

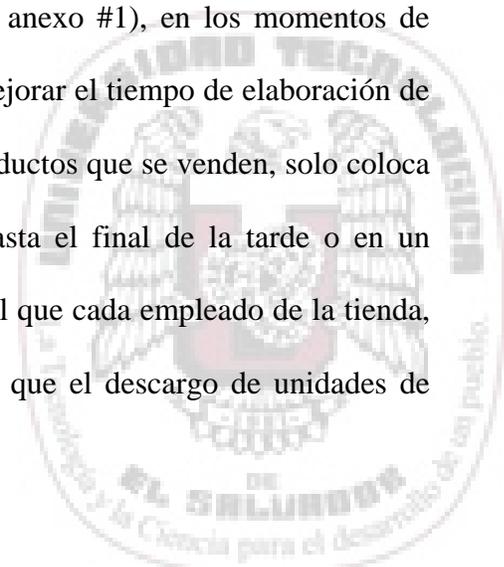
CAPITULO I

MARCO TEORICO

1. SITUACION PROBLEMATICA.

La Escuela Nacional de Agronomía “Roberto Quiñónez” cuenta con una tienda para la venta de productos elaborados en la institución, los cuales son ofertados a la población en general. Muchos de los productos comercializados en la tienda, son productos de consumo con un período entre los 20 y 40 días de vencimiento. Los controles manuales con los que actualmente se maneja la rotación de inventario de dichos productos, permiten que muchos de estos lleguen a su fecha de caducidad, sin tener conocimiento anticipado sobre la cantidad de productos a vencer en determinada fecha.

La elaboración manual de facturas se realiza entre 2 y 10 minutos según sean los diferentes productos que el cliente adquiere (ver anexo #1), en los momentos de mayor afluencia de clientes, para dar agilidad y mejorar el tiempo de elaboración de facturas, el personal de la tienda no detalla los productos que se venden, solo coloca el valor total a cancelar en la factura, siendo hasta el final de la tarde o en un momento en el que no se encuentren clientes, en el que cada empleado de la tienda, detalla los productos que se han vendido. Por lo que el descargo de unidades de



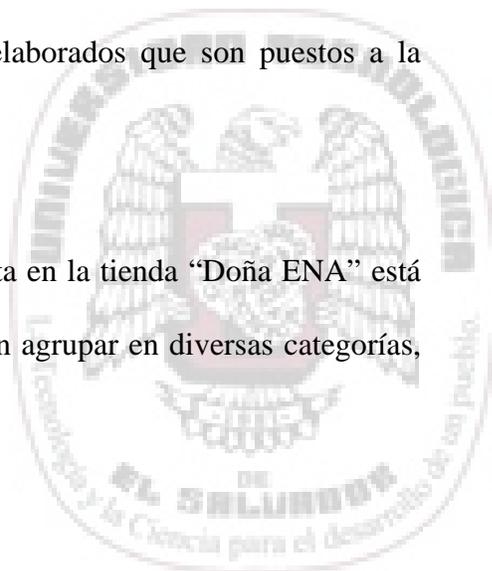
producto de los inventarios, no es preciso y confiable. Se agotan las existencias de algunos productos que se venden en la tienda “Doña ENA”, sin haberse solicitado un reabastecimiento de estos. Se desconoce, estadísticamente, que productos son los de mayor aceptación.

Los controles actuales que se llevan en la tienda “Doña ENA” no proporcionan información exacta para la toma de decisiones, y la elaboración de informes, en algunos casos demora hasta dos días.

2. JUSTIFICACION DE LA INVESTIGACION.

La Escuela Nacional de Agronomía cuenta con controles manuales para administrar los inventarios de productos elaborados por la institución. Sin embargo, la información proporcionada por dichos controles no es confiable. Por medio del desarrollo de una aplicación de software se propondrá un sistema que permita obtener información objetiva y confiable para la toma de decisiones, y llevar un control preciso de los inventarios de productos elaborados que son puestos a la venta en la tienda “Doña ENA”.

El catálogo de productos que son puestos a la venta en la tienda “Doña ENA” está compuesto por 290 productos, los cuales se pueden agrupar en diversas categorías,



entre las cuales se pueden mencionar: lácteos, carnes, vegetales, cereales, jaleas, plantas, semillas y abonos naturales.

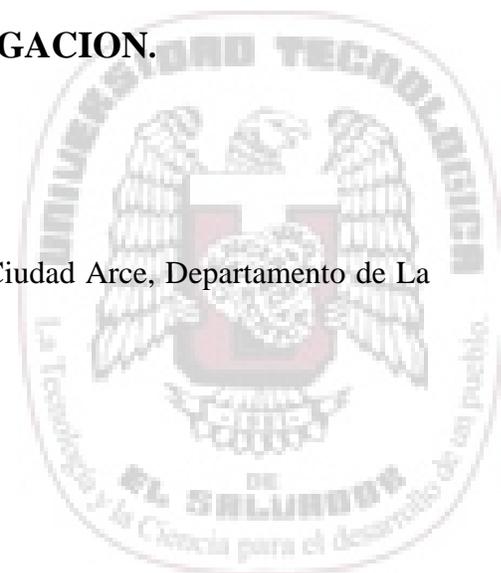
El sistema llevará un registro preciso de los movimientos de inventario de cada uno de los productos del catálogo, y de las fechas de caducidad de los productos que tienen vencimiento, lo que permitirá conocer anticipadamente las existencias en la tienda con corto vencimiento. También, emitirá facturas detalladas por cada venta, registrando así, los ingresos de venta reales de la tienda y las salidas de los productos del inventario.

A través del descargo automático de unidades de productos del inventario por medio de la facturación, se registrarán estadísticas de los productos de mayor aceptación que son elaborados por la E. N. A., se actualizará el kardex de cada producto en la tienda conociendo así la existencia de un determinado producto, lo que permitirá solicitar reabastecimiento anticipado de cada producto próximo a agotarse.

3. DELIMITACION DE LA INVESTIGACION.

3.1 Delimitación espacial

La investigación se realizará en el municipio de Ciudad Arce, Departamento de La Libertad.



3.2 Delimitación geográfica

La investigación se llevará a cabo en la Escuela Nacional de Agronomía “Roberto Quiñónez” (E. N. A.), que se encuentra ubicada en el kilómetro 33 ½ de la carretera Panamericana a Santa Ana, Ciudad Arce, Departamento de La Libertad.

3.3 Delimitación específica

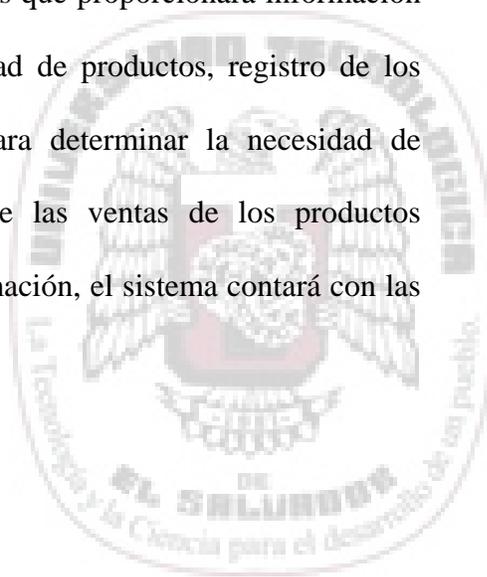
La investigación se desarrollará en la tienda de productos de la Escuela Nacional de Agronomía “Roberto Quiñónez”.

3.4 Delimitación temporal

La investigación de desarrollará entre el mes de Agosto de 2002 y el mes de Julio de 2003.

4. ALCANCES DE LA INVESTIGACIÓN.

Se desarrollará un sistema de control de inventarios que proporcionará información para la toma de decisiones en cuanto a, caducidad de productos, registro de los movimientos de los productos del inventario para determinar la necesidad de reabastecimiento, ingresos obtenidos a través de las ventas de los productos elaborados en la E. N. A. Para la captura de información, el sistema contará con las herramientas siguientes:



Módulo de reabastecimiento: Permitirá ingresar al inventario de la tienda “Doña ENA” los productos elaborados en la E. N. A. que serán puestos a la venta.

Módulo de facturación: Permitirá la elaboración de facturas mecanizadas agilizando el servicio prestado al cliente, registrando los montos de venta y el movimiento en el inventario de la tienda, descargando automáticamente los productos facturados.

Módulo de ajustes: Permitirá la descarga de productos averiados, en mal estado o caducados del inventario.

Módulo de requisición: Permitirá elaborar requisición de productos e insumos a la bodega general de la ENA.

Módulo de mantenimiento: Permitirá la actualización del catálogo de productos, usuarios del sistema, y parámetros generales del sistema.

Se presentará el manual de usuario del sistema, el cual servirá de soporte para los usuarios e introducción al sistema para nuevos usuarios.

Se presentará el manual de instalación del sistema.

Se presentará el manual de procedimientos y políticas del sistema de control de inventarios, cumpliendo así con las normas de control interno de la Corte de Cuentas de la República de El Salvador.



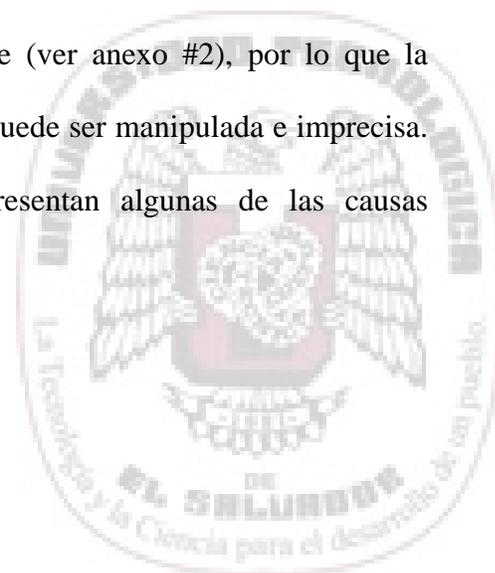
Se propondrá un plan de implantación con la finalidad de que la institución logre los beneficios del sistema desarrollado, poniendo en marcha el nuevo sistema.

5. ENUNCIADO DEL PROBLEMA.

¿Qué necesita la Escuela Nacional de Agronomía “Roberto Quiñónez” (E. N. A.), para lograr un mejor control en el manejo del inventario de productos elaborados por la institución, que se venden en la tienda “Doña ENA”, y así solventar los problemas existentes en el sistema actual?

Causas del problema.

La ineficiencia y deficiencia de los actuales controles de los movimientos de inventarios de productos elaborados por la Escuela Nacional de Agronomía, que se venden en la tienda “Doña ENA”, según el Ing. Francisco Avalos, jefe del departamento de informática de la E. N. A., principalmente se origina por el hecho de que los controles son realizados manualmente (ver anexo #2), por lo que la información proporcionada por dichos controles, puede ser manipulada e imprecisa. Por medio del diagrama de Ishikawa, se representan algunas de las causas detectadas por el Ing. Avalos (ver anexo #3).



El procesamiento de datos actual consume tiempo y esfuerzo del personal de la tienda, por lo que la información que se obtiene no permite conocer la cantidad de productos que caducarán próximamente, sino hasta que estos son detectados y observados por algún dependiente de la tienda.

