



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

**ESTRUCTURACIÓN DE LOS LABORATORIOS Y
DOCUMENTACIÓN DE APOYO PARA LOS CURSOS:
SISTEMAS OPERATIVOS 1
SISTEMAS OPERATIVOS 2 Y
MANEJO E IMPLEMENTACIÓN DE ARCHIVOS**

Francisco Raúl Cruz Orellana
Asesorado por el Ing. Jorge Armin Mazariegos

Guatemala, junio de 2009.

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**ESTRUCTURACIÓN DE LOS LABORATORIOS Y
DOCUMENTACIÓN DE APOYO PARA LOS CURSOS:
SISTEMAS OPERATIVOS 1
SISTEMAS OPERATIVOS 2 Y
MANEJO E IMPLEMENTACIÓN DE ARCHIVOS**

TRABAJO DE GRADUACIÓN DE EPS

PRESENTADO A JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA
POR

FRANCISCO RAUL CRUZ ORELLANA
ASESORADO POR EL ING. JORGE ARMIN MAZARIEGOS
AL CONFERÍRSELE EL TÍTULO DE
INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, JUNIO DE 2009.

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERIA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Murphy Olympo Paiz Recinos
VOCAL I	Inga. Glenda Patricia García Soria
VOCAL II	Inga. Alba Maritza Guerrero de López
VOCAL III	Ing. Miguel Angel Dávila Calderón
VOCAL IV	Br. José Milton De León Bran
VOCAL V	Br. Isaac Sultan Mejía
SECRETARIA	Inga. Marcia Ivónne Véliz Vargas

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO/A	Ing. Murphy Olympo Paiz Recinos
EXAMINADOR/A	Inga. Floriza Felipa Ávila Pesquera de Medinilla
EXAMINADOR/A	Ing. Marlon Antonio Pérez Turk
EXAMINADOR/A	Inga. Sonia Yolanda Castañeda Ramírez
SECRETARIO/A	Inga. Marcia Ivónne Véliz Vargas

HONORABLE TRIBUNAL EXAMINADOR

Cumpliendo con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de Ejercicio Práctico Supervisado (EPS) titulado:

**ESTRUCTURACIÓN DE LOS LABORATORIOS Y
DOCUMENTACIÓN DE APOYO PARA LOS CURSOS:
SISTEMAS OPERATIVOS 1
SISTEMAS OPERATIVOS 2 Y
MANEJO E IMPLEMENTACIÓN DE ARCHIVOS,**

tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, con fecha febrero de 2008.

Francisco Raúl Cruz Orellana.



Guatemala 08 de Noviembre de 2008


Ingeniera
Norma Ileana Sarmiento
Directora Unidad EPS
Facultad de Ingeniería
Universidad de San Carlos de Guatemala

Un cordial saludo.

Por medio de la presente hago de su conocimiento que el estudiante: *Francisco Raúl Cruz Orellana* con carnet No. 200312484, ha realizado satisfactoriamente su proyecto de EPS. Dicho proyecto se realizó dentro de la Escuela de Ciencias y Sistemas, Facultad de Ingeniería, en un periodo DE SEIS MESES comprendido del 4 de Febrero de 2008 al 4 de Septiembre de 2008, en un horario de 4 horas diarias de lunes a viernes en su casa, con realizando entregas en formato digital, impresa y por correo electrónico, con reuniones para revisión y retroalimentación los días sábados en horario de 14:00 a 16:00 horas.

Dicho proyecto lleva el título **"ESTRUCTURACIÓN DE LOS LABORATORIOS Y DOCUMENTACIÓN DE APOYO PARA LOS CURSOS: SISTEMAS OPERATIVOS 1, SISTEMAS OPERATIVOS 2 Y MANEJO E IMPLEMENTACIÓN DE ARCHIVOS"**.

Atentamente,


08 - Nov 2008

Ing. Jorge Armin Mazariegos
Asesor de Proyecto de EPS
Escuela de Ciencias y Sistemas
USAC



Guatemala, 24 de abril del 2009.
REF.EPS.DOC.627.04.09.

Inga. Norma Ileana Sarmiento Zeceña de Serrano
Directora Unidad de EPS
Facultad de Ingeniería
Presente

Estimada Ingeniera Sarmiento Zeceña.

Por este medio atentamente le informo que como Supervisora de la Práctica del Ejercicio Profesional Supervisado, (E.P.S) del estudiante universitario de la Carrera de Ingeniería en Ciencias y Sistemas, **Francisco Raúl Cruz Orellana** Carné No. **200312484**, procedí a revisar el informe final, cuyo título es **“ESTRUCTURACIÓN DE LOS LABORATORIOS Y DOCUMENTACIÓN DE APOYO PARA LOS CURSOS: SISTEMAS OPERATIVOS 1, SISTEMAS OPERATIVOS 2 Y MANEJO E IMPLEMENTACIÓN DE ARCHIVOS”**.

En tal virtud, **LO DOY POR APROBADO**, solicitándole darle el trámite respectivo.

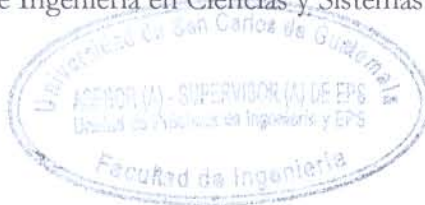
Sin otro particular, me es grato suscribirme.

Atentamente,

“Id y Enseñad a Todos”

Inga. Floriza Felipa Avila Pesquera de Medina
Supervisora de EPS
Área de Ingeniería en Ciencias y Sistemas

FFAPdM/RA





Guatemala, 24 de abril del 2009.
REF.EPS.D.239.04.09.

Ing. Marlon Antonio Pérez Turck
Director Escuela de Ingeniería Ciencias y Sistemas
Facultad de Ingeniería
Presente

Estimado Ingeniero Perez Turck.

Por este medio atentamente le envío el informe final correspondiente a la práctica del Ejercicio Profesional Supervisado, (E.P.S) titulado **“ESTRUCTURACIÓN DE LOS LABORATORIOS Y DOCUMENTACIÓN DE APOYO PARA LOS CURSOS: SISTEMAS OPERATIVOS 1, SISTEMAS OPERATIVOS 2 Y MANEJO E IMPLEMENTACIÓN DE ARCHIVOS”**, que fue desarrollado por el estudiante universitario **Francisco Raúl Cruz Orellana** Carné No. **200312484**, quien fue debidamente asesorado por el Ing. **Jorge Armin Mazariegos** y supervisado por la **Inga. Floriza Felipa Ávila Pesquera de Medinilla**

Por lo que habiendo cumplido con los objetivos y requisitos de ley del referido trabajo y existiendo la aprobación del mismo por parte del Asesor y de la Supervisora de EPS, en mi calidad de Directora apruebo su contenido solicitándole darle el trámite respectivo.

Sin otro particular, me es grato suscribirme.

Atentamente,

“Id y Enseñad a Todos”

Inga. Norma Ileana Sarmiento Zeceña de Serrano
Directora Unidad de EPS



NISZ/ra



Universidad San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 13 de Mayo de 2009

Ingeniero
Marlon Antonio Pérez Turk
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Pérez:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación del estudiante **FRANCISCO RAUL CRUZ ORELLANA**, titulado: **"ESTRUCTURACION DE LOS LABORATORIOS Y DOCUMENTACION DE LOS CURSOS: SISTEMAS OPERATIVOS 1, SISTEMAS OPERATIVOS 2 Y MANEJO E IMPLEMENTACION DE ARCHIVOS"**, y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

Al agradecer su atención a la presente, aprovecho la oportunidad para suscribirme,

Atentamente,


Ing. Carlos Alfredo Azurdia
Coordinador de Privados
y Revisión de Trabajos de Graduación



UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS Y SISTEMAS
TEL: 24767644

E
S
C
U
E
L
A

D
E

C
I
E
N
C
I
A
S
Y

S
I
S
T
E
M
A
S

El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del asesor con el visto bueno del revisor y del Licenciado en Letras de trabajo de graduación titulado **"ESTRUCTURACIÓN DE LOS LABORATORIOS Y DOCUMENTACIÓN DE APOYO PARA LOS CURSOS: SISTEMAS OPERATIVOS 1, SISTEMAS OPERATIVOS 2 Y MANEJO E IMPLEMENTACIÓN DE ARCHIVOS"**, presentado por el estudiante FRANCISCO RAÚL CRUZ ORELLANA, aprueba el presente trabajo y solicita la autorización del mismo.

"ID Y ENSEÑAD A TODOS"


Ing. Marlon Antonio Pérez Furk

Director, Escuela de Ingeniería Ciencias y Sistemas



Guatemala, 02 de junio 2009



El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al trabajo de graduación titulado: **ESTRUCTURACIÓN DE LOS LABORATORIOS Y DOCUMENTACIÓN DE APOYO PARA LOS CURSOS: SISTEMAS OPERATIVOS 1, SISTEMAS OPERATIVOS 2 Y MANEJO E IMPLEMENTACIÓN DE ARCHIVOS**, presentado por el estudiante universitario **Francisco Raúl Cruz Orellana**, autoriza la impresión del mismo.

