la República Mexicana, las cuales tiende a cambiar de lugar de origen. Con los datos que se actualizan podremos saber cuando una de ellas a sido dada de baja para esto aparte de todos sus datos tenemos un campo (Status) que nos determinara si esta activa o inactiva.

Cve_agen	Varcha(3)	Clave de Agencia expendedora
Nombre	Varchas(20)	Nombre de la Agencia expendedora
Status	Char(1)	Status: A = Activo, B = Baja.

Tabla 4.1.

Tabla: Líneas:

La Agencia de Viajes Turísticos mantiene una gran relación con las líneas camioneras las cuales les proporcionan el servicio de transporte, de esta forma tendrán un mayor control sobre las líneas que son más solicitadas por cubrir una extensión más grande dentro de la República Mexicana.

Cve_lin	Varcha(3)	Clave de Línea de autobús que realizara el viaje.
Cve_agen	Varcha(3)	Clave de Agencia expendedora
Nombre	Varchas(20)	Nombre de la línea de autobús
Status	Char(1)	Status: A = Activo, B = Baja.

Tabla 4.2.

Tabla: Origen

Se registran todos los orígenes del viaje que se va a realizar.

Cve_origen	Varcha(3)	Clave del Origen del viaje.
Descripción	Varchar(20)	Descripción del Origen
Status	Char(1)	Status: A = Activo, B = Baja.

Tabla 4.3.

Tabla: Destino.

Al igual que el anterior pero estos serán los destinos .

Cve_dest	Varcha(3)	Clave del Destino del viaje.
Descripción	Varchar(20)	Descripción del Destino
Status	Char(1)	Status: A = Activo, B = Baja.

Tabla 4.4.

Tabla: Mercado.

El mercado es muy importante ya que con esto se tiene un control del origen y destino. Por ejemplo México – Oaxaca, México – Puebla.

Cve_merc	Varcha(3)	Clave del Mercado
Cve_lin	Varcha(3)	Clave de Línea de autobús
Cve_agen	Varcha(3)	Clave de Agencia expendedora
Cve_dest	Varcha(3)	Clave de Destino del viaje
Cve_origen	Varcha(3)	Clave de Origen del viaje
Status	Char(1)	Status: A = Activo, B = Baja.

Tabla 4.5.

Tabla: Hoteles

Catalogo de hoteles por destino. Y están separados de acuerdo a la categoría del hotel.

Cve_hotel	Varcha(3)	Clave del Hotel
Cve_dest	Varcha(3)	Clave de Destino del hospedaje
Nombre	Varcha(30)	Nombre del hotel
Categoría	Varcha(5)	* ** *** **** y *****
Cant_habitaciones	Int	Número de habitaciones del hotel.
Status	Char(1)	Status: A = Activo, B = Baja.

Tabla 4.6.

Tabla: Habitaciones

Sirve para indicar quien esta ocupando una habitación. En un período determinado. Esto también sirve para verificar si el hotel todavía tiene capacidad para recibir mas gente en una fecha específica.

Num_habitacion	int	Número de la habitación.
Cve_hotel	Varcha(3)	Clave del Hotel
Cve_dest	Varcha(3)	Clave de Destino del hospedaje
Num_factura	Int	Número de factura
Tipo_habitacion	Varcha(5)	Sencilla, Doble, Triple, etc.
Tarifa	Decimal(12,2)	Tarifa del hospedaje
Fecha_entrada	Char(10)	Fecha de entrada al hotel
Fecha_salida	Char(10)	Fecha de salida del hotel
Nombre	Varcha(20)	Nombre del cliente
Ape_pat	Varcha(20)	Apellido paterno
Ape_mat	Varcha(20)	Apellido materno

Tabla 4.7.

Tabla: Cuartos

Sirve para indicar cuando esta ocupada una habitación y que fecha.

Num_habitacion	int	Número de la habitación.
Cve_hotel	Varcha(3)	Clave del Hotel
Cve_dest	Varchar(3)	Clave de Destino del hospedaje
Num_factura	Int	Número de factura
Fecha	Char(10)	Fecha de ocupación del cuarto.

Tabla 4.8.**Tabla: Viajes.**

En esta tabla se tendrán los horarios de cada mercado para tener un manejo más eficiente.