Se desconoce la existencia actual de un producto determinado, lo que no permite conocer cuando reabastecer de dicho producto el inventario de la tienda.

Para el Ing. Avalos, los controles pueden ser mecanizados, y proporcionar diversos informes objetivos en un menor tiempo y con menor esfuerzo que los actuales; sin embargo, su departamento no cuenta con el suficiente personal para realizar dicho proyecto.

6 OBJETIVOS

6.1 Objetivo general.

Desarrollar una aplicación de software que controle los inventarios de productos elaborados en la Escuela Nacional de Agronomía “Roberto Quiñónez” (E. N. A.), que se comercializan en la tienda “Doña ENA”.



6.2 Objetivos específicos.

Investigar las distintas teorías sobre técnicas y metodologías para el control y manejo de inventarios, y el desarrollo de software.

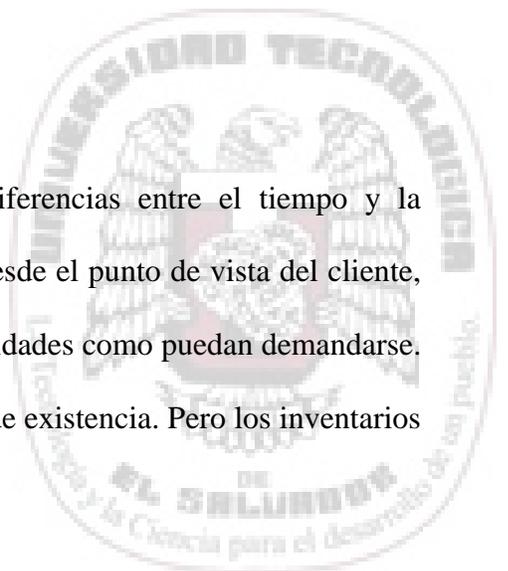
Realizar una investigación de campo que permita identificar la problemática existente y obtener los requerimientos necesarios para controlar los inventarios de productos elaborados en la Escuela Nacional de Agronomía, que son puestos a la venta en la tienda “Doña ENA”.

Elaborar un software que controle los inventarios de productos elaborados que se venden en la tienda “Doña ENA” de la Escuela Nacional de Agronomía “Roberto Quiñónez”.

7. MARCO TEORICO ACTUAL.

7.1 Inventarios.

La necesidad de los inventarios surge de las diferencias entre el tiempo y la localización de la demanda y el abastecimiento. Desde el punto de vista del cliente, el inventario de un artículo debe contener tantas unidades como puedan demandarse. De hecho, un artículo, nunca debería quedar fuera de existencia. Pero los inventarios

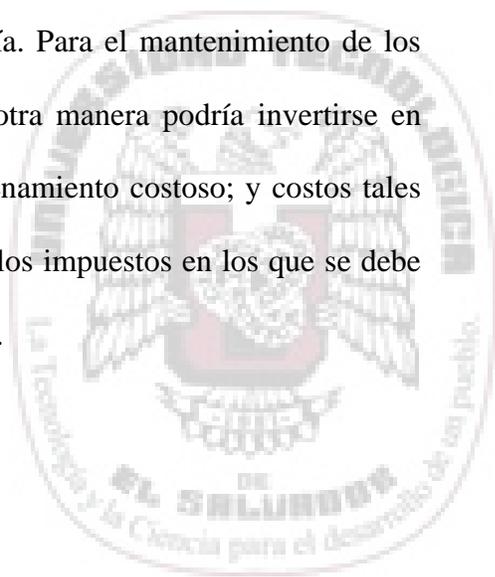


cuestan dinero y representan capital inútil, por lo que es necesario un balance entre los costos de inventario y el servicio al cliente.

Los inventarios proporcionan la flexibilidad de operación que asegura que las operaciones de una organización se realicen sin obstáculos y eficientemente. Con tanta inversión realizada en inventario actualmente, el control adecuado y la administración de ellos pueden traer ahorros considerables a una compañía.

Los inventarios pueden definirse ampliamente como la cantidad de artículos, mercancías y otros recursos económicos que son almacenados o se mantienen inactivos en un instante de tiempo dado. Los recursos económicos varían en cantidad con el tiempo en respuesta al proceso de abastecimiento de demanda que opera para reducir el nivel del inventario y el proceso de abastecimiento que opera para elevarlo.

Los inventarios actúan para reducir los costos y para proporcionarle servicio a los clientes, y evitar faltantes costosos en la mercancía. Para el mantenimiento de los mismos se necesita comprometer capital que de otra manera podría invertirse en cualquier otra cosa. Se requiere espacio de almacenamiento costoso; y costos tales como: seguro, deterioros, obsolescencia, pillaje, y los impuestos en los que se debe incurrir como resultado del mantenimiento de estos.



7.1.1 Tipos de inventarios

Los inventarios pueden clasificarse de varias maneras.

a) Por su grado de transformación:

Materia prima. Materiales utilizados para elaborar componentes de productos terminados tales como aceros, alimentos, maderas, etc.

Componentes. Mezclas de materias primas o partes, listas para utilizarse en el producto terminado o ensamble final.

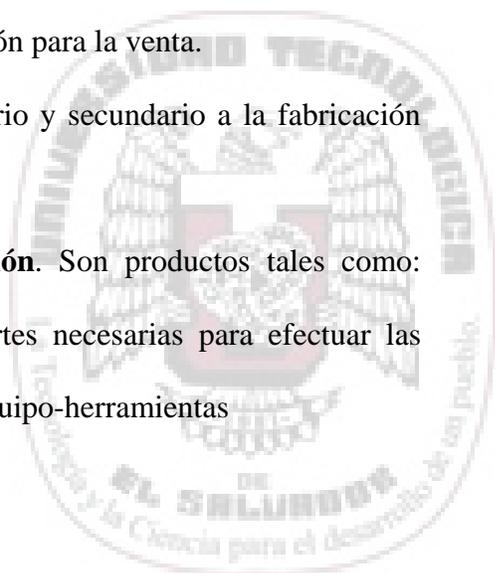
Productos en curso de fabricación. Se trata de materiales y componentes que están experimentando transformaciones, o que están en la planta entre dos operaciones consecutivas.

Productos semiterminados. Son los que han sufrido ya parte de las operaciones de producción y cuya venta no tendrá lugar hasta que no se complete dicho proceso productivo.

Producto Terminado. Material o producto listo para ser distribuido. Es el inventario en un sistema de distribución para la venta.

Subproductos. Son de carácter accesorio y secundario a la fabricación principal.

Materiales para consumo y reposición. Son productos tales como: combustible, material de oficina y partes necesarias para efectuar las funciones de reparación de maquinas-equipo-herramientas



Embalajes y envases. Son los necesarios para el transporte en condiciones adecuadas.

b) Por su categoría funcional:

Inventarios de ciclo. Son aquellos en los que las cantidades de reabastecimiento son superiores a las demandadas, dando así lugar a un inventario que es consumido a lo largo del tiempo¹.

Inventarios de seguridad. Constituidos como protección frente a la incertidumbre de la demanda y del plazo de entrega del pedido. Evitando, dentro de lo posible, la inexistencia de inventarios en un momento dado.

Inventarios estacionales. Su objeto es hacer frente a un aumento esperado de las ventas a un determinado período de tiempo. Por ejemplo, el material escolar a principios de año que acopia una papelería situada junto a un colegio.

Inventarios en tránsito. Se denomina así a los artículos que están circulando entre las diferentes fases de producción y de distribución, como, por ejemplo, entre el almacén de productos terminados y un almacén regional de distribución, o entre distintas fases del montaje.

¹ Curso de gestión de almacenes, <http://www.femz.es/cursos/Almacenes/>



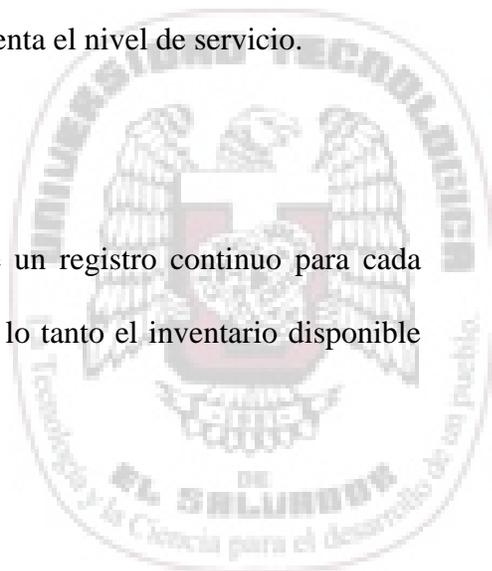
7.1.2 Sistemas de administración de inventarios.

Un buen sistema de administración de inventarios es capaz de describir y analizar las decisiones para determinar niveles de inventario, debe incluir reglas de decisión basados en modelos matemáticos, adicionalmente una estructura organizacional y procedimientos. El éxito de la administración de inventarios depende en un alto grado de la información que utiliza para desarrollar su actividad, es por eso la importancia de manejar registros exactos. Si los registros no son exactos dará como resultado la existencia de subsistemas de expeditación para evitar faltantes en vez de utilizar la planeación de materiales como el sistema que permite tener el componente correcto en el momento correcto.

El tener registros exactos es indispensable para cualquier sistema formal de planeación, lo cual tendrá como beneficios el tener una mayor productividad en el almacén, teniendo una disminución de perdidas de material por obsolescencia o desperdicio, además, se reduce el proceso de preparación de una orden de producción y como consecuencia de esto se incrementa el nivel de servicio.

Sistema de inventario perpetuo

En el sistema de inventario perpetuo se mantiene un registro continuo para cada artículo del inventario. Los registros muestran por lo tanto el inventario disponible



todo el tiempo. Este sistema ofrece un alto grado de control, porque los registros están siempre actualizados².

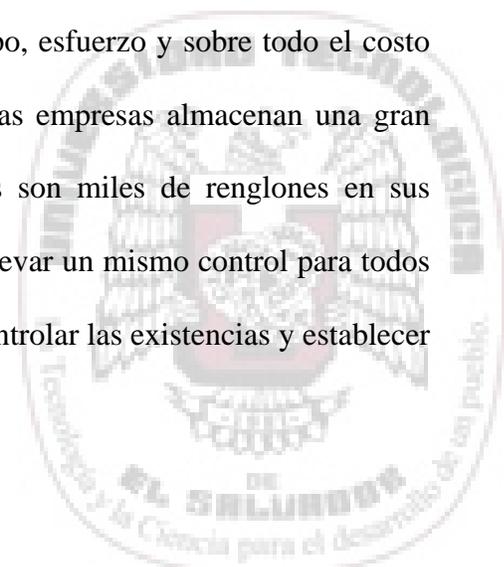
Sistema de inventario periódico

En el sistema de inventario periódico no se mantiene un registro continuo del inventario disponible, mas bien, al finalizar el período, se hace un conteo físico del inventario disponible y aplica los costos unitarios para determinar el costo del inventario final. El sistema periódico es conocido también como sistema físico, porque se apoya en el conteo físico real del inventario. Este sistema es generalmente utilizado para contabilizar los artículos del inventario que tienen un costo unitario bajo. Para utilizar este sistema de inventario, se debe tener la capacidad de controlar el inventario mediante la inspección visual.