IMPRÍMASE.

Ing. Murphy Olympo Paiz Recinos
DECANO



Guatemala, junio de 2009

/cc
c.c. archivo.

ACTO QUE DEDICO A:

DIOS:

Con un profundo agradecimiento por darme la oportunidad de alcanzar esta meta, por haber sido mi guía en todo momento, siendo la luz en el camino y una fuente inagotable del saber.

Mis padres:

Emma Yolanda Orellana de Cruz y Juan Francisco Cruz, mi más profunda gratitud por el apoyo y la comprensión a largo de mi carrera, a quienes dedico este triunfo.

AGRADECIMIENTOS A:

Mi asesor::	Ing. Armin Mazariegos, por el tiempo dedicado a la elaboración de este trabajo.
Universidad San Carlos de Guatemala, Facultad de Ingeniería:	Especialmente a la Escuela de Ciencias y Sistemas, por toda la formación que en ella recibí a través de todo el personal docente.
Mi hermana:	Maribel Cruz, por la ayuda y colaboración durante la carrera.
Mi tía:	Feliza Cruz, por su apoyo incondicional.

ÍNDICE GENERAL

ÍNDICE ILUSTRACIONES.....	V
GLOSARIO.....	IX
RESUMEN.....	XI
OBJETIVOS.....	XIII
INTRODUCCIÓN.....	XV

1. MARCO TEÓRICO

1.1 Principios de Pedagogía.....	01
1.2 Proceso de Enseñanza – Aprendizaje.....	03
1.3 Método de clases magistrales.....	05
1.4 Aprendizaje Práctico por medio de Laboratorios.....	07

2. DEFINICIÓN DE UNA GUÍA DEL INSTRUCTOR

2.1 Objetivos de una Guía de Instructor.....	09
2.2 Elementos de la Guía de Instructor.....	10
2.3 Proceso de construcción de la Guía del Instructor.....	11
2.4 Modo de uso de la Guía del Instructor.....	17

3. GUÍA DEL INSTRUCTOR DEL LABORATORIO DE SISTEMAS OPERATIVOS I

3.1	Datos generales.....	19
3.2	Distribución del curso.....	19
3.3	Evaluación.....	19
3.4	Detalle de sesiones.....	20
3.5	Distribución de sesiones.....	21
3.6	Detalle de tutoriales del curso.....	27
3.7	Detalle de prácticas del laboratorio.....	28
3.8	Detalle del proyecto.....	30
3.9	Detalle del caso de estudio.....	31
3.10	Detalle de conducción de exámenes.....	31
3.11	Ejemplo de examen.....	32
3.12	Bibliografía recomendada.....	33
3.13	Lab Handbook.....	33

4. GUÍA DEL INSTRUCTOR DEL LABORATORIO DE SISTEMAS OPERATIVOS II

4.1	Datos generales.....	39
4.2	Distribución del curso.....	39
4.3	Evaluación.....	40
4.4	Detalle de sesiones.....	40
4.5	Distribución de sesiones.....	41
4.6	Detalle de tutoriales del curso.....	46
4.7	Detalle de prácticas de laboratorio	47
4.8	Detalle del proyecto.....	49
4.9	Detalle de conducción de exámenes.....	49

4.10 Ejemplo de examen.....	50
4.11 Bibliografía recomendada.....	51
4.12 Lab Handbook.....	51

5. GUÍA DEL INSTRUCTOR DEL LABORATORIO DE MANEJO E IMPLEMENTACIÓN DE ARCHIVOS

5.1 Datos generales.....	57
5.2 Distribución del curso.....	58
5.3 Evaluación.....	58
5.4 Detalle de sesiones.....	59
5.5 Distribución de sesiones.....	59
5.6 Detalle de tutoriales del curso.....	61
5.7 Detalle de prácticas del laboratorio.....	62
5.8 Detalle de proyectos.....	63
5.9 Detalle de conducción de exámenes.....	64
5.10 Ejemplo de examen.....	65
5.11 Bibliografía recomendada.....	66
5.12 Lab Handbook.....	66

6. DOCUMENTACIÓN DE APOYO DEL CURSO DE SISTEMAS OPERATIVOS I

6.1 Marco general.....	71
6.2 Conceptos sistemas operativos.....	72
6.3 Procesos concurrentes asincrónicos.....	75
6.4 Semáforos.....	78
6.5 Mensajes.....	82

7. DOCUMENTACIÓN DE APOYO DEL CURSO DE SISTEMAS OPERATIVOS II

7.1 Interbloqueos y postergación indefinida.....	89
7.2 Administración de la memoria.....	92
7.3 Administración de dispositivos de E/S.....	96
7.4 Multiprocesamiento.....	100

8. DOCUMENTACIÓN DE APOYO DEL CURSO DE MANEJO E IMPLEMENTACIÓN DE ARCHIVOS

8.1 Arquitectura de una base de datos.....	109
8.2 Nivel Interno (Almacenamiento de una Base de Datos).....	113
8.3 Sistemas relacionales.....	117
8.4 Concurrencia.....	120

CONCLUSIONES.....	131
RECOMENDACIONES.....	133
BIBLIOGRAFÍA.....	135
ANEXOS.....	137

ÍNDICE ILUSTRACIONES

FIGURAS

1. Ejemplo de examen de Operativos 1.....	32
2. Ejemplo de examen de Operativos 2.....	50
3. Ejemplo de examen de Archivos.....	65
4. Páginas 1-4 “Introducción Sistemas Operativos”.....	70
5. Páginas 5-8 “Introducción Sistemas Operativos”.....	71
6. Páginas 9-12 “Introducción Sistemas Operativos”.....	71
7. Páginas 13-16 “Introducción Sistemas Operativos”.....	73
8. Páginas 17-20 “Introducción Sistemas Operativos”.....	74
9. Páginas 21-24 “Introducción Sistemas Operativos”.....	75
10. Páginas 25-28 “Introducción Sistemas Operativos”.....	76
11. Páginas 29-32 “Introducción Sistemas Operativos”.....	77
12. Páginas 33-36 “Introducción Sistemas Operativos”.....	78
13. Páginas 37-40 “Introducción Sistemas Operativos”.....	79
14. Páginas 41-44 “Introducción Sistemas Operativos”.....	80
15. Páginas 45-48 “Introducción Sistemas Operativos”.....	81
16. Páginas 49-52 “Introducción Sistemas Operativos”.....	82
17. Páginas 53-56 “Introducción Sistemas Operativos”.....	83
18. Páginas 57-60 “Introducción Sistemas Operativos”.....	84
19. Páginas 61-64 “Introducción Sistemas Operativos”.....	85
20. Páginas 65-68 “Introducción Sistemas Operativos”.....	86
21. Páginas 1-4 “Sistemas Operativos Avanzados”.....	88
22. Páginas 5-8 “Sistemas Operativos Avanzados”.....	89
23. Páginas 9-12 “Sistemas Operativos Avanzados”.....	90

24. Páginas 13-16 “Sistemas Operativos Avanzados”.....	91
25. Páginas 17-20 “Sistemas Operativos Avanzados”.....	92
26. Páginas 21-24 “Sistemas Operativos Avanzados”.....	93
27. Páginas 25-28 “Sistemas Operativos Avanzados”.....	94
28. Páginas 29-32 “Sistemas Operativos Avanzados”.....	95
29. Páginas 33-36 “Sistemas Operativos Avanzados”.....	96
30. Páginas 37-40 “Sistemas Operativos Avanzados”.....	97
31. Páginas 41-44 “Sistemas Operativos Avanzados”.....	98
32. Páginas 45-48 “Sistemas Operativos Avanzados”.....	99
33. Páginas 49-52 “Sistemas Operativos Avanzados”.....	100
34. Páginas 53-56 “Sistemas Operativos Avanzados”.....	101
35. Páginas 57-60 “Sistemas Operativos Avanzados”.....	102
36. Páginas 61-64 “Sistemas Operativos Avanzados”.....	103
37. Páginas 65-68 “Sistemas Operativos Avanzados”.....	104
38. Páginas 69 “Sistemas Operativos Avanzados”.....	105
39. Páginas 1-4 “Implementación de Archivos”.....	107
40. Páginas 5-8 “Implementación de Archivos”.....	108
41. Páginas 9-12 “Implementación de Archivos”.....	109
42. Páginas 13-16 “Implementación de Archivos”.....	110
43. Páginas 17-20 “Implementación de Archivos”.....	111
44. Páginas 21-24 “Implementación de Archivos”.....	112
45. Páginas 25-28 “Implementación de Archivos”.....	113
46. Páginas 29-32 “Implementación de Archivos”.....	114
47. Páginas 33-36 “Implementación de Archivos”.....	115
48. Páginas 37-40 “Implementación de Archivos”.....	116
49. Páginas 41-44 “Implementación de Archivos”.....	117
50. Páginas 45-48 “Implementación de Archivos”.....	118
51. Páginas 49-52 “Implementación de Archivos”.....	119
52. Páginas 53-56 “Implementación de Archivos”.....	120