Num_viaje	Varcha(3)	Número de viaje.
Fecha_viaje	Datetime	Fecha en que se realiza el viaje
Hora_viaje	Datetime	Hora en que se realiza el viaje
Cve_merc	Varcha(3)	Clave del Mercado
Cve_lin	Varcha(3)	Clave de Línea de autobús
Cve_agen	Varcha(3)	Clave de Agencia expendedora
Cve_dest	Varcha(3)	Clave de Destino del viaje
Cve_origen	Varcha(3)	Clave de Origen del viaje
Hora_salida	Datetime	Hora en que sale el autobús. (Tiempo real)
Km_viaje	Decimal(13,2)	Kilómetros del recorrido
Total_pasajeros	Int	Total de pasajeros del autobús
Vendidos	Int	Número de boletos vendidos
Importe	Decimal(15,2)	Importe de los boletos vendidos.
Status	Char(1)	Status: A = Activo, B = Baja.

Tabla 4.9.

Tabla: Itinerarios.

Se mencionaran todos los puntos intermedios que se encuentren en un mercado por ejemplo el de México – Oaxaca los puntos Puebla – Huichitan. Además también nos sirve para determinar cuanto se ha percibido de la venta de boletos.

Num_viaje	Varcha(3)	Número de viaje.
Fecha_viaje	Datetime	Fecha en que se realiza el viaje
Hora_viaje	Datetime	Hora en que se realiza el viaje
Cve_merc	Varcha(3)	Clave del Mercado
Cve_lin	Varcha(3)	Clave de Línea de autobús
Cve_agen	Varcha(3)	Clave de Agencia expendedora
Cve_dest	Varcha(3)	Clave de Destino del viaje
Cve_origen	Varcha(3)	Clave de Origen del viaje
Origen_paso	Varcha(3)	Esto en caso de que sea origen intermedio.
Destino_paso	Varcha(3)	Esto en caso de que sea destino intermedio.
Tarifa	Destino(15,2)	Tarifa del destino intermedio.
Status	Char(1)	Status: A = Activo, B = Baja.

Tabla 4.10.

Tabla: Asientos

Indica que asiento ya fue vendido y quien lo compro.

Num_asiento	Int	Numero del asiento.
Num_viaje	Varcha(3)	Número de viaje.
Fecha_viaje	Datetime	Fecha en que se realiza el viaje
Hora_viaje	Datetime	Hora en que se realiza el viaje
Cve_merc	Varcha(3)	Clave del Mercado
Cve_lin	Varcha(3)	Clave de Línea de autobús
Cve_agen	Varcha(3)	Clave de Agencia expendedora
Cve_dest	Varcha(3)	Clave de Destino del viaje
Cve_origen	Varcha(3)	Clave de Origen del viaje
Tipo_vta	Smallint	Tipo de Venta. (Posible Descuento).
Nombre	Varchar(20)	Nombre del pasajero
Ape_pat	Varchar(20)	Apellido paterno del pasajero
Ape_mat	Varchar(20)	Apellido materno del pasajero

Tabla 4.11.

PERSONALIA

Table 4.11

Sum of squares between and within for categorical independent

Source of Variation	Sum of Squares	Sum of Squares Between	Sum of Squares Within
Between			
Within			
Total			

Table 4.11

Table 4.12

Sum of squares between and within for categorical independent and one dummy for categorical dependent variable

Source of Variation	Sum of Squares	Sum of Squares Between	Sum of Squares Within
Between			
Within			
Total			

Table 4.12

Tabla: Usr_aplic

Será en base a la tabla de usuarios y aplicaciones ya que es una relación de muchos a muchos, esa tabla nos permitirá dar determinados niveles de autorización para el manejo de la información.

Id_usr	Varchar(3)	Identificación del Usuario
Cve_agen	Varchar(3)	Clave de la agencia
Niv_aut	tinyint	Indica que nivel tiene el usuario y permite acceder únicamente las aplicaciones permitidas.

Tabla 4.14.

Tabla: Aplicaciones

Se le proporciona un identificador (Id_aplic) a las aplicaciones.

Id_usr	Varchar(3)	Identificación del Usuario
Id_aplic	tinyint	Identificación de la aplicación
Cve_agen	Varchar(3)	Clave de la agencia
Nombre	Varchar(20)	Nombre de la aplicación
Status	Char(1)	Status: A = Activo, B = Baja.

Tabla 4.15.

4.2. DESCRIPCIÓN DE LA OPERACIÓN.

Inicialmente se determino que para que puedan ingresar al sistema se necesita digiten una clave de acceso con su respectiva contraseña, esto para que se determine a que niveles tendrá acceso el usuario.