Sistema de clasificación A. B. C.

Este sistema tiene como finalidad reducir el tiempo, esfuerzo y sobre todo el costo en el control de los inventarios. En la práctica, las empresas almacenan una gran cantidad de materiales que muchas de las veces son miles de renglones en sus inventarios, por lo que encuentran incosteable el llevar un mismo control para todos ellos. El costo tiempo y esfuerzo que implica el controlar las existencias y establecer

² El inventario, <http://www.monografias.com/trabajos10/inve/inve.shtml>

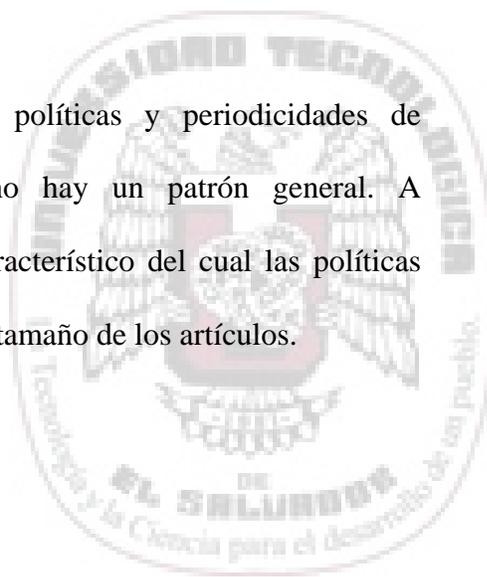


logísticamente las políticas de reabastecimiento lo dedican únicamente a una pequeña porción del total de los renglones del inventario, los cuales engloban la mayor parte del valor total en dinero que suma el inventario. En cambio, resulta incosteable llevar a cavo el mismo control con los elementos del inventario que suman poca inversión y que en la mayoría de los casos representan la mayoría de los artículos del inventario. De ahí parte el principio de separar los renglones de un inventario, según su valor de importancia, en tres clases:

- A. Incluye los artículos que por su alto costo de adquisición, por su alto nivel en el inventario merece el 100% de control.
- B. Comprende aquellos artículos que por ser de menor costo, valor, su control requiere menos esfuerzo y más bajo costo administrativo.
- C. Integrado por los artículos de poco costo, poca inversión y que requieren poca supervisión sobre el nivel de sus existencias.

Clasificación por precio unitario

Cada empresa establece rangos de precio, políticas y periodicidades de adquisición según sus necesidades, pues no hay un patrón general. A continuación se da un ejemplo de un caso característico del cual las políticas fueron tomadas según los rangos de precio y el tamaño de los artículos.



Clase	Condición	Políticas
A ₁	Alto precio unitario y tamaño muy voluminoso	Compra semanal o solamente lo requerido por la producción programada
A ₂	Alto precio unitario y tamaño poco voluminoso	Compra mensual para reponer el consumo al llegar al mínimo
B ₁	Precio mediano y tamaño muy voluminoso	Compra cada dos meses
B ₂	Precio mediano y tamaño poco voluminoso	Compra cada cuatro meses
C ₁	Bajo precio unitario y tamaño muy voluminoso	Compra para seis meses de consumo
C ₂	Bajo precio unitario y tamaño poco voluminoso	Compra cada ocho meses o una vez al año

Clasificación por utilización y valor

Se hace la división del inventario en tres clases A, B, C de acuerdo a su valor o costo unitario y uso ya sea anual, mensual o como se establezca.

- La clasificación A representa el 80% del costo y el 20% de los artículos
- La clasificación B representa el 15% del costo y el 30% de los artículos
- La clasificación C representa el 5% del costo y el 50% de los artículos

7.1.3 Rotación de inventarios.

La rotación de los inventarios es un cálculo de las veces que una materia o producto tienen que ser reabastecidos por compras o su fabricación en la planta o el número de veces que se agotan los inventarios para fabricar un producto.



Esta información se obtiene de los inventarios y del registro de las compras realizadas en un período determinado. Existen varios métodos para medir la rotación de los inventarios:

Método de cálculo de la razón

En este caso el costo de los productos vendidos se divide entre la inversión de los inventarios. Ejemplo:

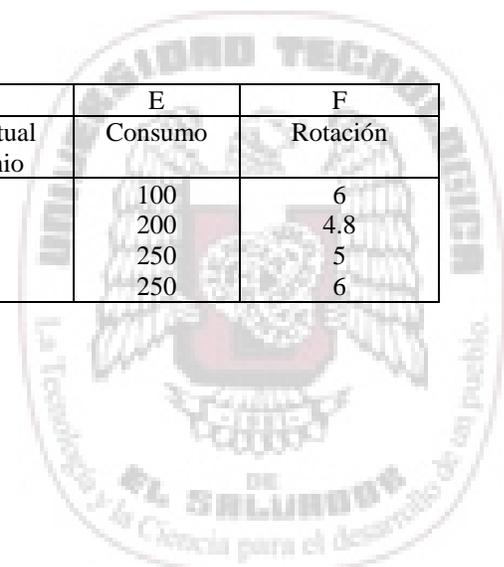
$$R = \frac{\text{Ventas}}{\text{Inventario}} = \frac{\$ 3,600}{\$ 600} = 6 \text{ veces}$$

Método de rotación mensual

En este caso el número de unidades consumidas durante el mes es dividido entre el número de unidades disponibles al principio del mismo, y esta razón multiplicada por 12, a fin de anualizar la relación, de la rotación anual de cada producto.

Ejemplo:

A	B	C	D	E	F
Nombre	Clave	Mes anterior 31/Mayo	Mes actual 30/Junio	Consumo	Rotación
Papel bond	Pb 25	200	100	100	6
Papel bond	Pb 35	500	300	200	4.8
Sobres c.	Sc 20	600	350	250	5
Carpetas	Cr 40	500	250	250	6



Método del período anual.

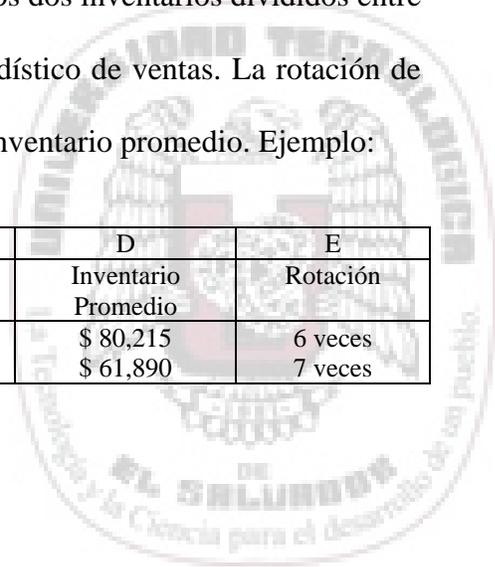
En este método la rotación se calcula dividiendo la cantidad vendida, según el record de ventas, entre un inventario promedio. El inventario promedio se estima al promediar la cantidad disponible al principio del año y la cantidad disponible al fin del mismo. La relación anual de rotación es simplemente, el número de unidades vendidas dividido entre el inventario promedio. Ejemplo:

A	B	C	D	E	F	G
Nombre	Clave	Anterior 29/02/2000	Actual 31/03/2001	Vendido	Promedio	Rotación
Papel B.	Pb 25	2,200	3,400	28,000	2,800	10
Papel B.	Pb 35	6,000	5,000	38,500	5,500	7
Sobres C.	Sc 20	8,100	10,900	38,500	9,500	4
Sobres O.	So 30	7,500	7,000	57,000	7,250	8

Cálculo de rotación anual del inventario total de existencias

Este cálculo se hace con valor monetario, independientemente de los cálculos por unidades hechos artículo por artículo en los métodos anteriores. Se muestra el costo de ventas, el inventario inicial y el inventario final para un período de 12 meses. El inventario promedio es el resultado de la suma de los dos inventarios divididos entre 2, el costo de lo vendido se tomó del registro estadístico de ventas. La rotación de inventarios, es el costo de ventas dividido entre el inventario promedio. Ejemplo:

A	B	C	D	E
Costo de lo vendido 97/98	Inventario Inicial	Inventario Final	Inventario Promedio	Rotación
\$ 485,320	\$ 79,545	\$ 80,885	\$ 80,215	6 veces
\$ 450,150	\$ 65,330	\$ 58,450	\$ 61,890	7 veces



7.1.4 Costeo de inventarios.

El costo del inventario es el precio que el negocio para adquirir el inventario, incluye el precio de factura, menos cualquier descuento de compras, más el impuesto sobre ventas, aranceles, cargos por transporte, seguros mientras está en tránsito y todos los costos en que se incurre para lograr que las mercancías, estén disponibles para la venta³.

Los métodos de costeo de inventarios son:

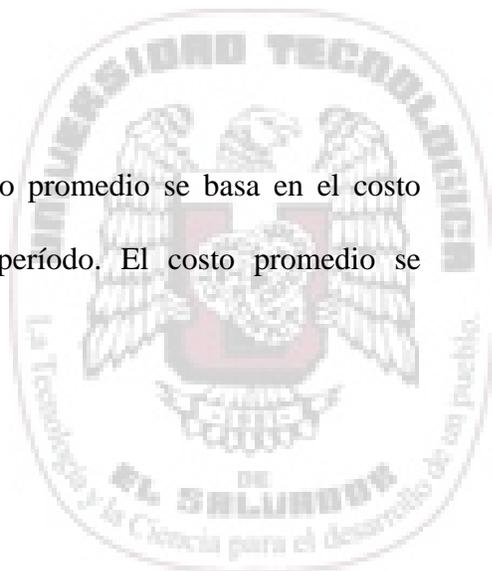
Costo unitario específico

Algunas empresas tratan con artículos de inventario que pueden identificarse de manera individual, como los automóviles. Estas empresas costean, por lo general, su inventario al costo específico de la unidad en particular.

Costo promedio ponderado

Este método, llamado a menudo método del costo promedio se basa en el costo promedio ponderado del inventario durante el período. El costo promedio se

³ El inventario, <http://www.monografias.com/trabajos10/inve/inve.shtml>



determina de la manera siguiente: divida el costo de las mercancías disponibles para la venta (inventario inicial + compras) entre el número de unidades disponibles.

Costo de primeras entradas, primeras salidas (PEPS)

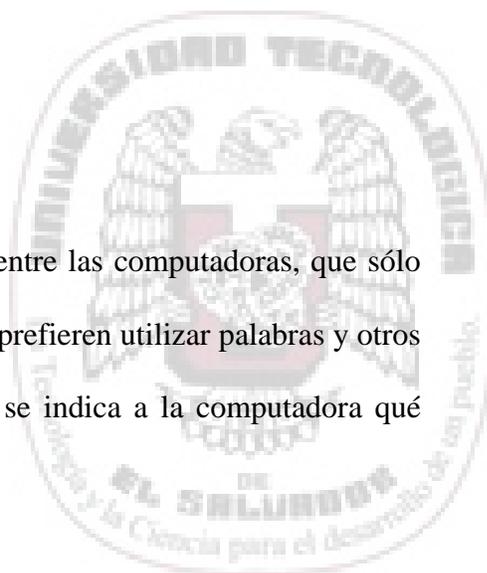
Bajo este método, se debe llevar un registro del costo de cada unidad comprada del inventario. El costo de la unidad utilizado para el inventario final, puede ser diferente de los costos unitarios utilizados para calcular el costo de las mercancías vendidas. Bajo PEPS, los primeros costos que entran al inventario son los primeros que salen al costo de las mercancías vendidas. El inventario final se basa en los costos de las compras más recientes.