53. Páginas 57-60 “Implementación de Archivos”	121
54. Páginas 61-64 “Implementación de Archivos”	122
55. Páginas 65-68 “Implementación de Archivos”	123
56. Páginas 69-72 “Implementación de Archivos”	124
57. Páginas 73-76 “Implementación de Archivos”	125
58. Páginas 77-80 “Implementación de Archivos”	126
59. Páginas 81-84 “Implementación de Archivos”	127

TABLAS

I. Distribución de horas y actividades.	12
II. Forma de evaluación	13
III. Detalle de sesiones	13
IV. Distribución de horas de operativos1	19
V. Forma de evaluación de operativos1	19
VI. Detalle de Sesiones de operativos1	20
VII. Distribución de sesiones de operativos1	21
VIII. Detalle de tutoriales de operativos1.....	27
IX: Prácticas de operativos1.....	28
X: Proyectos de operativos1.....	30
XI: Distribución de horas de operativos2.....	38
XII: Forma de Evaluación de operativos2	39
XIII: Detalle de Sesiones de operativos2.....	39
XIV: Distribución de sesiones de operativos2.....	40
XV: Detalle de tutoriales de operativos2.....	46
XVI. Prácticas de operativos2.....	47
XVII. Proyectos de operativos2.....	49

XVIII. Distribución de horas de archivos.....	56
XI. Forma de Evaluación de archivos	56
XX. Detalle de Sesiones de archivos	57
XXI. Distribución de sesiones de archivos.....	57
XXII. Detalle de tutoriales de archivos.....	61
XXIII. Prácticas de archivos.....	62
XXIV. Proyectos de archivos.....	63

GLOSARIO

Instructor Guideline	Guía del Instructor, se define como una programa detallado del laboratorio de un curso específico para uso interno de los auxiliares.
EPS	Ejercicio Profesional Supervisado
Módulo	Se define como un componente o unidad que esta dentro de un curso específico, que agrupa un contenido específico e independiente de las demás unidades.
Sesión	Desde un punto de vista educativo se define así a la agrupación de un tema a específico que será desarrollado en un espacio temporal de tiempo definido.

RESUMEN

Debido al avance acelerado que sufren las tecnologías de la información y la comunicación, determina que una carrera como la Ingeniería en Ciencias y Sistemas, se mantenga al ritmo de estos cambios, por lo que es necesario estar actualizando los contenidos y las estrategias de enseñanza para proveer a los egresados en el mercado laboral un nivel competitivo.

Por lo tanto, es necesario definir una estructura a cada uno de los laboratorios que pueda facilitar estos cambios, por lo que en el siguiente Ejercicio Práctico Supervisado se elaboró de una metodología de estructuración, en la cual se organizan los contenidos, sesiones, proyectos, a través del uso de la tecnología y así lograr mejorar la eficiencia en la enseñanza de los laboratorios de los cursos de la carrera.

Asimismo es importante proveer una documentación de apoyo relacionada a cada curso en la cual se detallen los temas más importantes explicados de una forma clara y sencilla, entendible para el estudiante, y con varias referencias de apoyo sobre artículos de interés de los eventos más recientes sobre dichos temas.

Dentro del proceso de definición y concepción del producto a entregar para el ejercicio práctico supervisado se contó con el apoyo del personal del Centro Tecnológico de la India con su sede en el campus universitario, siendo de vital importancia el apoyo obtenido por las personas de dicha institución en el refinamiento de un proceso definido para estructuración de laboratorios,

basado en las experiencias en capacitación obtenidas por la empresa transnacional TATA Consultancy Services.

El proceso de capacitación se desarrolló en el período comprendido entre los meses de enero – abril del año 2008. Posteriormente se elaboró de forma individual las guías temáticas y documentación relacionada con los contenidos de laboratorios y los contenidos de cursos cubiertos por el presente proyecto de graduación.

En este trabajo se aplicará la metodología mencionada a los cursos: Sistemas Operativos 1, Sistemas Operativos 2 y Manejo e Implementación de Archivos.

OBJETIVOS

GENERAL

Proveer un contenido estructurado, estandarizado y eficiente sobre los contenidos, prácticas y proyectos de los laboratorios de tres cursos que se imparten por la Escuela de Ciencias y Sistemas aplicando la metodología de diseño de un "Instructor Guideline" para que pueda ser implementado y enseñado por cualquier auxiliar o catedrático capacitado, además brindar un documento que contenga los temas que se imparten en cada curso, para darle facilidad y mejorar el proceso de enseñanza de los cursos.

ESPECÍFICOS

1. Elaborar un "Instructor Guideline", que detalle el contenido, proceso de enseñanza, evaluación y promoción de los laboratorios de los cursos:
 - Sistemas Operativos 1
 - Sistemas Operativos 2
 - Manejo e Implementación de Archivos
2. Elaborar un "Lab Handbook", en el cual se encuentre la definición de los requerimientos de los proyectos, prácticas y tareas propias de laboratorio y la forma de implementarlas.
3. Proporcionar un material audiovisual que sirva de soporte para el proceso de enseñanza de los laboratorios.

4. Desarrollar documentación de apoyo que provee a los estudiantes los conceptos más importantes del contenido del curso de una forma práctica y sencilla para fomentar el autoaprendizaje y mejorar el proceso de enseñanza, sobre los cursos:
 - Sistemas Operativos 1
 - Sistemas Operativos 2
 - Manejo e Implementación de Archivos

5. Documentar y proponer forma de estructuración de los exámenes de laboratorio.

6. Documentar una serie de artículos relacionados a cada una de las unidades del curso para mejorar el nivel de información relacionado a los cursos.

INTRODUCCIÓN

Mantener el nivel educativo de cada uno de los cursos que se imparten en la Escuela de Ciencias y Sistemas es una tarea muy importante para garantizar el conocimiento que tendrán los egresados de la carrera, por lo que es necesario mantenerlos actualizados de acuerdo al acelerado cambio que sufren las tecnologías de información y comunicación, por lo que es necesario estandarizarlos de una forma estructurada para poder impartir los temas más adecuados a los estudiantes.

Además, el control que se tiene sobre la entrega de proyectos, prácticas, etc. es muy pobre, pues no se tiene la debida información haciendo que se traslapen las entregas de varios cursos en ciertos días en particular.

Por lo tanto, la Escuela de Ciencias y Sistemas, dedicando sus esfuerzos a brindar una mejor enseñanza, ha creado el India-Guatemala IT Education Centre of Excellence, en el cual se imparten cursos certificados acerca de temas de interés sobre tecnología de la información y comunicación, y en un afán de mejorar los laboratorios existentes se pretende aplicar la metodología que se utiliza en estos cursos y así fortalecer los contenidos y desarrollar un pensum de alto nivel.

Se realizó un análisis del contenido actual de tres laboratorios, para lo cual fue necesario realizar una planeación educativa de acuerdo al curso impartido por el Señor Mrutunjaya Panda, el cual consistió en la enseñanza para la creación de un Instructor Guideline o Guía del Instructor, en el cual se detalla

el contenido de cada sesión que se impartirá en el curso, así como los proyectos, tareas, etc. que le serán para uso interno al auxiliar del curso.

Por lo tanto el presente documento presentará primero un capítulo introducción a la metodología y los elementos de la cual esta compuesta, posteriormente se detallarán el diseño de cada una de las guías desarrolladas para los cursos de Sistemas Operativos 1, Sistemas Operativos 2 y Manejo e Implementación de Archivos, asimismo la documentación respectiva de cada curso.

1. MARCO TEÓRICO

1.1. Principios de Pedagogía

Los principios de una buena práctica pedagógica pretenden incentivar al estudiante a alcanzar sus expectativas más altas, siempre respetando los diferentes estilos de aprendizaje, que se encuentran basados en la interacción entre profesor y estudiante. Esta interacción debe de fomentar la cooperación entre los participantes y la correcta administración del tiempo, desarrollando así el potencial de cada estudiante de una manera activa y basada en la retroalimentación.

De acuerdo a Chickerin y Gamson, los principios son siete.

a. Propiciar el contacto entre estudiantes y profesores.

Este principio se refiere a involucrar y motivar al estudiante en su propio aprendizaje. La comunicación con el profesor permite que el estudiante avance y resuelva los conflictos que puede encontrar en su proceso de aprendizaje. Además, conocer bien a los profesores aumenta el compromiso intelectual que posee el alumno y de una referencia sobre su propio futuro.

b. Fomentar la cooperación entre los estudiantes.

El trabajo en equipo es muy importante porque el aprendizaje de los estudiantes aumenta, siempre en un ambiente de cooperación y social. Además el compartir ideas y aprender a relacionarse con los demás facilita la comprensión de las ideas, conceptos y profundiza en la mejora del pensamiento.

c. Propiciar el aprendizaje activo.