Por lo que se determino usar la siguiente pantalla donde se solicita la clave de usuario y la clave de acceso del usuario, con esta información se puede saber exactamente de donde es el usuario, a que nivel tiene permitido acceder y quien es.

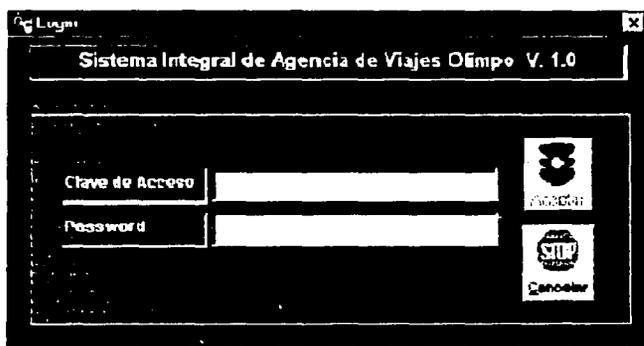


Figura 4.2.

Esto se debe a que tenemos relacionadas las tablas **usuarios => usr_aplic => aplicación**.

Inicialmente se valida la en **usuarios** para determinar si el usuario existe, obtenemos además del nombre el **id_usr** (Identificador del usuario), con el que validamos en la tabla **usr_aplic** (usuarios-aplicaciones), donde checamos el nivel de autorización que tiene el usuario en cuestión, así como el **id_aplic** (identificador de aplicación), con los

que vamos a checar a la tabla **aplicaciones** para determinar a que acceso tiene permiso el usuario.

La pantalla principal de la aplicación será la siguiente:



Figura 4.3.

Como podemos observar en la pantalla principal (Fig.4.3) el sistema cuenta con 6 opciones las cuales son:

Admón.- Esta opción sirve para registrar todos los catálogos necesarios para poder operar correctamente la aplicación, - solo podrán acceder los que tengan nivel de administrador -, entre ellos:

- a) **Agencias.-** Como es una agencia que tiene sucursales en el resto del País. Aquí se darán de alta tanto la matriz como sus sucursales.-

The image shows a screenshot of a software window titled "Catalogo de Agencias". The window has a title bar with a close button (X) on the right. The main content area is titled "CATÁLOGO DE AGENCIAS" in bold, uppercase letters. Below the title, there are three input fields with labels: "Clave:", "Nombre:", and "Estatus del Registro:". Each label is followed by a vertical line that connects to a horizontal line representing the input field. At the bottom of the window, there are two buttons: "Aceptar" and "Cancelar".

Figura 4.4.

- b) **Líneas de autobuses.-** Como todavía no cuenta con una gran cantidad de autobuses tiene que hacer uso de agencias externas que le darán el servicio y en esta opción se dan de alta.

TESIS CON
FALLA DE ORIGEN

Catálogo de Líneas

CATÁLOGO DE LINEAS

Clave de Línea

Clave de Agencia

Nombre

Estatus

Aceptar Cancelar

Figura 4.5.

- c) **Usuarios.**- Es aquí donde se registraran los usuarios que tienen permitido acceder la aplicación.

Catálogo de Usuarios

CATÁLOGO DE USUARIOS

Clave de Agencia

Identificador de Usuario

Apellido Paterno

Apellido Materno

Nombres

Clave de Acceso

Contraseña

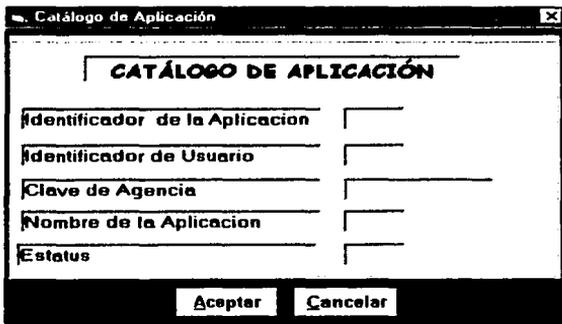
Estatus

Aceptar Cancelar

Figura 4.6.