Costo de últimas entradas, primeras salidas (UEPS)

En este método depende también de los costos por compras de un inventario en particular. Los últimos costos que entran al inventario son los primeros que salen al costo de mercancías vendidas. Este método deja los costos mas antiguos en el inventario final.

7.2 Lenguajes de programación.

Los lenguajes de programación cierran el abismo entre las computadoras, que sólo trabajan con números binarios, y los humanos, que prefieren utilizar palabras y otros sistemas de numeración. Mediante los programas se indica a la computadora qué



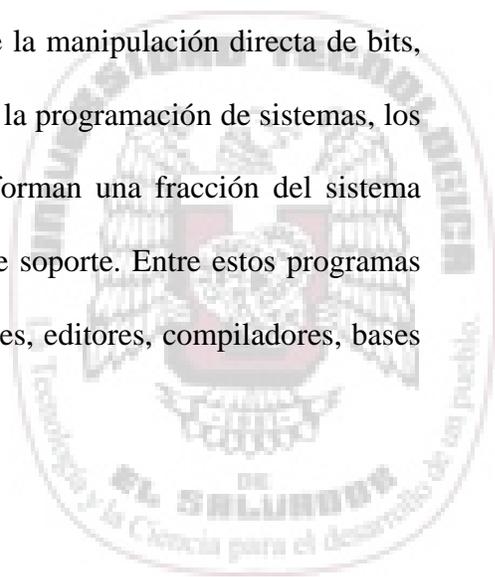
tarea debe realizar y como efectuarla, pero para ello es preciso introducir estas ordenes en un lenguaje que el sistema pueda entender.

Lenguaje ensamblador

El lenguaje ensamblador usa una representación simbólica del lenguaje binario real que la computadora ejecuta directamente, es un lenguaje de bajo nivel. Aunque es un lenguaje que da a los programadores mucha potencia de realizar tareas con mucha flexibilidad y eficiencia, es notoriamente difícil de trabajar con el en el desarrollo y depuración de programas. Su falta de estructuración hace que los programas sean difíciles de leer, de mejorar y de mantener.

Lenguaje C

El lenguaje C nació en los laboratorios Bell de AT&T y ha sido estrechamente asociado con el sistema operativo UNIX, ya que su desarrollo se realizó en este sistema. A menudo se denomina al lenguaje C como un lenguaje de nivel medio, ya que combina elementos de lenguajes de alto nivel con la funcionalidad de un lenguaje de bajo nivel. Es un lenguaje que permite la manipulación directa de bits, bytes y direcciones. Inicialmente fue utilizado para la programación de sistemas, los cuales eran una amplia clase de programas que forman una fracción del sistema operativo de la computadora o de sus utilidades de soporte. Entre estos programas de sistemas tenemos: sistemas operativos, intérpretes, editores, compiladores, bases de datos, hojas de cálculo, etc.



C++

C++ es un lenguaje de programación de propósito general basado en el lenguaje de programación C. Este fue diseñado para: ser mejor que C, soportar la abstracción de datos y soportar la programación orientada a objetos. Su nombre se debe al carácter evolutivo de las transformaciones de C.

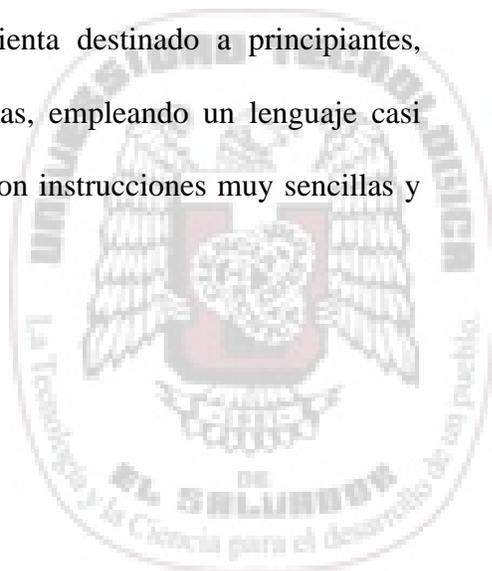
COBOL

(Common Business Oriented Language)

Se convirtió en la época de los 70, en el lenguaje de programación estándar para aplicaciones administrativas tanto en la industria como en actividades estatales. Es un lenguaje de alto nivel y fue diseñado para que los no programadores pudieran leer y comprender el programa.

BASIC

El lenguaje de programación BASIC (Beginner's All purpose Symbolic Instruction Code) nació en el año 1964 como una herramienta destinado a principiantes, buscando una forma sencilla de realizar programas, empleando un lenguaje casi igual al usado en la vida ordinaria (en inglés), y con instrucciones muy sencillas y escasas.



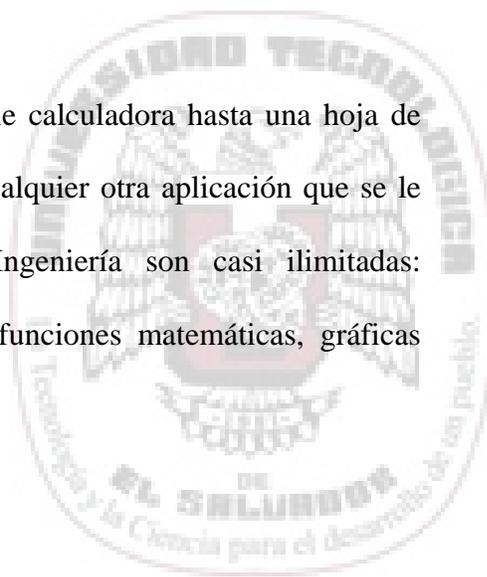
JAVA

Java es un lenguaje de programación orientada a objetos que fue desarrollado por Sun Microsystems, elaborado con base en C++. Fue diseñado para ser pequeño, simple y transportable entre plataformas y sistemas operativos, tanto a nivel código fuente como binario. Se caracteriza por poder crear applets, los cuales pueden usarse para crear animación, figuras o áreas que pueden responder a peticiones del usuario, juegos u otros efectos interactivos.

Visual Basic

Visual Basic es una herramienta de programación que permite crear aplicaciones propias para Windows, en la que éstas se desarrollan en una gran parte a partir del diseño de una interfaz gráfica. En una aplicación Visual Basic, el programa está formado por una parte de código puro (BASIC), y otras partes asociadas a los objetos que forman la interfaz gráfica. Es por tanto un término medio entre la programación tradicional, formada por una sucesión lineal de código estructurado, y la programación orientada a objetos.

Con Visual Basic se puede crear desde una simple calculadora hasta una hoja de cálculo, pasando por un procesador de textos o cualquier otra aplicación que se le ocurra al programador. Sus aplicaciones en Ingeniería son casi ilimitadas: representación de movimientos mecánicos o de funciones matemáticas, gráficas termodinámicas, simulación de circuitos, etc.



7.3 Bases de datos.

Base de Datos es un conjunto exhaustivo no redundante de datos estructurados organizados independientemente de su utilización y su implementación en máquina, accesibles en tiempo real y compatibles con usuarios concurrentes con necesidad de información diferente y no predicable en tiempo. Algunas ventajas del uso de las bases de datos son:

Independencia de datos y tratamiento, el cambio en datos no implica cambio en programas y viceversa.

Coherencia de resultados. Reduce redundancia: por medio de acciones lógicamente únicas y evita inconsistencia.

Mejora en la disponibilidad de datos

Cumplimiento de ciertas normas. La base de datos posee restricciones de seguridad, tales como: accesos (usuarios a datos) y operaciones (operaciones sobre datos).

7.3.1 Tipos de modelos de datos.

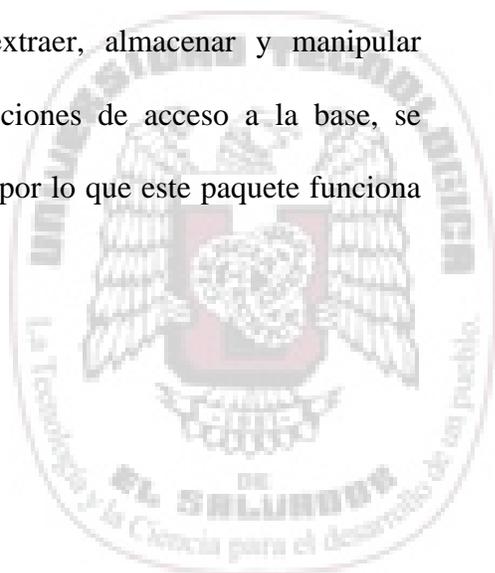
Existen fundamentalmente tres alternativas disponibles para diseñar las bases de datos:



- El modelo jerárquico puede representar dos tipos de relaciones entre los datos: relaciones de uno a uno y relaciones de uno a muchos.
- El modelo de red permite la representación de muchos a muchos, de tal forma que cualquier registro dentro de la base de datos puede tener varias ocurrencias superiores a él. El modelo de red evita redundancia en la información, a través de la incorporación de un tipo de registro denominado el conector.
- El modelo relacional, se está empleando con más frecuencia en la práctica, debido a la ventajas que ofrece sobre los dos modelos anteriores, entre ellas, el rápido entendimiento por parte de usuarios que no tienen conocimientos profundos sobre sistemas de bases de datos.

7.3.2 Sistema manejador de base de datos (DBMS).

Un DBMS es una colección de numerosas rutinas de software interrelacionadas, cada una de las cuales es responsable de una tarea específica. El objetivo primordial de un sistema manejador base de datos es proporcionar un contorno que sea a la vez conveniente y eficiente para ser utilizado al extraer, almacenar y manipular información de la base de datos. Todas las peticiones de acceso a la base, se manejan centralizadamente por medio del DBMS, por lo que este paquete funciona como interfase entre los usuarios y la base de datos.



Los sistemas manejadores de bases de datos o gestores de bases de datos permiten que varios usuarios hagan operaciones sobre ella al mismo tiempo: uno puede hacer una consulta al mismo tiempo que otro, situado en un lugar diferente, está introduciendo datos en la base. Gestores de este tipo son: *Oracle*, *PLA*, *DB2* o *SQL Server*, que está pensados únicamente para este uso y no se emplean para bases de datos personales. *Access*, originariamente pensado para uso personal, tiene capacidad de red que hace de él, un programa muy apto para su empleo en bases de datos de pequeñas empresas, que no necesitan un número de accesos simultáneos muy alto.

7.3.3 Utilidad de una base de datos.

Las tres cosas básicas que debe permitir un gestor de base de datos son: introducir datos, almacenarlos y recuperarlos. Al mismo tiempo permiten otra serie de funciones que hacen de ellos herramientas incomparablemente superiores a los métodos tradicionales de almacenamiento de datos: archivadores, carpetas, etc.