El principio se refiere a que la enseñanza no solo debe ser escuchar lo que el profesor dice, y memorizar tareas pre-definidas, sino que se debe generar un ambiente de discusión en el que se puedan intercambiar experiencias y aprender de ellas.

d. Proporcionar retroalimentación a tiempo.

Este principio indica que es necesaria una retroalimentación adecuada y a tiempo para que se permita al estudiante poder aprender de las sugerencias y mejoras que se indiquen y así poder evaluar su propio progreso.

e. Enfatizar el uso apropiado del tiempo.

Este principio se refiere a hacer una planificación de las actividades a realizar previamente y así obtener resultados más rápidos y eficaces.

f. Propiciar altas expectativas al estudiante.

Este principio se refiere a definir lo que se espera de la enseñanza, las expectativas que se tienen al finalizar, pues esta se vuelven realidad cuando los profesores hacen un esfuerzo extra por establecer un ambiente de aprendizaje mas ameno y adecuado.

g. Respetar los diferentes estilos de aprendizaje

Este principio se refiere a que los miembros de una comunidad educativa siempre pueden aprender unos de otros, así poder expandir nuevos horizontes a través del enriqueciendo de las experiencias personales lo que incrementará el número de oportunidades educativas.

1.2. Proceso de Enseñanza – Aprendizaje

Educación

Se define como un proceso de socialización de los sujetos en una sociedad donde se desarrollan capacidades intelectuales, habilidades, destrezas y técnicas a los estudiantes.

Enseñanza

Se define como una actividad realizada conjuntamente mediante la interacción de tres elementos: un profesor o docente, uno o varios alumnos y el objeto de conocimiento.

Por medio del proceso de enseñanza, el docente transmite sus conocimientos al o a los alumnos a través de diversos medios, técnicas y herramientas de apoyo; siendo él, la fuente del conocimiento, y el alumno un simple receptor ilimitado del mismo.

Aprendizaje

Se define como el proceso de adquirir conocimientos, habilidades, actitudes o valores a través del estudio, la experiencia o la enseñanza. Se define técnicamente como un cambio relativamente estable en la conducta del sujeto como resultado de la experiencia, producido a través del establecimiento de asociaciones entre estímulos y respuestas mediante la práctica en un nivel elemental.

Es importante considerar los elementos que componen el proceso de enseñanza-aprendizaje, el cual se puede decir que le da valor y sentido al programa curricular, que esta compuesto por los sub-procesos, a través de

cursos, sesiones, interacción con el estudiante. Por lo tanto se puede enfocar en tres perspectivas: alumno, profesor y cuerpo de conocimiento, que es el resultado de un aprendizaje en donde el alumno ha desarrollado habilidades y actitudes a partir de ciertos contenidos específicos.

El proceso de enseñanza metafóricamente hablando se puede ver como un viaje por una autopista bien asfaltada que lo lleva a uno a un lugar determinado, pero considerando las nuevas tecnologías de informática y la comunicación, se visualizaría más como un viaje hacia el espacio, en el cual se tiene una misión bien definida pero existen muchas rutas para navegar este espacio que no posee fronteras.

Por lo tanto, es necesario tener una guía y una clara estrategia que evite que nos perdamos durante la exploración, allí es donde se observa la importancia que posee el profesor en este proceso.

Interacción profesor - conocimiento

Un profesor se puede definir como un diseñador de las secuencias del aprendizaje, así como de los medios y los materiales que faciliten el aprendizaje de manera significativa.

Interacción alumno - conocimiento

Esto se refiere a que el alumno es responsable de su aprendizaje.

Interacción profesor - alumno

Facilitación del aprendizaje a través del vínculo.

Acción didáctica

La enseñanza y el aprendizaje son dos fenómenos correlativos y relacionados, por lo que se denomina la relación didáctica. Se distinguen tres etapas en la acción didáctica:

1. **Planteamiento.** En esta etapa se formulan los objetivos educativos y los planes de trabajo adaptados a los objetivos previstos. La formulación de un plan implica la toma de decisiones anticipada y la reflexión con anterioridad a la puesta en práctica.

2. **Ejecución.** Posteriormente al planteamiento, el profesor pone en práctica los recursos y métodos didácticos, desarrollándose el proceso de enseñanza.

3. **Evaluación.** Es la etapa en la que se verifican los resultados obtenidos con la ejecución, materializándose en el proceso de evaluación.

1.3. Método de clases magistrales

Se le llama **clase magistral** a aquella en que el profesor habla sin interrupciones durante todo el período de tiempo, en una forma parecida a lo que sería una conferencia, y una **clase activa** aquella en que se motiva a la participación ya sea preguntando o dialogando con los alumnos, durante la exposición que efectúa.

En una buena clase magistral el alumno presta atención, toma notas, sigue detenidamente lo que expone el profesor, pero su actitud intelectual y física es casi enteramente pasiva, es decir solamente receptora y no creadora.

La clase magistral tiene en cambio cierta utilidad para el profesor, pues le permite a este ensayar las cosas, a la vez sobre si mismo, formulándolas, y sobre un auditorio, observando como este reacciona.

El método de ensayo que supone una clase magistral previamente investigada y escrita sobre un asunto novedoso que se desarrolla a lo largo de un curso, tiene sentido si el tema del curso no se repite y los resultados son visibles a través de publicaciones regulares del profesor.

La clase magistral se puede hacer más amena e interesante, y más sencilla de retener, con varios recursos:

- **La voz**

Es indispensable no sólo contar con un volumen adecuado de voz o un micrófono, sino que también se requiere modularla para ofrecer diferentes registros y no una nota que torne difícil distinguir un pensamiento de otro.

- **El cuerpo**

Un adecuado manejo del cuerpo es esencial. Dependiendo del auditorio, y cuidando de no hacer un papel algo exagerado, es más fácil prestar atención a la exposición de alguien que habla de pie, o incluso moviéndose, que alguien que lo haga sentado.

- **Las Imágenes visuales**

El uso del pizarrón (o diapositivas, películas, etc.) permite al que escucha variar su nivel de atención, que lógicamente aumenta ante cada elemento nuevo que se le plantea y le facilita la comprensión y retención del tema expuesto.

- **Invitados**

Traer varios profesionales invitados para que tomen a su cargo aspectos parciales de la exposición aproxima bastante la clase a una mesa redonda y aumenta también la atención del auditorio.

- **El plan de exposición**

Distribuir al comienzo el plan de la exposición. Esto permite al alumno saber cuál es el esquema total, a qué se dirige, ubicarse en todo momento respecto a qué parte del temario se está tratando, etc.

- **El texto de la clase**

Si se agrega la entrega de la versión escrita de la clase la atención del alumno se puede multiplicar, con beneficio para su aprendizaje, entre visualizar, escuchar, leer lo que escucha y marcar o hacer anotaciones al texto de la clase magistral.

- **Resumen final**

Este ayuda a fijar las conclusiones de la exposición.

1.4. Aprendizaje práctico por medio de laboratorios

Se basa en el paradigma "**learning by doing**", que se caracteriza por su alta interacción con el estudiante, pero posee un bajo nivel de popularidad debido a factores tecnológicos y económicos; su equivalente en el sistema educativo convencional los constituyen los laboratorios. En este tipo de sistemas, el estudiante pone en práctica los conocimientos teóricos que ha adquirido, mediante la utilización de herramientas de aprendizaje activo,

siguiendo este paradigma. Una de las principales bondades de este paradigma es el valor pedagógico que éste ofrece.

Este cambio implica el abandono de la idea básica que ve al profesor como centro de la transmisión de conocimientos, para asumir una nueva en la cual el alumno se transforma en socio responsable del aprendizaje y centro del proceso, pasando a ser el profesor un facilitador del acceso al conocimiento.

2. DEFINICIÓN DE UNA GUÍA DEL INSTRUCTOR

Se define como un documento que tiene por objeto planificar, estandarizar y optimizar los procesos de enseñanza-aprendizaje en función de un detalle específico de objetivos y actividades para cada sesión de estudio.

El proceso enseñanza-aprendizaje requiere de una programación más concreta la cual consiste en la preparación del trabajo que se va a desarrollar en cada sesión en particular. Tanto las clases teóricas como las prácticas requieren de una planificación específica que determine las distintas etapas que se van a desarrollar, así como el orden y la coordinación de las actividades a realizar.

La planificación se concreta en un documento escrito, lo cual implica la elaboración del programa de la asignatura. Un programa académico se puede definir como el conjunto de especificaciones y estrategias que realiza el profesor o grupo de profesores de cara al aprendizaje y mejora de actitudes de los alumnos en el ámbito específico de una asignatura.

Siendo un esquema y resumen de los temas asignados para un curso específico, el *Instructor Guideline* es exclusivo para instructores o auxiliares de cátedra o catedráticos.

El proceso de planificación precisa la realización de cuidadoso análisis y la toma en consideración de unos principios para su aplicación eficaz.