TESIS CON
FALLA DE ORIGEN

- d) **Aplicaciones.**- Para poder realizar el acceso a la aplicación se tiene que verificar a que aplicación se tiene acceso mediante la clave del usuario:

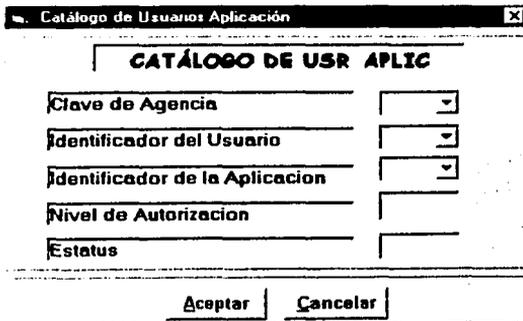


CATÁLOGO DE APLICACIÓN	
Identificador de la Aplicación	<input type="text"/>
Identificador de Usuario	<input type="text"/>
Clave de Agencia	<input type="text"/>
Nombre de la Aplicación	<input type="text"/>
Estatus	<input type="text"/>

Aceptar Cancelar

Figura 4.7.

- e) **Usuarios – Aplicaciones.**- Esta es la unión para determinar a que opción tiene acceso el usuario.



CATÁLOGO DE USR APLIC	
Clave de Agencia	<input type="text"/>
Identificador del Usuario	<input type="text"/>
Identificador de la Aplicación	<input type="text"/>
Nivel de Autorización	<input type="text"/>
Estatus	<input type="text"/>

Aceptar Cancelar

Figura 4.8.

TESIS CON
FALLA DE ORIGEN

- f) **Orígenes.**- Como es una agencia de viajes, se necesita dar de alta todos los orígenes desde donde puede partir un viaje ya establecido.

Orígenes

CATÁLOGO DE ORIGENES

Clave del Origen

Descripción

Status

Aceptar Cancelar

Figura 4.9.

- g) **Destinos.**- Igual que el anterior, es aquí donde se darán de alta los destinos preestablecidos con los que ya trabaja la agencia.

Destinos

CATÁLOGO DE DESTINOS

Clave de Destino

Descripción del Destino

Status del Destino

Aceptar Cancelar

Figura 4.10.

TESIS CON
FALLA DE ORIGEN

- h) **Mercados.**- Como se ha explicado anteriormente el mercado es muy importante y se entiende por mercado la liga entre orígenes y destinos así como la línea que realiza el viaje.

Mercados [X]

CATÁLOGO DE MERCADOS

Clave del Mercado

Clave de Origen

Clave del Destino

Clave de la Agencia

Clave de la Línea

Estatus del Registro

Aceptar Cancelar

Figura 4.11.

- i) **Viajes.**- Es aquí donde se registra la información mas importante de cada viaje, como se podrá ver se registra desde el Origen - Destino, hasta el kilometraje del viaje, hora de salida real, total de pasajeros para este viaje, boletos vendidos, etc.

Viajes [X]

CATÁLOGO DE VIAJES

Número de Viaje Origen

Fecha de Viaje Destino

Hora de Viaje Línea

Agencia Mercado

Hora de Salida Kilometros del Viaje Total de Pasajeros

Vendidos Impte Total Vendidos Status de la Corrida

Aceptar Cancelar

Figura 4.12.

TESIS CON
FALLA DE ORIGEN

- j) **Itinerarios.**- Como existen viajes que pasan por mas de una ciudad donde puede existir una sucursal, se crean los destino de paso, además es aquí donde se puede ir verificando por que ciudad va a pasar el viaje.

CATÁLOGO DE ITINERARIOS

Número de Viaje	<input type="text"/>	Origen	<input type="text"/>
Fecha de Viaje	<input type="text"/>	Destino	<input type="text"/>
Hora de Viaje	<input type="text"/>	Línea	<input type="text"/>
Agencia	<input type="text"/>	Mercado	<input type="text"/>
Origen de Paso	Destino de Paso	Tarifa	Status del Itinerario
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Figura 4.13.

- k) **Hoteles.**- Como una opción importante la agencia tiene convenios con algunos hoteles, es aquí donde se darán de alta para que se pueda realizar la reservación en un futuro del hospedaje. El cual puede ser desde una estrella hasta cinco estrellas.

CATÁLOGO DE HOTELES

Clave de Hotel	<input type="text"/>
Clave de Destino	<input type="text"/>
Nombre	<input type="text"/>
Categoría	<input type="text"/>
Cantidad de Habitaciones	<input type="text"/>
Status del Hotel	<input type="text"/>

TESIS CON
FALLA DE ORIGEN

Figura 4.14.