Cualquier gestor debe permitir: ordenar los datos, realizar búsquedas, mostrar distintas vistas de los datos, realizar cálculos sobre ellos, resumirlos, generar informes a partir de ellos, importarlos y exportarlos.



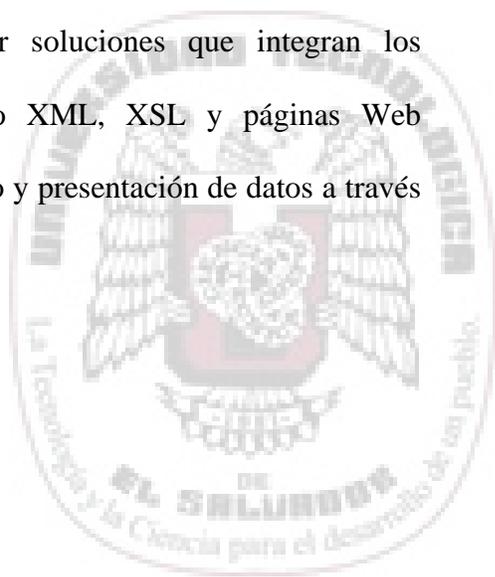
7.3.4 Access.

Access es un programa para la gestión de información. Permite diseñar las estructuras para almacenar la información y los medios para su introducción y explotación. Access es un sistema gestor de bases de datos relacional⁴ (RDBMS). Incluye un comprensivo lenguaje de programación, Microsoft Visual Basic for Applications (VBA), que puede ser utilizado para crear robustas aplicaciones que pueden ser compartidas por muchos usuarios. Soporta objetos OLE y controles ActiveX.

Las mas reciente versión de Access, incluye el novedoso Motor de Escritorio SQL Server 2000 (SQL Server 2000 Desktop Engine), esta herramienta, es completamente compatible con SQL Server y permite a los usuarios fácilmente crear y modificar bases de datos compatibles con SQL Server, de manera similar a que los usuarios desarrollen su base de datos en SQL Server.

Access proporciona herramientas para construir soluciones que integran los estándares de internet, herramientas tales como XML, XSL y páginas Web dinámicas, que permiten mejorar el compartimiento y presentación de datos a través de intranets e internet.

⁴ Viescas, John L. Running Microsoft Access 97, Pág. 3 - 18



7.3.5 SQL Server.

SQL Server es el RDBMS de elección para una amplia gama de clientes corporativos y Proveedores Independientes de Software. SQL Server disminuye el costo total de propiedad a través de características como administración multi-servidor y con una sola consola; ejecución y alerta de trabajos basadas en eventos; seguridad integrada; y scripting administrativo⁵.

Algunas características de SQL Server son:

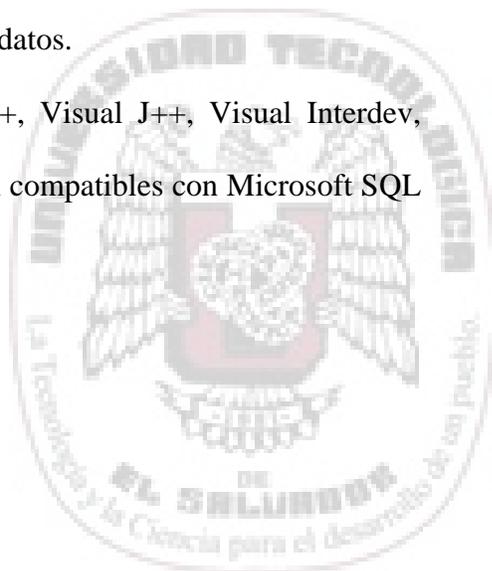
Escalabilidad: Se adapta a las necesidades de la empresa, soportando desde unos pocos usuarios a varios miles. Empresas centralizadas u oficinas distribuidas, replicando cientos de sites.

Potencia: Microsoft SQL Server es la mejor base de datos para Windows NT Server. Posee los mejores registros de los benchmarks independientes (TCP) tanto en transacciones totales como en coste por transacción.

Gestión: Con un completo interfaz gráfico que reduce la complejidad innecesaria de las tareas de administración y gestión de la base de datos.

Orientada al desarrollo: Visual Basic, Visual C++, Visual J++, Visual Interdev, Microfocus Cobol y muchas otras herramientas son compatibles con Microsoft SQL Server.

⁵ Monografías: Programación orientada a objetos, Oracle Y SQL Server,
<http://www.monografias.com/trabajos4/basesdatos/basesdatos.shtml>



7.3.6 ORACLE.

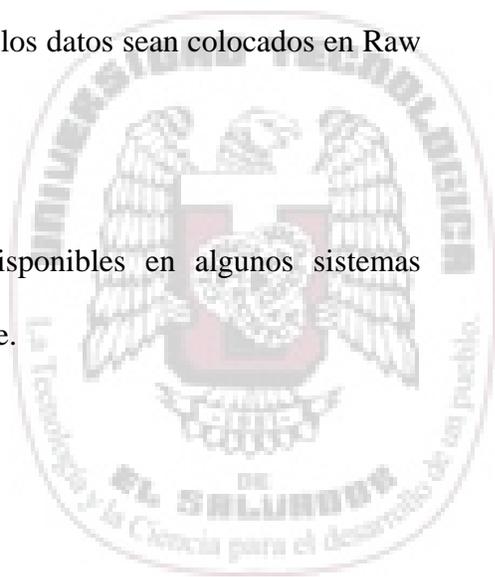
Oracle es el mayor y más usado Sistema Manejador de Base de Dato Relacional (RDBMS) en el mundo, hace uso de los recursos del sistema informático en todas las arquitecturas de hardware, para garantizar su aprovechamiento al máximo en ambientes cargados de información. La Corporación Oracle ofrece este RDBMS como un producto incorporado a la línea de producción.

Oracle corre en computadoras personales (PC), microcomputadoras, mainframes y computadoras con procesamiento paralelo masivo. Soporta unos 17 idiomas, corre automáticamente en más de 80 arquitectura de hardware y software distinto sin tener la necesidad de cambiar una sola línea de código. Esto es porque más del 80% de los códigos internos de Oracle son iguales a los establecidos en todas las plataformas de sistemas operativos⁶.

Oracle soporta dos tipos de almacenamiento, por caracter (RAW) o por bloques (Files System), generalmente es recomendable que los datos sean colocados en Raw Device.

Raw device: es un dispositivo de caracteres disponibles en algunos sistemas operativos el cual es asignado directamente a Oracle.

⁶ Monografías: Programación orientada a objetos, Oracle Y SQL Server,
<http://www.monografias.com/trabajos4/basesdatos/basesdatos.shtml>



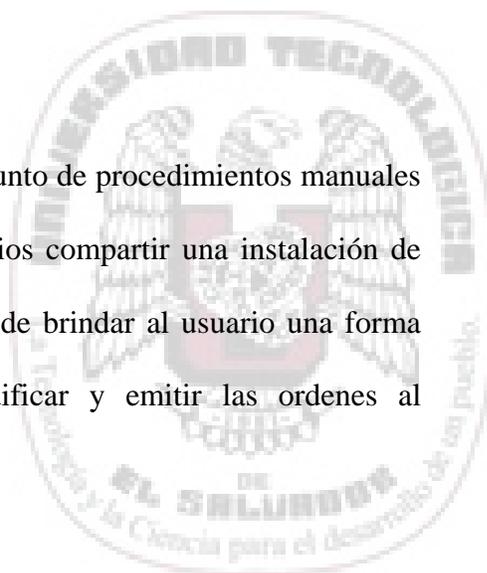
Oracle corre más rápidamente con Raw Device que con *Files System*, por varias razones:

- El I/O (Input/Output) es realizado directamente en el disco por Oracle, independientemente del sistema operativo.
- El buffer cache del sistema del sistema operativo es dejado a un lado.
- Los buffers del sistema operativo y de oracle son independiente entre sí.

7.4 Sistemas operativos.

El objetivo fundamental de los sistemas de computación es ejecutar los programas de los usuarios y facilitar la resolución de sus problemas. El hardware se construye con este fin, pero como este no es fácil de utilizar, se desarrollan programas de aplicación que requieren ciertas operaciones comunes, como el control de dispositivos de entrada y salida. Las funciones comunes de control y de asignación de recursos se integran para formar un solo fragmento de software: el sistema operativo.

Un sistema operativo (SO) se define como un conjunto de procedimientos manuales y automáticos, que permiten a un grupo de usuarios compartir una instalación de computadora eficazmente. El SO es el encargado de brindar al usuario una forma amigable y sencilla de operar, interpretar, codificar y emitir las ordenes al

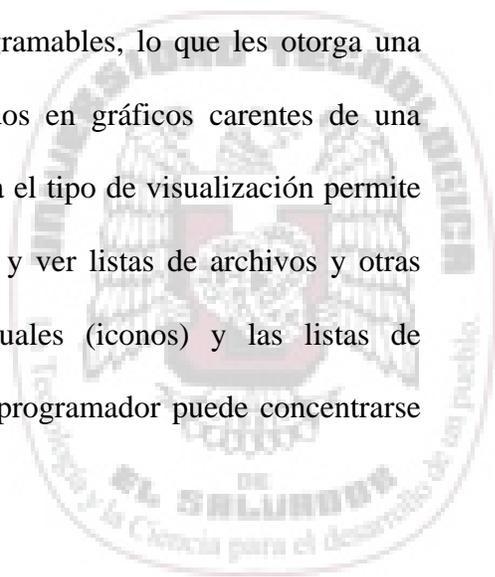


procesador central para que este realice las tareas necesarias y específicas para completar una orden. Cuando se enciende una computadora, lo primero que ésta hace es llevar a cabo un autodiagnóstico en el que identifica su memoria, sus discos, su teclado, su sistema de vídeo y cualquier otro dispositivo físico conectado a ella. Posteriormente, la computadora busca un SO para arrancar.

Una vez que la computadora ha puesto en marcha su SO, mantiene al menos parte de éste en su memoria en todo momento. Mientras la computadora esté encendida, el SO tiene 4 tareas principales:

- Proporcionar interfaz al usuario, ya sea de línea de comando o gráfica, para que este se pueda comunicar con la computadora.