La planificación del proceso de enseñanza-aprendizaje debe poseer las siguientes características:

- **Flexible.** Todo plan deberá poder adaptarse a las circunstancias y prever diferentes alternativas.
- **Realista.** Se debe adecuar a las restricciones materiales, temporales, capacidades de los estudiantes y a las condiciones concretas en las que se desarrollará la enseñanza.
- **Preciso.** El plan debe ser detallado, incluyendo indicaciones exactas sobre el modo de proceder. Las líneas generales de actuación y los objetivos generales deben ser precisados en una secuencia de acciones concretas.

2.1. Objetivos de un instructor Guideline

Los objetivos principales de un instructor Guideline son planificar, estandarizar y optimizar el proceso de enseñanza aprendizaje.

El instructor Guideline sirve para instructores o auxiliares de cátedra o catedráticos, con el fin de evitar la confusión entre las políticas que se tomaran en relación a un curso en específico, estableciendo expectativas claras y detallando el material que se va aprender.

2.2. Elementos de un instructor Guideline

Un *instructor Guideline* está compuesto para varios elementos, entre los más importantes están:

- Información general del curso.
- Distribución de horas y actividades.

- Forma de evaluación.
- Detalle de sesiones.
- Distribución de sesiones.
- Detalle de tareas asignadas
- Detalle de tutoriales.
- Detalle de exámenes.
- Ejemplo de exámenes.
- Bibliografía.

Adicionalmente se cuenta con un documento adicional que es el llamado “**Lab Handbook**”, este es un documento de apoyo para la definición de tareas, prácticas y proyectos, en el cual se definen ciertos parámetros, lenguajes de programación, técnicas o temas específicos a incluir en el enunciado que se desarrolle.

2.3. Proceso de construcción de la guía del instructor

Información general del curso

Este elemento trata de describir la idea general e introductoria del curso, esta debe contener:

- Nombre del curso, según pensum, puede incluir el código del curso.
- Prerrequisitos, todo curso anterior que es obligatorio haber ganado.
- Objetivo, es el fin determinado que el curso busca.
- Módulos, partes del curso con temática diferente.

Distribución de horas y actividades.

Se debe definir la distribución del total de horas que dura el curso dentro de cada módulo que se ha designado, se debe especificar entre teoría y laboratorio, además debe describir qué actividades se realizarán por módulo.

Entre esas actividades podemos mencionar: tareas, exámenes, proyectos y tutoriales.

La siguiente tabla muestra un ejemplo de la distribución:

Tabla I: Distribución de horas y actividades.

Módulos	Teoría	Pract.	TOTAL	Tareas	Exam.	Tutorial	Proy.
Mod. 1	3 HRS	3 HRS	6 HRS	3	1	1	
Mod. 2	2 HRS	2 HRS	4 HRS	2	1	1	1
TOTAL	5 HRS	5 HRS	10HRS	5	2	2	1

Forma de evaluación.

Es la forma de asignarle valor a cada actividad que se realizará para llegar a un 100% de la nota total del laboratorio, como se muestra en la siguiente tabla de ejemplo:

Tabla II. Forma de evaluación

	NUMERO	PORCENTAJE	TOTAL
TUTORIAL	3	2.5%	7.50 %
CASO DE ESTUDIO	1	2.5%	2.50 %
TAREAS	4	10.0%	40.00%
PROYECTO	1	30.0%	30.00%
EXAMEN	1	20.0%	20.00%
		TOTAL	100.00%

Detalle de sesiones.

Se define el número de sesiones que constará cada módulo del curso, así como su duración, definiendo si existirá alguna tarea en cada sesión. Por ejemplo, se puede definir el detalle de sesiones de la siguiente manera:

Tabla III. Detalle de sesiones

NO.	TEORIA	NO.	PRACTICA	TAREA
1	MÓDULO 1 – SESIÓN 1	2	MÓDULO 1 – SESIÓN 1	
3	MÓDULO 1 – SESIÓN 2	4	MÓDULO 1 – SESIÓN 2	TAREA 1
5	MÓDULO 1 – SESIÓN 3	6	MÓDULO 1 – SESIÓN 3	TAREA 2
7	MÓDULO 2 – SESIÓN 1	8	MÓDULO 2 – SESIÓN 1	

Distribución de sesiones.

Para el desarrollo de una sesión educativa se tiene que tener en cuenta los siguientes puntos:

- Conocimientos previos de los alumnos
- Estimado del avance de la sesión
- Nivel de desarrollo intelectual y madurez del alumno
- Entorno en el que se va a desarrollar la labor docente
- Duración disponible

Debemos realizar la distribución de sesiones de tal forma que se divida por módulo y por número de sesión, en el documento debemos identificar el nombre del curso, el módulo que pertenece las sesiones siguientes, el número de sesión, el objetivo, la distribución de temas y por último el objetivo del laboratorio, si y solo si exista un laboratorio por clase, de lo contrario se debe hacer una distribución adicional solo para sesiones de laboratorio.

Detalle de tareas asignadas

Se debe detallar cada elemento de la tarea, desde el tipo de asignación, que criterios de evaluación se utilizará, que documentación contendrá, a qué módulo pertenece, el número de tarea, el objetivo de la tarea, en qué sesión será asignada, en qué sesión será entregada y si será calificación personal.

Detalle de tutoriales.

Se debe detallar cada elemento del tutorial, cuál es el objetivo del tutorial, qué criterios de evaluación se utilizará, que documentación contendrá, a qué módulo pertenece, el número de tarea, el objetivo de la tarea, en qué sesión será asignada, en qué sesión será entregada y si será calificación personal.

Detalle de casos de estudio

Se debe detallar cada elemento del caso de estudio, cuál es el objetivo del caso de estudio, qué criterios de evaluación se utilizará, qué documentación contendrá, a qué módulo pertenece, el numero de caso de estudio, el objetivo del caso de estudio, en qué sesión será asignada, en qué sesión será entregada y si será calificación personal.

Detalle de exámenes.

Se debe definir la forma de conducción de un examen, siempre mencionando el curso, el módulo a que pertenece, el total de número de exámenes y el detalle de cada examen.

El detalle de cada examen debe mencionar el número de sesiones que se deben impartir antes del examen, la duración del examen y la forma de distribución entre subjetiva y objetiva.

Ejemplo de exámenes.

Para que el instructor tenga una idea de la forma en que se debe evaluar y más o menos los criterios a evaluar, así como la distribución del mismo, es importante adjuntar ciertos exámenes tipo, los cuales deben ayudar al instructor a realizar el suyo mismo o modificar el ejemplo.

Bibliografía.

Es el conjunto de material didáctico que servirá para aumentar el nivel intelectual del tema, así como definir la referencia de donde se extrajo los temas a impartir por sesión.

Lab Handbook

Este es un elemento adicional en el que se detallan los requerimientos necesarios para la realización de los enunciados de tareas, prácticas, tutoriales y proyectos.

Está dividido en las secciones:

Sección 1 – Ejercicios de Laboratorio

Los ejercicios de laboratorio, sirven para evaluar algún subtema de la sesión da, que necesite algún tipo de reforzamiento práctico. Estas se realizan durante la sesión.

Sección 2 – Lab Assignments

Los Lab Assignments son prácticas de programación que los estudiantes desarrollaran para fortalecer sus conocimientos sobre un tema específico.

Sección 3 – Tutoriales

Los tutoriales son tareas cortas de investigación que sirven de complemento para un tema teórico.

Sección 4 – Proyectos

Los proyectos sirven para evaluar el conocimiento práctico adquirido a lo largo de todo el módulo integrando cada uno de los conceptos aprendidos.

2.4. Forma de uso de la guía del instructor

La guía del instructor está destinada a ser utilizada por los auxiliares de los laboratorios de cada uno de los cursos, en el cual podrán encontrar la información sobre las sesiones que deben de realizar y otras actividades relacionadas como proyectos, prácticas, tutoriales.

Adicionalmente se provee del Lab Handbook que servirá como una herramienta de guía para las prácticas y proyectos, donde indicará las especificaciones principales y mínimas que estos deben de llevar.

3. GUÍA DEL INSTRUCTOR DE SISTEMAS OPERATIVOS 1

A continuación se presenta el desarrollo de la Guía del curso de Sistemas Operativos 1, de acuerdo al formato sugerido en el curso de “Estructuración de laboratorios”, impartido por el señor. Mrutunjaya Panda.

3.1. Datos generales

NOMBRE DE CURSO:	Sistemas Operativos I (281)
PRE- REQUISITOS:	Organización de Lenguajes y Compiladores 2 (781) Arquitectura de Computadores y Ensambladores 1 (778)
POST – REQUISITOS:	Sistemas Operativos 2 (285)
OBJETIVO:	Este curso busca que el estudiante comprenda de forma práctica y sencilla, todos los aspectos a evaluar sobre los sistemas operativos, para que pueda distinguir las ventajas y desventajas de cada uno de ellos, tomando como punto de partida la administración de recursos. Mostrando la evolución de los sistemas operativos.