DESARROLLO

Cabe hacer la aclaración que en esta opción para cada uno de los catálogos se puede realizar además de la alta del registro, la baja, consulta y modificación del mismo.

Gráficas.- Es aquí donde se podrán obtener gráficas de la venta por mercado, por números de viaje o por la venta total, estas gráficas pueden ser por puntos por barras o líneas, la pantalla principal de las gráficas es la siguiente:

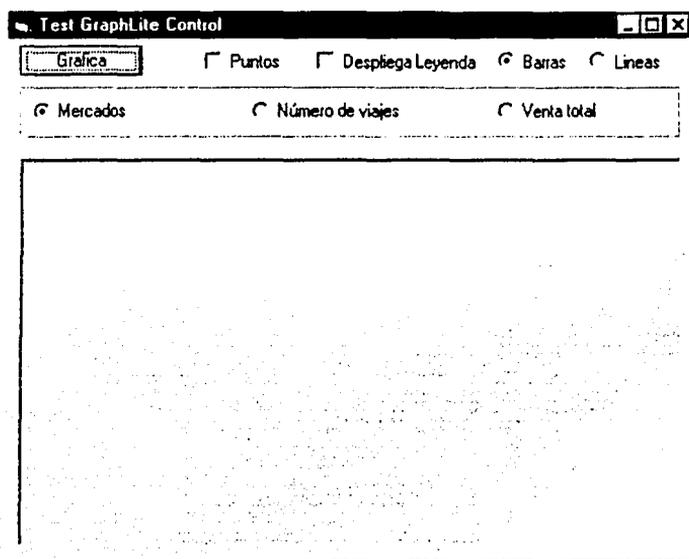


Figura 4.15.

TESIS CON
FALLA DE ORIGEN

Un ejemplo de la gráfica es la siguiente:

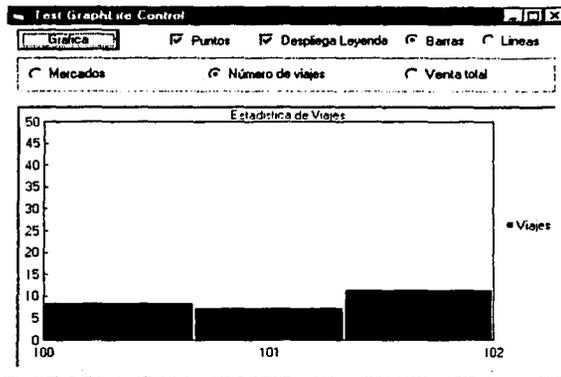


Figura 4.16.

Como presentación también tenemos líneas en lugar de barras y es de la siguiente manera:

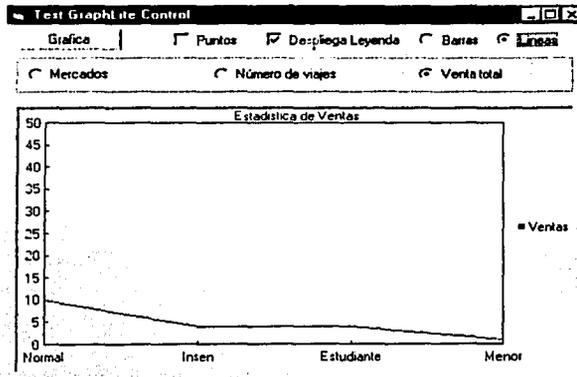


Figura 4.17.

TESIS CON
FALLA DE ORIGEN

Reportes. - Para la agencia es importante tener algunos datos por escrito, de esta necesidad se creo el modulo de reportes el cual consiste en un Reporte de Relación de Pasajeros y el Reporte de Itinerarios de Viajes.:

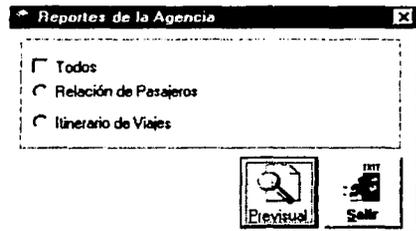


Figura 4.18.

Se puede seleccionar de dos formas los reportes (Relación de pasajeros e Itinerarios de Viajes) estas pueden ser que la impresión se pueda hacer en forma individual o en forma global. Esto quiere decir, por ejemplo si seleccionamos únicamente la opción de Relación de pasajeros nos aparecerá otra opción que nos permite escoger un solo número de viaje de esta forma lograríamos la impresión individual, ya que se imprimirán solo los pasajeros del viaje especificado.