En la interfaz de línea de comando: se caracteriza porque el usuario para comunicarse con el sistema operativo debe escribir comandos utilizando un lenguaje de comandos especial. Estos sistemas se consideran más difíciles de aprender y utilizar que los de las interfaces gráficas. Sin embargo, los sistemas basados en comandos son por lo general programables, lo que les otorga una flexibilidad que no tienen los sistemas basados en gráficos carentes de una interfaz de programación. En la interfaz gráfica el tipo de visualización permite al usuario elegir comandos, iniciar programas y ver listas de archivos y otras opciones utilizando las representaciones visuales (iconos) y las listas de elementos del menú. En la interfaz gráfica el programador puede concentrarse



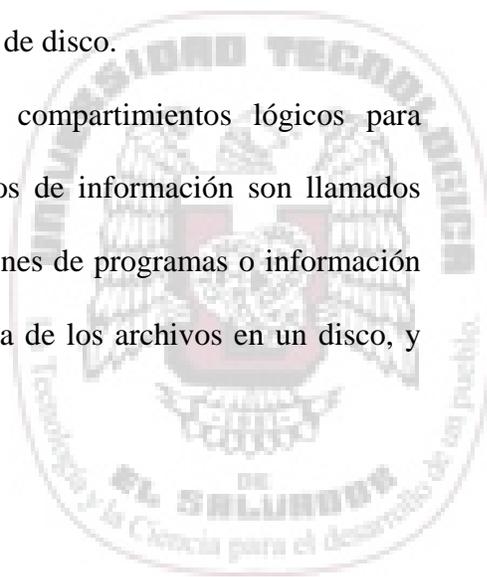
en la funcionalidad, ya que no esta sujeto a los detalles de la visualización ni a la entrada a través del mouse o el teclado. También permite a los programadores crear programas que realicen de la misma forma las tareas más frecuentes, como guardar un archivo, porque la interfaz proporciona mecanismos estándar de control como ventanas y cuadros de diálogo. Otra ventaja es que las aplicaciones escritas para una interfaz gráfica de usuario son independientes de los dispositivos: a medida que la interfaz cambia para permitir el uso de nuevos dispositivos de entrada y salida, como un monitor de pantalla grande o un dispositivo óptico de almacenamiento, las aplicaciones pueden utilizarlos sin necesidad de cambios.

- Administrar los dispositivos de hardware en la computadora.

Cuando se ejecutan los programas, se necesita utilizar la memoria, el monitor, las unidades de disco, los puertos de entrada y salida (impresoras, módems, etc.). El SO sirve de intermediario entre los programas y el hardware.

- Administrar y mantener los sistemas de archivo de disco.

Los SO agrupan la información dentro de compartimientos lógicos para almacenarlos en distintos medios. Estos grupos de información son llamados archivos, los cuales pueden contener instrucciones de programas o información creada por el usuario. El SO mantiene una lista de los archivos en un disco, y



proporciona las herramientas necesarias para organizar y manipular estos archivos.

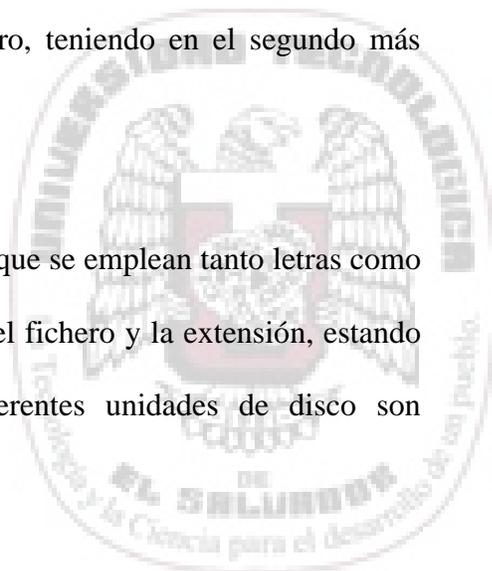
- Apoyar a otros programas.

Otra de las funciones importantes del SO es proporcionar servicios a otros programas. Por ejemplo, listar archivos, grabarlos a disco, eliminar archivos, revisar espacio disponible, etc.

7.4.1 MS-DOS.

El significado de estas letras es el de Microsoft Disk Operating System. Microsoft es el nombre de la compañía que diseñó este sistema operativo, e IBM la compañía que lo hizo estándar al adoptarlo en sus microordenadores. Este SO emplea discos flexibles con una organización determinada. Los discos se pueden grabar por una o por dos caras y la información se organiza en 40 pistas de 8 ó 9 sectores de un tamaño de 512 caracteres, reservándose el sistema para la propia información del disco, que puede ser disco removible o disco duro, teniendo en el segundo más capacidad pero similar estructura.

Los nombres de los ficheros en MS-DOS, para los que se emplean tanto letras como números, se componen de dos partes: el nombre del fichero y la extensión, estando ambos datos separados por un punto. Las diferentes unidades de disco son

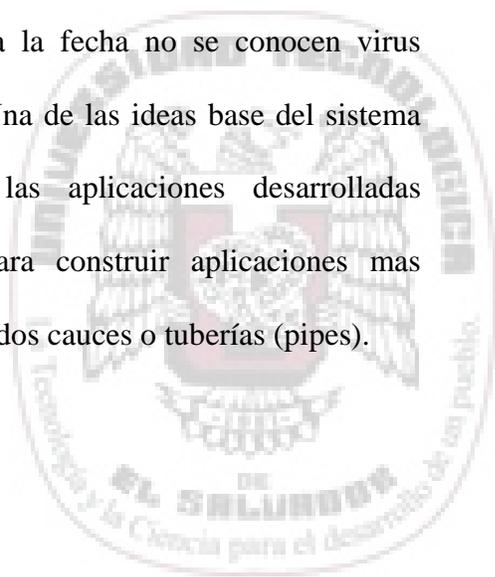


identificadas por el MS-DOS a través de una letra seguida de dos puntos. En MS-DOS existe la posibilidad de subdividir el disco en subdirectorios que permiten un empleo más ágil de toda la información. Este sistema operativo está lejos de ser el ideal, ya que es un sistema monotarea, que presenta problemas de diseño, los cuales provocan que el comportamiento de la máquina sea poco fiable.

7.4.2 UNIX.

UNIX fue concebido en 1969 en los laboratorios Bell de AT&T por un grupo de desarrolladores quienes lo diseñaron para su propio uso. Fue escrito originalmente en lenguaje ensamblador, y a mediados de los 70 fue reprogramado en lenguaje C. Es un SO que desde sus inicios se caracterizó por ser multiusuario y multitarea, esto significa que muchos usuarios pueden utilizar un solo computador al mismo tiempo, y que es capaz de correr más de un programa a la vez.

El sistema de archivos de UNIX proporciona un método sencillo de organizar archivos y permite la protección de éstos. Hasta la fecha no se conocen virus informáticos que afecten este sistema operativo. Una de las ideas base del sistema UNIX ha sido la posibilidad de reutilizar las aplicaciones desarrolladas independientemente unas de otras y unir las para construir aplicaciones más complejas por medio de mecanismos simples llamados cauces o tuberías (pipes).



7.4.3 LINUX.

Linux fue creado originalmente por Linus Benedict Torvalds en la Universidad de Helsinki en Finlandia. Este ha sido desarrollado con la ayuda de muchos programadores a través de Internet. Linus originalmente inició el hacking del núcleo como su proyecto favorito, inspirado por su interés en MINIX, un pequeño sistema UNIX. El se propuso a crear lo que en sus propias palabras sería un mejor Minix que el Minix. El 5 de octubre de 1991, Linus anunció su primera versión "oficial" de linux, versión 0.02. Desde entonces, muchos programadoras han respondido a su llamado, y han ayudado a construir Linux como el sistema operativo completamente funcional que es hoy⁷.

Algunas características de Linux son:

- a) Precio.
- b) Estabilidad, no se traba a cada rato.
- c) Seguridad, es mucho más seguro que otros servidores.
- d) Compatibilidad, reconoce la mayoría de los otros sistemas operativos en una red.
- e) Velocidad, es mucho más veloz para realizar las tareas.
- f) Posee el apoyo de miles de programadores a nivel mundial.

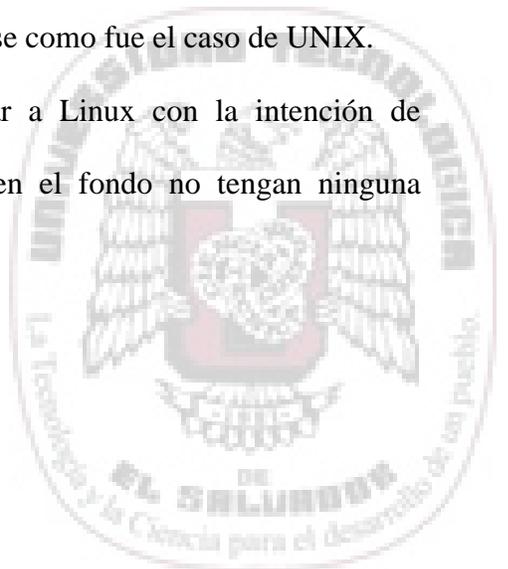
⁷ Maestros del Web LINUX, La buena opción en sistemas operativos, <http://www.maestrosdelweb.org/editorial/articulo.asp?linux>



- g) El paquete incluye el código fuente, lo que permite modificarlo de acuerdo a las necesidades del usuario.
- h) Ideal para la programación, ya que se puede programar en Linux para distintas plataformas, como para Windows.
- i) Un sistema de crecimiento rápido.
- j) Se puede usar en casi cualquier computadora.
- k) Multitareas real.
- l) Puede manejar múltiples procesadores. Incluso hasta 16 procesadores.
- m) Libre de virus, aun no se conoce ningún virus para Linux.
- n) Maneja discos duros de hasta 16 TeraBytes.
- o) Se consiguen parches con facilidad, además de ser gratuitos.
- p) Se posee el apoyo de millones de usuarios a nivel mundial.

Desventajas de Linux:

- a) Linux no cuenta con una empresa que lo respalde, por lo que no existe un verdadero soporte como el de otros sistemas operativos.
- b) Linux corre el riesgo de llegar a fragmentarse como fue el caso de UNIX.
- c) Algunas empresas pueden llegar a ayudar a Linux con la intención de mejorar sus relaciones públicas, aunque en el fondo no tengan ninguna intención de utilizarlo fielmente.



7.4.4 WINDOWS 95.

Es un entorno multitarea dotado de una interfaz gráfica de usuario, que a diferencia de las versiones anteriores (Windows 3.x) no necesita del MS-DOS para ser ejecutado, ya que es un sistema operativo.

Este SO está basado en menús desplegables, ventanas en pantalla y un dispositivo señalador llamado mouse. Una de las características principales de los sistemas Windows 95 es que los nombres de los archivos no están restringidos a ocho caracteres y tres de la extensión, pueden tener hasta 256 caracteres para tener una descripción completa del contenido del archivo. Además posee una tecnología desarrollada conjuntamente por los fabricantes de computadoras (Plug & Play), con la cual un usuario puede fácilmente instalar o conectar dispositivos permitiendo al sistema automáticamente alojar los recursos del hardware sin la intervención de usuario. Provee soporte para aplicaciones de 32 bits, multitarea con desalojo, soporte de red incorporado (TCP/IP, IPX, SLIP, PPP, y Windows Sockets). Incluye MS-DOS 7.0 como una aplicación.

7.4.5 WINDOWS 98.

Windows 98 es una nueva versión del sistema operativo Windows. Podría decirse que es una compilación de características. Muchas de estas características ya se



encontraban en Internet Explorer 4.0 (al ser instalado con la actualización de escritorio) y en Windows 95 OSR-2. Permite soporte para la FAT32 (al igual que Win95 OSR-2) y el Active Desktop (de Internet Explorer 4)⁸. Además, se caracteriza de Windows 95 porque posee soporte para dispositivos USB, asistente para la conexión a internet, posibilidad de utilizar varios monitores, herramientas para comunicación y funciones avanzadas para la navegación en internet, asistente para la solución de problemas, administrador de actualización, administrador de conflictos de versión, mejoras en la instalación y apagado más rápido.