3.2. Distribución del curso

Tabla IV: Distribución curso Sistemas Operativos 1

MODULO	TEORÍA	PRÁCTICA	TOTAL	EXAMEN	LAB ASSIGNMENTS	PROYECTOS
Introducción a los Sistemas Operativos usando Linux	28 horas	12 horas	40 horas	1 ½ horas	4	1
			40 horas			

3.3. Evaluación

Ponderación de tareas, prácticas y proyectos.

Tabla V: Evaluación curso Sistemas Operativos 1

ACTIVIDAD	NUMERO	PORCENTAJE	TOTAL
TUTORIAL	3	2.5 %	7.50 %
CASO DE ESTUDIO	1	2.5%	2.50%
LAB ASSIGNMENTS	4	10.0 %	40.00 %
PROYECTO	1	30.0 %	30.00 %
EXAMEN	1	20.0 %	20.00 %
TOTAL			100.00 %

3.4. Detalle de sesiones

Tabla VI: Sesiones curso Sistemas Operativos 1

No	TEORIA	No	PRACTICA	TUTORIAL / PROYECTO / LAB ASSIGNMENT
1	Introducción a los Sistemas Operativos usando Linux - 1			Tutorial – 1
2	Introducción a los Sistemas Operativos usando Linux - 2			
3	Introducción a los Sistemas Operativos usando Linux - 3			
4	Introducción a los Sistemas Operativos usando Linux - 4	5	Introducción a los Sistemas Operativos usando Linux – 4	LAB Assignment – 1
6	Introducción a los Sistemas Operativos usando Linux – 5			
7	Introducción a los Sistemas Operativos usando Linux – 6			Caso de Estudio – 1
8	Introducción a los Sistemas Operativos usando Linux – 7			
9	Introducción a los Sistemas Operativos usando Linux – 8			Tutorial – 2
10	Introducción a los Sistemas Operativos usando Linux – 9	11	Introducción a los Sistemas Operativos usando Linux – 9	LAB Assignment – 2
12	Introducción a los Sistemas Operativos usando Linux – 10			Tutorial – 3
13	Introducción a los Sistemas Operativos usando Linux – 11	14	Introducción a los Sistemas Operativos usando Linux – 11	LAB Assignment – 3
15	Introducción a los Sistemas Operativos usando Linux – 12	16	Introducción a los Sistemas Operativos usando Linux – 12	LAB Assignment – 4
17	Introducción a los Sistemas Operativos usando Linux – 13	18	Introducción a los Sistemas Operativos usando Linux – 13	Proyecto – 1
19	Introducción a los Sistemas Operativos usando Linux – 14	20	Introducción a los Sistemas Operativos usando Linux – 14	

3.5. Distribución de sesiones

Tabla VII: Sesiones curso Sistemas Operativos 1

SESIÓN NO. 1:	
OBJETIVO:	Definir los conceptos básicos de Sistemas Operativos y mostrar su evolución a través del tiempo.
CONTENIDO:	<ol style="list-style-type: none"> 1. Definición de Sistemas Operativos <ol style="list-style-type: none"> 1.1. Funciones 1.2. Características 2. Evolución de los Sistemas Operativos <ol style="list-style-type: none"> 2.1. Primera Etapa: Procesamiento en Serie 2.2. Segunda Etapa: Procesamiento por Lotes 2.3. Tercera Etapa: Multiprogramación y Tiempo Compartido 2.4. Cuarta Etapa: Redes de Ordenadores
SESIÓN NO. 2:	
OBJETIVO:	Identificar la estructura básica de un sistema operativo y su respectiva clasificación.
CONTENIDO:	<ol style="list-style-type: none"> 1. Estructura de los Sistemas Operativos <ol style="list-style-type: none"> 1.1. Sistema Monolítico 1.2. Modelo Cliente – Servidor 1.3. Máquina Virtual 1.4. Modelo Orientado a Objetos 2. Clasificación de los Sistemas Operativos <ol style="list-style-type: none"> 2.1. Respecto al modo de Trabajo del Usuario 2.2. Respecto al número de usuarios 2.3. Respecto al propósito 2.4. Al existir varios procesadores
SESIÓN NO. 3:	
OBJETIVO:	Definir que es un proceso dentro del Sistema Operativo
CONTENIDO:	<ol style="list-style-type: none"> 1. Definición y control de procesos <ol style="list-style-type: none"> 1.1. ¿Qué es un proceso? 1.2. Estados de un proceso 1.3. Bloque de control de Procesos (PCB)

	<ul style="list-style-type: none"> 1.3.1. Operaciones sobre procesos 1.3.2. Diagrama de estados 1.4. Control de un Proceso <ul style="list-style-type: none"> 1.4.1. Modos de Ejecución 1.4.2. Cambio de Proceso 1.4.3. Cambio de Contexto
SESIÓN NO. 4:	
OBJETIVO:	Definir que es la planificación de procesos e identificar los algoritmos para realizarla.
CONTENIDO:	<ul style="list-style-type: none"> 1. Planificación de procesos <ul style="list-style-type: none"> 1.1. Niveles de planificación 1.2. Objetivos y criterios de planificación 1.3. Planificación apropiativa y no apropiativa 1.4. Planificación con prioridades 1.5. Algoritmos de planificación
SESIÓN NO. 5:	
OBJETIVO:	Practicar los conceptos aprendidos en las sesión 4.
CONTENIDO:	Realización del LAB Assignment 1
SESIÓN NO. 6:	
OBJETIVO:	Definir los conceptos básicos de Linux y su funcionalidad.
CONTENIDO:	<ul style="list-style-type: none"> 1. Introducción Linux 2. Estructura general del Sistema 3. Modo Núcleo - modo usuario <ul style="list-style-type: none"> 3.1 Principios 3.2 Llamadas al Sistema 4. Desarrollo bajo Linux <ul style="list-style-type: none"> 4.1 Herramientas 4.2 Ejecutables 4.3 Bibliotecas

SESIÓN NO. 7:	
OBJETIVO:	Definir los conceptos básicos del núcleo de Linux, identificar su estructura, compilación e instalación.
CONTENIDO:	<ul style="list-style-type: none"> 1. El núcleo de Linux <ul style="list-style-type: none"> 1.1 Definición 1.2 Estructura 1.3 Características 1.4 Compilación e Instalación de un nuevo núcleo 1.5 Proceso de Arranque del núcleo
SESIÓN NO. 8:	
OBJETIVO:	Definir los procesos en Linux, identificar sus operaciones básicas y determinar cómo se implementan a nivel de Sistema Operativo.
CONTENIDO:	<ul style="list-style-type: none"> 1. Procesos en Linux <ul style="list-style-type: none"> 1.1 Conceptos Básicos <ul style="list-style-type: none"> 1.1.1 Definición 1.1.2 Estados 1.1.3 Atributos 1.1.4 Identificadores 1.1.5 Grupos 1.1.6 Sesiones 1.2 Operaciones <ul style="list-style-type: none"> 1.2.1 Creación 1.2.2 Finalización 1.2.3 Lectura de atributos 1.2.4 Ejecución 1.2.5 Coordinación 1.2.6 Personalización 1.3 Implementación <ul style="list-style-type: none"> 1.3.1 Tabla de procesos 1.3.2 Registros del procesador 1.3.3 Sincronización de procesos 1.3.5 Ámbitos de ejecución
SESIÓN NO. 9:	
OBJETIVO:	Definir el concepto de Hilos y como implementarlos en Linux.

CONTENIDO:	<ul style="list-style-type: none"> 1. Hilos <ul style="list-style-type: none"> 1.1 Definición 1.2 Hilos a nivel de núcleo y nivel de usuario 1.3 Modelos Multihilos 1.4 Implementación 1.5 Ejemplo
SESIÓN NO. 10:	
OBJETIVO:	Definir que es concurrencia a nivel de procesos y como solucionarla.
CONTENIDO:	<ul style="list-style-type: none"> 1. Concurrencia: Exclusión Mutua y Sincronización <ul style="list-style-type: none"> 1.1 Definición 1.2 Programación concurrente 1.3 Especificación 1.4 Características 1.5 Problemas 1.6 Región Crítica 1.7 Requisitos para la exclusión mutua 1.8 Soluciones por Hardware 1.9 Soluciones por Software <ul style="list-style-type: none"> 1.9.1 Algoritmo de Dekker 1.9.2 Algoritmo de Peterson
SESIÓN NO. 11:	
OBJETIVO:	Practicar los conceptos aprendidos en la sesiones 8,9 y 10.
CONTENIDO:	Realización del LAB Assignment 2
SESIÓN NO. 12:	
OBJETIVO:	Definir el concepto de Semáforos y como implementarlos en Linux.
CONTENIDO:	<ul style="list-style-type: none"> 1. Semáforos <ul style="list-style-type: none"> 1.1 Definición 1.2 Procedimientos 1.3 Tipos