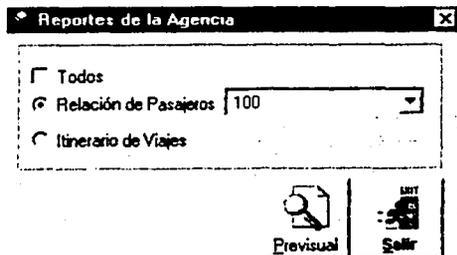


Figura 4.19.

TESIS CON
FALLA DE ORIGEN

En otro caso si seleccionamos la opción Relación de Pasajeros y la opción de Todos tendríamos un reporte global ya que se imprimirán todos los pasajeros de todos los viajes que se encuentre dados de alta.

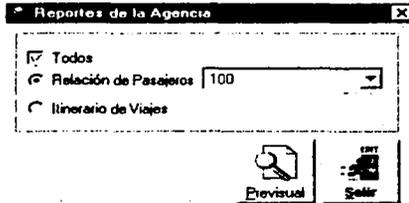


Figura 4.20.

En el reporte de Relación de pasajeros se toman en cuenta los siguientes campos:

No. De viaje ya que para la agencia es una clave de identificación para cada recorrido que realizan sus autobuses es el primer campo que se muestra.

La fecha nos permite saber que días existe mayor demanda para poder prevenir la falta de lugares en determinados días del año y poder dar un mejor servicio, de esta manera se podrá programar mas menos viajes según sea el caso.

La hora al igual que la fecha nos permitirá tener una mejor visión sobre la horas de mayor afluencia de pasajeros.

El origen y destino estos campos especifican el recorrido que se realizara. El Asiento de esta forma podremos llevar un mejor control para evitar la duplicación de los mismos y tener la penosa necesidad de estar reacomodando al pasajero en el mismo momento de abordar.

El nombre es una forma de personalizar el boleto de viaje ya que con ello se pretende un trato directo con el pasajero, siendo este lo mas importante para el servicio que se esta proporcionando.

TESIS CON
FALLA DE ORIGEN

**SISTEMA INTEGRAL DE AGENCIAS DE VIAJES
OLIMPO**

Ejemplo: 1/16/11 11:42:14 p.m.

Reporte de Relación de Pasajeros

No. Viaje	Fecha	Hora	Origen	Destino	Asientos	Nombre
100	2001/02/16	06:00:00 p.m.	DISTRITO FEDERAL	GUANAJUATO	7	VALCA CARRALES CLARISA
100	2001/02/16	06:00:00 p.m.	DISTRITO FEDERAL	GUANAJUATO	9	MENDEZ CARRANZA ARMANDO
100	2001/02/16	06:00:00 p.m.	DISTRITO FEDERAL	GUANAJUATO	10	MENDEZ RICO JOSE DE JESUS
100	2001/02/16	06:00:00 p.m.	DISTRITO FEDERAL	GUANAJUATO	26	LUJA META IVAN
100	2001/02/16	06:00:00 p.m.	DISTRITO FEDERAL	GUANAJUATO	28	UVAL LUN MARI
100	2001/02/16	06:00:00 p.m.	DISTRITO FEDERAL	GUANAJUATO	35	PANIAGUA BORRA URSULA
100	2001/02/16	06:00:00 p.m.	DISTRITO FEDERAL	GUANAJUATO	37	LOPEZ LOPEZ MACARIA
100	2001/02/16	06:00:00 p.m.	DISTRITO FEDERAL	GUANAJUATO	38	RAMIREZ LOZANO CLEMENTE

Figura 4.21.

En el reporte de Itinerario de Viajes se tomaron los siguientes campos:
No. De viaje ya que para la agencia es una clave de identificación para cada recorrido que realizan sus autobuses es el primer campo que se muestra.

La fecha nos permite saber que días existe mayor demanda para poder prevenir la falta de lugares en determinados días del año y poder dar un mejor servicio, de esta manera se podrá programar mas menos viajes según sea el caso.

Agencia como existen varias agencias de viajes, es importante identificar cual de ellas es la que mas vende ya que nos es grato estimular este compromiso de servicio.
Línea es bueno importante identificar cual de nuestras líneas tiene mayor demanda.

El mercado es el que identifica el recorrido que se esta programando y este abarca origen, destino, origen paso y destino de paso.