7.4.6 WINDOWS NT.

El sistema operativo Windows NT fue desarrollado por Microsoft para superar los obstáculos impuestos por la vieja arquitectura de sus sistemas operativos MS-DOS y Windows 3.x. Su diseño modular y el estar escrito en lenguaje C le permiten ser un SO de mucha portabilidad. NT hereda las interfaces de usuario desarrollados para la familia Windows. Algunas de sus características principales son⁹:

- Direccionamiento de 32-bits.
- Soporte de memoria virtual.
- Soporte para multiprocesador.
- Arquitectura cliente/servidor.
- Compatibilidad con otros Sistemas Operativos.

⁸ Windows, detalle de la historia, <http://www.fortunecity.com/skyscraper/fatbit/607/winstory/storydetails.html>

⁹ Sistemas operativos, <http://www.tau.org.ar/base/lara.pue.udlap.mx/sistoper/index.html>



Independencia de plataformas.

Robustez: NT es un SO estable y robusto, que impide a las aplicaciones mal escritas estropear al resto del sistema.

Seguridad e integridad del sistema.

NT fue diseñado para satisfacer criterios de seguridad típicos de organismos oficiales y empresas cuyos datos y aplicaciones deben quedar a salvo de accesos no autorizados.

7.4.7 WINDOWS 2000.

El sistema operativo Windows 2000 posee las siguientes características:

Sistema operativo de 32 bits

Diseñado especialmente para empresas y usuarios avanzados

Soporte para dos procesadores.

Soporte para USB.

Soporte completo para Plug & Play.

Soporte para DirectX

Seguridad y estabilidad de un sistema NT

Protección de archivos Windows.

Mejorado para reducir los reinicios.



7.4.8 WINDOWS ME.

El sistema operativo Windows ME posee las siguientes características:

Diseñado especialmente para usuarios domésticos y pequeñas empresas.

Soporte de Windows auto update.

Soporte para System restore (permite recuperar archivos del sistema dañados que se almacenan en una zona protegida del disco duro)

Soporte mejorado para redes.

Soporte para DirectX

Soporte para USB.

Sistema operativo de 16/32 Bits (arquitectura basada en Windows 9x)

Soporte completo para Plug & Play.

7.4.9 WINDOWS XP.

El sistema operativo Windows XP posee las siguientes características:

Protección de archivos del Windows.

Arquitectura de modo de núcleo protegido.

Cambio rápido de usuario.

Agrupación en la barra de tareas. Los diferentes archivos que hay abiertos se agrupan según el tipo de aplicación.



Asistente para limpieza de escritorio. Comprueba periódicamente el escritorio y ofrece la oportunidad de mover los accesos directos que no se utilizan.

Modo de compatibilidad. Permite que una aplicación se ejecute en un modo de compatibilidad que ayude a imitar versiones anteriores de Windows.

Restauración de controladores de dispositivos.

Arquitectura de 32 bits.

Grabación de CD integrada.

Modelo de memoria completamente protegida. Permite que las aplicaciones errantes no provoquen daños al equipo.

7.5 Herramientas para el desarrollo de software.

Según James A. Senn, “los sistemas de información basados en computadora sirven para diversas finalidades que van desde el procesamiento de las transacciones de una empresa, hasta proveer de la información necesaria para decidir”¹⁰. Existen tres distintos enfoques para el desarrollo de sistemas de información basados en computadora:

Método del ciclo de vida (SDLC).



¹⁰ Senn, James A. Análisis y diseño de sistemas de información. Pág. 33

Este método consta de las siguientes actividades: investigación preliminar, determinación de requerimientos, diseño del sistema, desarrollo del software, prueba de los sistemas, implantación y evaluación.

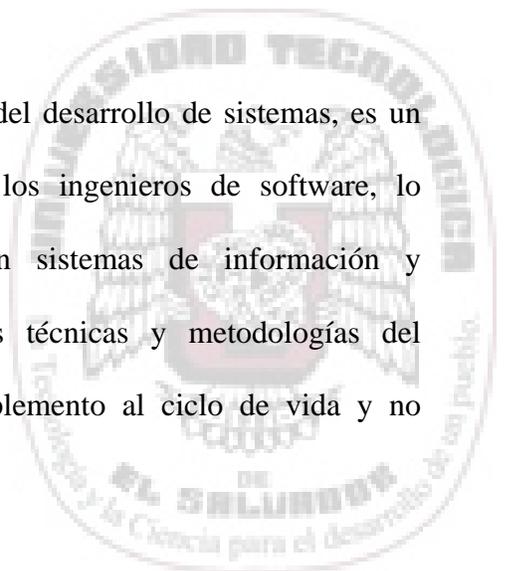
Método del desarrollo del análisis estructurado (SADT).

Este método tiene como finalidad superar la dificultad de comprender de manera completa sistemas grandes y complejos, y para conseguir este fin, el método divide el sistema en componentes, y construye un modelo del sistema. El método incorpora elementos tanto de análisis como de diseño.

Método del prototipo de sistemas.

Un prototipo es un sistema que funciona, desarrollado con la finalidad de probar ideas y suposiciones relacionadas con el nuevo sistema. Este método se fundamenta en que los usuarios pueden señalar las características que les agradaría o no tener, junto con los problemas que presenta un sistema que existe y funciona, con mayor facilidad que si se les pidiese que las describieran en forma teórica o por escrito.

Jeffrey L. Whitten, propone que el ciclo de vida del desarrollo de sistemas, es un proceso por el cual los analistas de sistemas, los ingenieros de software, los programadores y los usuarios finales elaboran sistemas de información y aplicaciones informáticas. Según Whitten: “Las técnicas y metodologías del desarrollo de sistemas pretenden servir de complemento al ciclo de vida y no



reemplazarlo”¹¹. Señala que entre las técnicas y metodologías que complementan al ciclo de vida se encuentran:

Las técnicas estructuradas

Son métodos formales de división de un problema de empresa en fragmentos y relaciones manejables, la ulterior reunión de estos fragmentos y relaciones útil para resolver el problema. Entre estas técnicas se encuentran:

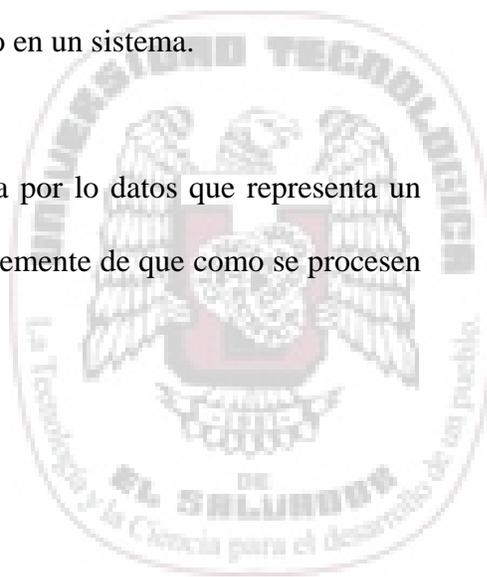
La programación estructurada: técnica orientada a procesos para el diseño y la escritura de programas con mayor claridad y consistencia.

El diseño estructurado: técnica orientada a procesos utilizada para fragmentar un programa grande en un conjunto jerarquizado de módulos y obtener un programa más fácil de implantar y mantener.

El análisis estructurado: técnica centrada en los procesos que se utilizan para realizar modelos de las necesidades de usuario en un sistema.

La modelización de datos: técnica orientada por lo datos que representa un sistema en función de sus datos. Independientemente de que como se procesen dichos datos para producir información.

¹¹ Whitten, Jeffrey L. Análisis y diseño de sistemas de información. Pág. 95 - 146



Técnica del desarrollo conjunto de aplicaciones (DCA)

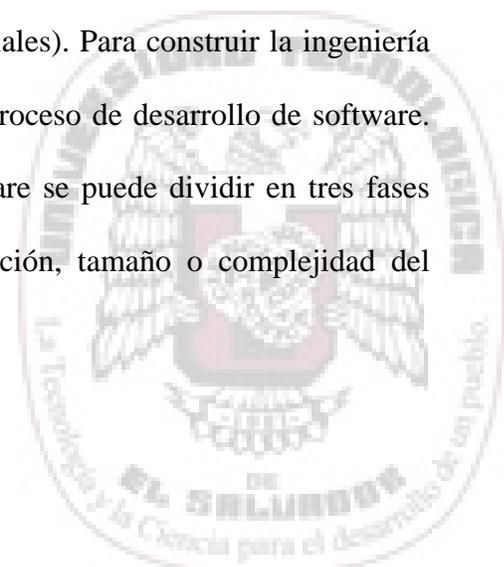
El desarrollo conjunto de aplicaciones (DCA) es una forma de trabajo altamente estructurada que lleva a los usuarios, los directivos y los especialistas en sistemas de información a definir y especificar conjuntamente las necesidades de los usuarios, las opciones técnicas y los diseños externos.

Técnicas de prototipos y desarrollo rápido (DRA)

El diseño de prototipos es una popular técnica de ingeniería utilizada para desarrollar modelos a escala de un producto o de sus componentes. Implica la creación de modelos interactivos de trabajo de un sistema o subsistema. El desarrollo rápido de aplicaciones (DRA) es una combinación de diversas técnicas estructuradas con técnicas de prototipos y de desarrollo conjunto de aplicaciones cuyo fin es acelerar el desarrollo de sistemas.

Según Roger S. Pressman, la ingeniería es el análisis, diseño, construcción, verificación y gestión de entidades técnicas (o sociales). Para construir la ingeniería del software adecuadamente, se debe definir un proceso de desarrollo de software. El trabajo que se asocia a la ingeniería de software se puede dividir en tres fases genéricas, con independencia del área de aplicación, tamaño o complejidad del proyecto¹².

¹² Pressman, Roger S. Ingeniería del software. Un enfoque práctico. Pág. 18



La fase de definición se centra sobre el qué. Es decir, se identifica la información que ha de ser procesada, que función y rendimiento se desea. Por tanto, han de identificarse los requisitos clave del sistema y del software.

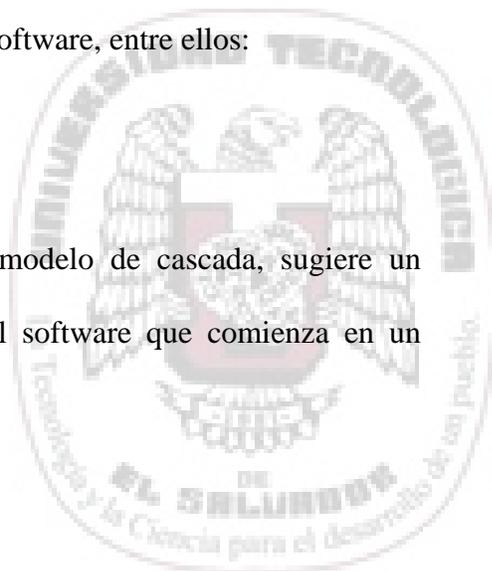
La fase de desarrollo se centra en el cómo. Es decir, durante el desarrollo se intenta definir cómo han de diseñarse las estructuras de datos, cómo ha de implementarse la función como arquitectura del software, cómo han de implementarse detalles procedimentales, cómo han de caracterizarse las interfaces,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

La fase de mantenimiento se centra en el cambio que va asociado a la corrección de errores, a las adaptaciones requeridas a medida que evoluciona el entorno del software, y a cambios debidos a las mejoras producidas por los requisitos cambiantes del cliente.