	<ul style="list-style-type: none"> 1.4 Ejemplo 1.5 Exclusión mutua con semáforos 1.6 Estructuras Básicas de Linux 1.7 Representación interna 1.8 Implementación
SESIÓN NO. 13:	
OBJETIVO:	Identificar los problemas clásicos de concurrencia.
CONTENIDO:	<ul style="list-style-type: none"> 1. Problemas Clásicos de Concurrencia <ul style="list-style-type: none"> 1.1 Productor/Consumidor 1.2 Lector/Escritor 1.3 Filósofos Comensales 1.4 Barberos dormilones 1.5 Otros Algoritmos
SESIÓN NO. 14:	
OBJETIVO:	Practicar los conceptos aprendidos en las sesiones 12 y 13.
CONTENIDO:	Realización del LAB Assignment 3
SESIÓN NO. 15:	
OBJETIVO:	Definir el concepto de cola de mensajes y como implementarla en Linux.
CONTENIDO:	<ul style="list-style-type: none"> 1. Cola de Mensajes <ul style="list-style-type: none"> 1.1 Definición 1.2 Sincronización 1.3 Direccionamiento 1.4 Formato de mensajes 1.5 Implementación 1.6 Estructuras Básica 1.7 Creación 1.5 Envío 1.6 Recepción 1.7 Representación interna

SESIÓN NO. 16:	
OBJETIVO:	Practicar los conceptos adquiridos en las sesión 15.
CONTENIDO:	Realización del LAB Assignment 4
SESIÓN NO. 17:	
OBJETIVO:	Definir el concepto de Sockets y como implementarlos en Linux.
CONTENIDO:	<ul style="list-style-type: none"> 1. Sockets <ul style="list-style-type: none"> 1.1 Definición 1.2 Características 1.3 Propiedades 1.4 Tipos 1.3 Dominio 1.5 Operaciones <ul style="list-style-type: none"> 1.5.1 Creación 1.5.2 Nombramiento 1.5.3 Creación de cola de sockets 1.5.4 Aceptando conexiones 1.5.5 Solicitando conexiones 1.6 Ejemplo Cliente/Servidor
SESIÓN NO. 18:	
OBJETIVO:	Practicar los conceptos aprendidos a lo largo de todo el módulo.
CONTENIDO:	Realización del proyecto 1
SESIÓN NO. 19:	
OBJETIVO:	Esta sesión es por atrasos en las sesiones teóricas previas.
CONTENIDO:	

SESIÓN NO. 20:	
OBJETIVO:	Esta sesión es por atrasos en las sesiones prácticas previas.
CONTENIDO:	

3.6. Detalle de tutoriales

ASSIGNMENTS:

Criterio de Evaluación:

- Correcto - 70
- Conocimiento técnico - 20
- Documentación - 10

Documentación a entregar: SI

- Enunciado

**MÓDULO: Introducción a los Sistemas Operativos
 usando Linux**

Tabla VIII: Tutoriales curso Sistemas Operativos 1

TUTORIAL 1:	Evaluar los conceptos adquiridos en la sesión No. 1
OBJETIVO:	Comparar entre los diferentes Sistemas Operativos de acuerdo sus características principales.
Iniciada en sesión	1
Entregada en sesión	2
Viva / Demostración	NO
TUTORIAL 2:	Evaluar los conceptos adquiridos en la sesión No. 4
OBJETIVO:	Practicar los conocimientos adquiridos sobre hilos y las librerías para poder implementarlos

Iniciada en sesión	4
Entregada en sesión	5
Viva / Demostración	NO
TUTORIAL 3:	Evaluar los conceptos adquiridos en la sesión No. 6
OBJETIVO:	Investigar los distintos problemas clásicos de concurrencia.
Iniciada en sesión	6
Entregada en sesión	7
Viva / Demostración	NO

3.7. Detalle de lab assignments

PRÁCTICAS:

Criterio de Evaluación:

- **Funcionamiento** - **75**
- **Presentación** - **05**
- **Conocimiento técnico** - **10**
- **Documentación** - **10**

Documentación a entregar: SI

- **Enunciado**

MÓDULO: Introducción a los Sistemas Operativos usando Linux

Tabla IX: Prácticas curso Sistemas Operativos 1

LAB ASSIGNMENT 1:	Poner en práctica los conocimientos adquiridos sobre planificación de procesos.
OBJETIVO:	Implementar los algoritmos de planificación de procesos más representación para fortalecer el conocimiento de los mismos y comparar su rendimiento.

Iniciada en sesión	4
Entregada en sesión	5
Viva / Demostración	SI
LAB ASSIGNMENT 2:	Poner en práctica los conocimientos sobre el Bloque de control de procesos y los algoritmos de dekker.
OBJETIVO:	Simular el funcionamiento del bloque de control de procesos y los algoritmos de dekker por medio de la implementación de hilos en Linux.
Iniciada en sesión	5
Entregada en sesión	6
Viva / Demostración	SI
LAB ASSIGNMENT 3:	Poner en práctica los conocimientos sobre semáforos.
OBJETIVO:	Simular el funcionamiento de dos problemas clásicos de concurrencia por medio del uso de semáforos en Linux.
Iniciada en sesión	7
Entregada en sesión	8
Viva / Demostración	SI
LAB ASSIGNMENT 4:	Poner en práctica los conocimientos sobre las colas de mensajes.
OBJETIVO:	Simular el funcionamiento de dos problemas clásicos de concurrencia por medio del uso de colas de mensajes en Linux.
Iniciada en sesión	8
Entregada en sesión	9
Viva / Demostración	SI

3.8. Detalle del proyecto

PROYECTO:

Criterio de evaluación:

- **Funcionamiento** - 70
- **Conocimiento Técnico** - 10
- **Documentación** - 20

Documentación a entregar: SI

- **Enunciado**

**MÓDULO: Introducción a los Sistemas Operativos
 usando Linux**

Tabla X: Proyecto curso Sistemas Operativos 1

PROYECTO 1:	Poner en práctica los conocimientos del curso, así como su espíritu investigativo que le ayuda a obtener más conocimientos que le puedan servir en su vida de profesional.
OBJETIVO:	Emular el funcionamiento de un clúster, un conjunto de computadoras interconectadas capaces de compartir procesos para completar o procesar un proceso padre, aplicando arquitectura cliente servidor, en la cual la computadora servidor se encarga de distribuir los procesos a los nodos hijos para procesarlos y monitorearlos, haciendo la conexión a través de sockets en Linux.
Iniciada en sesión	9
Entregada en sesión	10
Viva / Demostración	SI

3.9. Detalle del caso de estudio

ASSIGNMENT: Realizar la instalación de una distribución de Linux stage 4, realizando una configuración y personalización del núcleo y su respectiva compilación.

OBJETIVO: Usar el conocimiento adquirido sobre el núcleo de Linux para poder instalarlo y configurarlo respectivamente.

TUTORIAL:

Iniciada en sesión 2

Entregada en sesión 4

Viva / Demostración NO

Criterio de Evaluación:

- | | | |
|-----------------|---|----|
| • Presentación | - | 10 |
| • Completo | - | 20 |
| • Correcto | - | 50 |
| • Documentación | - | 20 |

Documentación: SI

- Enunciado

3.10. Detalle del examen

EXAMEN:

Número total de exámenes 1

Examen realizado después de sesión 10

Temas para el examen Introducción a los Sistemas Operativos usando Linux

Duración 1 ½ hrs.

Criterio de Evaluación:

- **Objetivo** - **40%**
- **Subjetivo** - **60%**

3.11. Examen de ejemplo

Serie I (30 pts): Se debe de implementar la solución para el siguiente problema, utilizando semáforos de C, o C++ o QT, indicando las librerías y el código específico en el lenguaje seleccionado.

Se considera un puente levadizo de una única vía, en el que cruzan vehículos y barcos, con las siguientes características:

En cuanto al cruce por parte de vehículos:

- Pueden haber varios vehículos a la vez en el puente

En cuanto al cruce por parte de barco

- Los barcos tendrán prioridad para el cruce, debiendo poder hacer el cruce tan pronto como se haya vaciado de vehículos y elevado el puente.
- Solamente se admitirá el cruce de un único barco a la vez.
- En caso de presentarse cola de barcos, se desea evitar que el puente se eleve y descienda para cada uno de ellos

Serie II (30 pts): Desarrollo los siguientes temas.

- Escriba el código de la creación de un hilo y un ejemplo de su utilización en Java.
- Escriba el código de la creación de un socket y un ejemplo de su utilización en C++.
- Escriba la definición y ejemplo de utilización de un semáforo en C.
- Describa dos algoritmos de planificación y comparé su rendimiento.
- Realizar el diagrama de los niveles de planificación.

Serie III (40 pts): Escriba la respuesta correcta.