**TESIS CON
FALLA DE ORIGEN**

SISTEMA INTEGRAL DE AGENCIAS DE VIAJES
OLIMPO

Fecha: 10/07/2006 11:41:54 a.m.

Reporte de Itinerario de Viajes

No. Viaje	Fecha	Bora	Agencia	Línea	Mercado	Origen	Destino	Origen Fase	Destino Fase
100	10/07/2006	00:00 p.m.	AGENCIA DEL D.F.	TRANSPORTES UNIDOS	01	DISTRITO FEDERAL	GUANAJUATO	DISTRITO FEDERAL	GUANAJUATO
100	10/07/2006	00:00 p.m.	AGENCIA DEL D.F.	TRANSPORTES UNIDOS	01	DISTRITO FEDERAL	GUANAJUATO	DISTRITO FEDERAL	GUANAJUATO
100	10/07/2006	00:00 p.m.	AGENCIA DEL D.F.	TRANSPORTES UNIDOS	01	DISTRITO FEDERAL	GUANAJUATO	DISTRITO FEDERAL	GUANAJUATO
100	10/07/2006	00:00 p.m.	AGENCIA DEL D.F.	TRANSPORTES UNIDOS	01	DISTRITO FEDERAL	GUANAJUATO	DISTRITO FEDERAL	GUANAJUATO
100	10/07/2006	00:00 p.m.	AGENCIA DEL D.F.	TRANSPORTES UNIDOS	01	DISTRITO FEDERAL	GUANAJUATO	DISTRITO FEDERAL	GUANAJUATO
100	10/07/2006	00:00 p.m.	AGENCIA DEL D.F.	TRANSPORTES UNIDOS	01	DISTRITO FEDERAL	GUANAJUATO	DISTRITO FEDERAL	GUANAJUATO
100	10/07/2006	00:00 p.m.	AGENCIA DEL D.F.	TRANSPORTES UNIDOS	01	DISTRITO FEDERAL	GUANAJUATO	DISTRITO FEDERAL	GUANAJUATO
100	10/07/2006	00:00 p.m.	AGENCIA DEL D.F.	TRANSPORTES UNIDOS	01	DISTRITO FEDERAL	GUANAJUATO	DISTRITO FEDERAL	GUANAJUATO

Figura 4.22.

Consultas.- En este modulo se muestra la misma información que en el modulo anterior (**Reportes**), solo que en pantalla y su pantalla principal es:

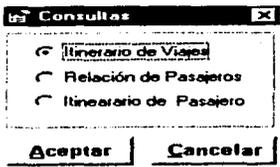


Figura 4.23.

TESIS CON
FALLA DE ORIGEN

Solo se necesita dar el numero de viaje y este se solicita en las pantallas siguientes:

Consulta de Viajes X

CONSULTA DE VIAJES

Número de Viaje	<input type="text"/>	Origen	<input type="text"/>
Fecha de Viaje	<input type="text"/>	Destino	<input type="text"/>
Hora de Viaje	<input type="text"/>	Línea	<input type="text"/>
Agencia	<input type="text"/>	Mercado	<input type="text"/>
Origen de Paso	<input type="text"/>	Destino de Paso	<input type="text"/>

Figura 4.24.

En la consulta de Itinerario de pasajeros se solicita el nombre en la siguiente pantalla:

Consulta por Nombre

Introduzca El Nombre Que Desea Seleccionar

Figura 4.25.

TESIS CON
FALLA DE ORIGEN

Inmediatamente después mostrara en pantalla el itinerario correspondiente al pasajero digitado.

CONSULTA DE PASAJEROS				
No. Viaje	Origen	Destino	Fecha	Hora
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
No. Asiento	Nombre	Apellido Paterno	Apellido Materno	Tipo de Venta
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Figura 4.26.

Ventas.- Este modulo es importante en el sentido de que es en donde se realizan tanto las reservaciones como la venta de boletos de los viajes, la pantalla principal será la siguiente:

Ventas _ □ X

Seleccione una opción y presione Aceptar.

Boletos

Habitaciones

Aceptar
Cerrar

Figura 4.27.