Pressman identifica varios modelos de proceso de software, entre ellos:

Modelo Lineal Secuencial

Llamado algunas veces ciclo de vida básico o modelo de cascada, sugiere un enfoque sistemático, secuencial del desarrollo del software que comienza en un



nivel de sistemas y progresa con el análisis, diseño, codificación, pruebas y mantenimiento.

Modelo de construcción de prototipos

Se utiliza cuando un cliente define un conjunto de objetivos generales para el software, pero no identifica los requisitos detallados de entrada, procesamiento o salida.

Modelo DRA

El Desarrollo Rápido de Aplicaciones (DRA) (Rapid Application Development, RAD) es un modelo de proceso del desarrollo del software lineal secuencial que enfatiza un ciclo de desarrollo extremadamente corto. Este modelo es una adaptación a alta velocidad del modelo lineal secuencial en el que se logra el desarrollo rápido utilizando un enfoque de construcción basado en componentes. Si se comprenden bien los requisitos y se limita el ámbito del proyecto, el proceso DRA permite al equipo de desarrollo crear un sistema completamente funcional dentro de períodos cortos de tiempo.

Modelos de procesos evolutivos

Estos modelos son iterativos, se caracterizan por la forma en que permiten a los ingenieros del software desarrollar versiones cada vez más completas del software.



7.5.1 Análisis de sistemas.

El análisis de sistemas es una tarea de ingeniería del software que cubre la distancia entre la definición del software a nivel sistema y el diseño del software. El análisis de sistemas es el estudio de una aplicación del sistema de información y de empresa actual, y la definición de las necesidades y las prioridades de usuario para conseguir una aplicación nueva o mejorada. Este se lleva a cabo con los siguientes objetivos en mente:

Identificar la necesidad del cliente.

Evaluar el concepto del sistema para establecer la viabilidad.

Realizar un análisis técnico y económico.

Asignar funciones al hardware, software y personal del sistema.

Establecer restricciones de presupuesto y planificación temporal.

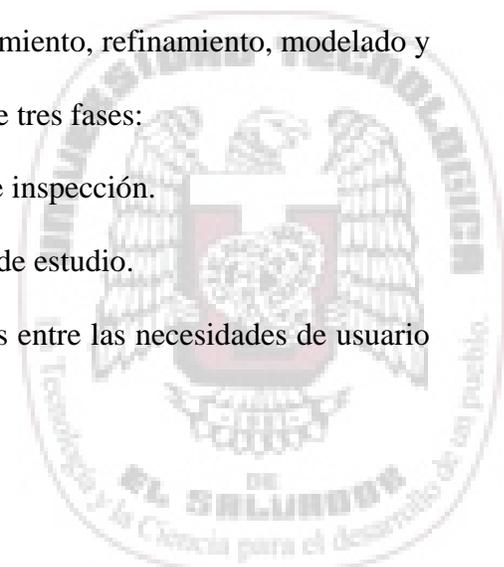
Crear una definición de sistema, que forme el fundamento de ingeniería subsiguiente.

La tarea del análisis, es un proceso de descubrimiento, refinamiento, modelado y especificación. El análisis de sistemas consta de tres fases:

Estudio de viabilidad del proyecto o fase de inspección.

Estudio y análisis del sistema actual o fase de estudio.

Definición y establecimiento de prioridades entre las necesidades de usuario o fase de definición.



Modelo de análisis

En un nivel técnico la ingeniería de software empieza con una serie de tareas de modelado que llevan a una especificación completa de los requisitos y a una representación del diseño general del software a construir. El modelo de análisis debe lograr tres objetivos primarios: describir lo que requiere el cliente, establecer una base para la creación de un diseño de software, y definir un conjunto de requisitos que se puedan validar una vez que ha construido el software¹³.

Para lograr estos objetivos, el modelado se apoya en las técnicas siguientes:

Diagramas entidad relación: es la piedra angular del modelo de datos. Proporciona al analista una notación concisa para analizar datos dentro del contexto de una aplicación de procesamiento de datos.

Diagramas de flujo de datos: es una técnica que representa el flujo de la información y las transformaciones que se aplican a los datos al moverse desde la entrada hasta la salida.

¹³ Pressman, Roger S. Ingeniería de Software. Un enfoque práctico. Pág. 202



Diagramas de transición de estados: es una especificación secuencial del comportamiento

7.5.2 Diseño de sistemas

El diseño de sistemas es la evaluación de las distintas soluciones alternativas y la especificación de una solución detallada de tipo informático. Mientras que el análisis concentra su interés en los aspectos lógicos, el diseño trata los aspectos físicos. Una vez obtenida las necesidades de una empresa con respecto a la elaboración de un sistema de información mejorado, se puede planear el modo en que funcionará dicho sistema.

El diseño podría definirse como *“el proceso de aplicar distintas técnicas y principios con el propósito de definir un dispositivo, un proceso o un sistema con suficiente detalle como para permitir su realización física”*¹⁴.

El objetivo del diseño es producir un modelo o representación de una entidad que se va a construir posteriormente.

¹⁴ Pressman, Roger S. Ingeniería de Software. Un enfoque práctico. Pág. 229



El diseño es un proceso multifase en el que se sintetizan representaciones de la estructura de datos, estructura del programa, características de la interfaz y detalles procedimentales desde los requisitos de la información.

Diseño de datos.

Es la primera de las cuatro actividades que se llevan a cabo durante la ingeniería del software. La actividad principal del diseño de datos es seleccionar representaciones lógicas de objetos de datos identificadas durante la fase de definición y especificación de requisitos.

Diseño arquitectónico.

El objetivo de este, es desarrollar una estructura de programa modular y representar las relaciones de control entre los módulos.

Diseño de la interfaz.

Se concentra en tres áreas importantes: el diseño de interfaces entre los módulos del software, el diseño de interfaces entre el software y otros productores y consumidores no humanos de información, y el diseño de la interfaz entre el hombre y la computadora.



Diseño procedimental.

Se realiza después de los diseños de datos, arquitectónico y de interfaz. Los fundamentos del diseño procedimental proponen el uso de un conjunto de construcciones lógicas con las que se puede formar cualquier programa, estas son la secuencia, la condición y la repetición. Estas tres construcciones son fundamentales para la programación estructurada, una importante técnica de diseño procedimental. El diagrama de flujo y las tablas de decisión son otras técnicas utilizadas para el diseño procedimental.

7.6 Marco teórico propuesto.

7.6.1 Propuesta sobre inventarios.

Por su grado de transformación el inventario es del tipo de producto terminado.

Debido al volumen de artículos que tiene el inventario se propone como método de costeo, el promedio ponderado.

Para obtener información determinante para el reabastecimiento de productos, se propone utilizar el método de rotación mensual, para determinar el consumo



mensual de un producto y luego establecer un inventario de seguridad, evitando así las faltas y excesos de existencia de dicho producto.

7.6.2 Propuesta de lenguajes de programación.

Cuadro comparativo de lenguajes de programación

Características	C++	Cobol	Java	Visual Basic
Interfaz utilizada	Caracter	Caracter	Gráfico	Gráfico
Orientado a objetos	Si	No	Si	No*
Uso mas frecuente:				
- Aplicaciones administrativas	No	Si	No	Si
- Sistemas operativos	Si	No	No	No
- Animación y efectos interactivos	Si	No	Si	Si

* Visual Basic no tiene la característica de herencia entre objetos.

Se propone Visual Basic, por ser una herramienta con interfaz gráfico, de rápido desarrollo, que tiene la propiedad de poder desarrollar aplicaciones interactuando con bases de datos como Access y SQL Server.

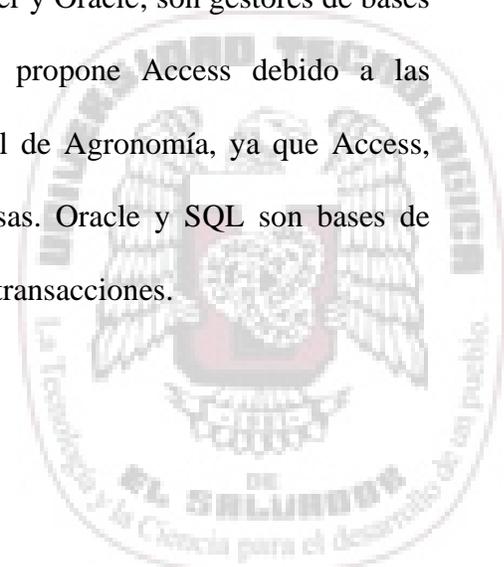


7.6.3 Propuesta de base de datos.

Cuadro comparativo de bases de datos

Características	Access	SQL Server	Oracle
Interfaz utilizada	Gráfico	Gráfico	Gráfico
RDBMS	Si	Si	Si
Interacción con plataformas Windows	Si	Si	Si
Uso mas frecuente: - Pequeñas empresas - Grandes corporaciones	Si No	No Si	No Si
Herramientas de desarrollo:	Visual Basic VBA	Visual Basic Visual C++	Oracle JDeveloper Suite
Servicios de Internet	Si	Si	Si

En cuanto a base de datos tanto Access, SQL Server y Oracle; son gestores de bases de datos relacionales de ambiente gráfico, se propone Access debido a las necesidades de aplicación de la Escuela Nacional de Agronomía, ya que Access, está orientado para sistemas de pequeñas empresas. Oracle y SQL son bases de datos robustas, diseñadas para altos volúmenes de transacciones.



7.6.4 Propuesta de sistemas operativos.

Cuadro comparativo de sistemas operativos

Características	MS-DOS	UNIX	LINUX	Windows 98	Windows XP
Interfaz utilizada	Caracter	Gráfico	Gráfico	Gráfico	Gráfico
Multiusuario	No	Si	Si	No	Si
Multitarea	No	Si	Si	Si	Si
Soporte de fabricante	Si	Si	No	Si	Si
Protección de archivos	No	Si	Si	No	Si
Soporte para red	No	Si	Si	Si	Si
Soporte para múltiples procesadores	No	Si	Si	No	Si

En cuanto a sistema operativo se propone el sistema operativo Windows 98, por tener soporte por parte del fabricante, ser un sistema operativo de fácil aprendizaje para los nuevos usuarios. Además, por ser un sistema operativo compatible con el lenguaje de programación Visual Basic y con la base de datos Access.



7.6.5 Propuesta de herramientas para el desarrollo de software.

Entre los distintos autores existe un enfoque distinto en cuanto a los métodos para el desarrollo de software, al analizar las teorías de los autores, la diferencia se encuentra en conceptualizar al ciclo de vida como una herramienta, como una técnica o como un método. Sin embargo, todos convergen en que el ciclo de vida, es una forma para desarrollar software, y entre las distintas técnicas, las de análisis y diseño estructurado, como una de las mas utilizadas para dar apoyo al ciclo de vida. Por lo que se propone utilizar ciclo de vida como un método para el desarrollo de software, apoyado por las técnicas estructuradas, para determinar los requerimientos del sistema a diseñar y desarrollar.