TESIS CON
FALLA DE ORIGEN

En el caso de la venta de los boletos se utiliza lo siguiente:

Venta de Boletos

ORIGEN	DESTINO	FECHA	HORA
▼	▼	▼	▼

ASIENTO	NOMBRE	APELLIDO PATERNO	APELLIDO MATERNO	TIPO DE VENTA
1				▼
2				▼
3				▼
4				▼
5				▼
6				▼
7				▼
8				▼
9				▼
10				▼

1 2	3 4
5 6	7 8
9 10	11 12
13 14	15 16
17 18	19 20
21 22	23 24
25 26	27 28
29 30	31 32
33 34	35 36
37 38	WC
CAFETERIA	

Asiento Seleccionado
 Asientos Vacios
 Asientos Vendidos

Aceptar
Cerrar

Figura 4.27.

Todos los campos que se solicitan van ligados, por lo que después del Origen, se mostraran todos los destinos que estén ligados a ese Origen, posteriormente, se mostraran todas las fechas relacionadas a ese viaje y al final se indicara la hora también ligada al viaje y a la fecha. En pantalla del lado izquierdo se indicaran los lugares ya ocupados y los que estén vacíos. Posteriormente se emitirá en papel el boleto de cada persona que se indique.

En cuanto a la reservación de habitaciones se tiene la pantalla siguiente:

TESIS CON
FALLA DE ORIGEN

DESARROLLO

Por último el administrador tiene la opción de poder realizar el respaldo de la base de datos, esto por seguridad, además el sistema detecta si tiene mas de 10 días de no realizarse un respaldo de la información y al momento de ingresar indica esto, para evitar en lo posible una perdida de información.

**TESIS CON
FALLA DE ORIGEN**

CAPITULO V

CONCLUSION

TESIS CON
FALLA DE ORIGEN

5. CONCLUSION

El sistema de información es la parte medular de la agencia de viajes ya que de este depende que la información se encuentre disponible en cualquier momento de una forma rápida y veraz.

Este proyecto permitió aplicar conceptos para definir los sistemas, la formación de base de datos y redes de computadoras todo aplicado al desarrollo de un sistema de información. Estos conocimientos conjuntamente nos llevan a formular una automatización del Sistema Integral de Agencia de Viajes. Por lo tanto llegamos a las siguientes conclusiones:

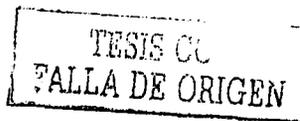
La base de datos es una herramienta muy importante utilizada en el control de datos así mismo en la comercialización de servicios, se desarrollan según las necesidades de las empresas evitando la duplicidad de información y mejorando la circulación de la información para que al personal encargado de manejarla le sea más sencilla la tarea de reservación de boletos.

En este desarrollo comprobamos la necesidad de que se tiene que definir adecuadamente la estructura de la base de datos para que a partir de la misma se pueda implementar una automatización que satisfagan las necesidades de la empresa en nuestro caso la Agencia de Viajes

Como expectativa de servicio y su base de información pretende que se tenga un mejor control en reservar los boletos de los diferentes recorridos y fechas en tiempo real para no vender boletos duplicados así como contar con suficiente información también para el área administrativa y reduciendo los tiempos de respuesta.

Otra de las ventajas encontradas fue que con la automatización del sistema de Agencia de Viajes, se puede consultar en cualquier momento los movimientos que se tenga sobre las reservaciones de boletos

Comprobamos el concepto de **cliente/servidor** ya que fue necesario tener un servidor que contiene toda la información (datos) la cual la pueden consultar diferentes usuarios dependiendo de sus necesidades y desde cualquier computadora que se encuentre en la red.



BIBLIOGRAFIA

Bases de datos con Visual Basic 6. Jeffrey P. McManus. Prentice Hall.

Análisis y Diseño de Sistemas. Kendall y Kendall. Prentice Hall.

Visual Basic 6.0 Manual de programador, Microsoft Corporation.

Bases de Datos Relacionales y el Algebra Relacional. Prof. María de Jesús Berríos.
1999. <http://coqui.metro.inter.edu/~mdejesus/Clas3>.

Introducción a los Sistemas de Bases de Datos Vol. 1 C. J. Date Addison Wesley Iberoamericana S.A.

Networking Essentials Hands-On, Self-Paced Training for Supporting Local and Wide Area Networks.

Visual Basic Programación Cliente/Servidor. Alfonso Gonzalez. Alfaomega Ra-Ma.

SQL Server 7. Gayle Coffman. Osborne Mc Graw-Hill.

Mastering Visual Basic 6 Development , Microsoft Corporation.

TESIS CON
FALLA DE ORIGEN