1. _____ es una secuencia de instrucciones escrita en un lenguaje dado.
2. _____ es una instancia de ejecución de un programa, que tiene recursos asignados.
3. _____ es un conjunto de programas que controlan directamente los recursos físicos del computador y proporcionan una interfaz entre el usuario y el hardware.
4. _____ comando de Linux para mostrar los procesos que se están ejecutando en el sistema.
5. _____ es el corazón de un sistema operativo.
6. _____ es una fase de instalación de un sistema operativo que posee solo un pequeño grupo de paquetes, destinados para una función específica.
7. _____ es la carpeta estándar donde se configura el grub de gentoo.
8. _____ es la librería para el manejo de sockets en QT.
9. _____ tipo de socket que provee un envío de datos de dos vías, confiable y sin duplicados.
10. _____ es el problema de la cuarta versión del algoritmo de dekker.

Figura 1: Examen Ejemplo Sistemas Operativos 1

3.12. Bibliografía sugerida

1. Abraham Silberschatz & James, **Operating Systems Concepts.**
2. Francisco Manuel Marquez Addison Wesley, **Unix. Programación Avanzada**
3. Lina García Cabrera, Francisco Martínez del Rio & Miguel A. Redondo Duque, **Sistemas Operativos**
4. Andrew S. Tanenbaum, **Sistemas operativos modernos** 2a edición.

3.13. Lab handbook

Lab Assignment 1

Simular gráficamente los algoritmos de planificación de procesos.

Especificaciones:

- Implementar 5 o más algoritmos
- Realizar la opción de carga manual de los procesos o utilizando los procesos que se ejecutan en el sistema.
- Simular el reloj del sistema, mostrado el tiempo que va transcurriendo en pantalla.
- Realizar documentación técnica y de usuario.
- Trabajarlo sobre sistema operativo LINUX.
- Implementarlo en Java o QT.

Lab Assignment 2

Simular gráficamente el bloque de control de procesos (PCB) del sistema, e implementar cada uno de las versiones del algoritmo de Dekker.

Especificaciones:

- Mostrar gráficamente los cambios estados y transiciones del PCB.
- Número de procesos a ejecutar, 15 procesos o más.
- Manejar Id del proceso, quantum y memoria a utilizar.
- Mostrar gráficamente el porcentaje de memoria utilizada.
- Implementar las 5 versiones del algoritmo de Dekker.
- Utilizar threads(hilos) para la implementación.
- Animar cada versión del algoritmo de Dekker en situaciones donde se visualice el problema que cada versión posee.
- Se debe de implementar una bitácora con el registro de todas las acciones realizadas por cada proceso.
- Realizar documentación técnica y de usuario.
- Trabajarlo sobre sistema operativo LINUX.
- Implementarlo en Java o QT.

Lab Assignment 3

Simular algoritmos de concurrencia utilizando semáforos para resolver la exclusión mutua.

Especificaciones:

- Implementar gráficamente dos algoritmos o más.
- Utilizar librerías del sistema de c o c++, para la utilización de los semáforos.
- Realizar documentación técnica y de usuario.
- Trabajarlo sobre sistema operativo LINUX.

- Implementarlo en Java o QT.

Lab Assignment 4

Simular algoritmos de concurrencia utilizando mensajes para resolver la exclusión mutua.

Especificaciones:

- Implementar gráficamente dos algoritmos o más.
- Utilizar librerías del sistema de c o c++, para la utilización de los mensajes.
- Realizar documentación técnica y de usuario.
- Trabajarlo sobre sistema operativo LINUX.
- Implementarlo en Java o QT.

Tutorial 1

Realizar una comparación entre los diferentes Sistemas Operativos.

Especificaciones:

- Investigar 5 Sistemas Operativos diferentes o más.
- Describir las características técnicas, graficas y un análisis comparativo de costos.
- Incluir referencias bibliográficas.

Tutorial 2

Investigar sobre las librerías que se utilizar para la implementación de los threads (hilos)

Especificaciones

- Investigar 4 lenguajes de programación diferentes o más.

- Describir características generales, métodos de creación, eliminación, etc.
- Realizar un ejemplo de su uso en cada lenguaje investigado.
- Incluir referencias bibliográficas.

Tutorial 3

Investigar problemas clásicos de concurrencia.

Especificaciones

- Investigar 3 problemas diferentes o más.
- Describir características generales, solución, forma de implementarlos, etc.
- Realizar un ejemplo de implementación en java o c++.
- Incluir referencias bibliográficas.

Proyecto No. 1

El proyecto consiste en realizar la comunicación entre diferentes computadores, a través de la simulación de un cluster, que sea capaz de compartir procesos.

Especificaciones:

- Se utilizará la arquitectura cliente/servidor
- Se debe desarrollar un módulo en el servidor que emule una base de datos donde se almacenaran todos los procesos ejecutados.
- Se debe desarrollar un módulo en servidor que distribuya los procesos y pueda verlos como si fueran propios del sistema operativo.
- Se debe desarrollar una aplicación cliente que permita la creación de los procesos, a través de una interfaz gráfica.
- Se debe implementar seguridad a la aplicación por medio de un modulo de usuarios, con diferentes roles y manejo de sesiones.

- Se debe de implementar una bitácora que guarde la información de cada acción realizada.
- Se debe hacer un instalador interactivo, para la aplicación cliente y para la aplicación del servidor.
- Realizar documentación técnica, de usuario y manual de instalación.
- Trabajarlo sobre sistema operativo LINUX.
- Implementarlo en Java o QT.

4. GUÍA DEL INSTRUCTOR DE SISTEMAS OPERATIVOS 2

A continuación se presenta el desarrollo de la Guía del curso de Sistemas Operativos 2, de acuerdo al formato sugerido en el curso de “Estructuración de laboratorios”, impartido por el señor. Mrutunjaya Panda.

4.1. Datos generales

NOMBRE DE CURSO: Sistemas Operativos II (285)

PRE- REQUISITOS: Sistemas Operativos 1 (281)

POST – REQUISITOS: Ninguno

OBJETIVO: Este curso busca que el estudiante ponga en práctica los conocimientos adquiridos sobre Sistemas Operativos y los utilice para poder desarrollar efectivamente configuraciones y desarrollo en el núcleo del sistema.

MODULOS: Sistemas Operativos Avanzados usando Linux

4.2. Distribución del curso

Tabla XI: Distribución curso Sistemas Operativos 2

MODULO	TEORÍA	PRACTICA	TOTAL	EXAMEN	LAB ASSIGNMENTS	PROYECTOS
SISTEMAS OPERATIVOS AVANZADOS USANDO LINUX	22 horas	12 horas	34 horas	1 ½ horas	5	1
			34 horas			

4.3. Evaluación

Ponderación de tareas, prácticas y proyectos.

Tabla XII: Evaluación curso Sistemas Operativos 2

ACTIVIDAD	NUMERO	PORCENTAJE	TOTAL
TUTORIAL	2	2.5 %	5.00 %
LAB ASSIGNMENTS	5	8.0 %	40.00 %
PROYECTO	1	30.0 %	30.00 %
EXAMEN	1	25.0 %	25.00 %
TOTAL			100.00 %

4.4. Detalle de sesiones

Tabla XIII: Sesiones curso Sistemas Operativos 2

No	TEORIA	No	PRACTICA	TUTORIAL / PROYECTO / LAB ASSIGNMENT
1	Sistemas Operativos Avanzados usando Linux – 1	2	Sistemas Operativos Avanzados usando Linux – 1	LAB Assignment – 1
3	Sistemas Operativos Avanzados usando Linux – 2	4	Sistemas Operativos Avanzados usando Linux – 2	LAB Assignment – 2
5	Sistemas Operativos Avanzados usando Linux – 3	6	Sistemas Operativos Avanzados usando Linux – 3	LAB Assignment – 3
7	Sistemas Operativos Avanzados usando Linux – 4			
8	Sistemas Operativos Avanzados usando Linux – 5			Tutorial – 1
9	Sistemas Operativos Avanzados usando Linux - 6			Tutorial – 2

10	Sistemas Operativos Avanzados usando Linux - 7	11	Sistemas Operativos Avanzados usando Linux – 7	LAB Assignment – 4
12	Sistemas Operativos Avanzados usando Linux - 8			
13	Sistemas Operativos Avanzados usando Linux - 9			
14	Sistemas Operativos Avanzados usando Linux – 10	15	Sistemas Operativos Avanzados usando Linux – 10	LAB Assignment – 5
16	Sistemas Operativos Avanzados usando Linux - 11	17	Sistemas Operativos Avanzados usando Linux - 11	Proyecto – 1

4.5. Distribución de sesiones

Tabla XIV: Sesiones curso Sistemas Operativos 2

SESIÓN NO. 1:	
OBJETIVO:	Definir el concepto de un módulo y su implementación en el Sistema Operativo
CONTENIDO:	<ol style="list-style-type: none"> 1. Módulos Cargables <ol style="list-style-type: none"> 1.1. Conceptos 1.2. Compilación del kernel 1.3. Creación y Compilación 1.4. Carga manual 1.5. Carga dinámica 1.6. Ejemplo 1.7. Implementación
SESIÓN NO. 2:	
OBJETIVO:	Practicar los conceptos aprendidos en las sesión 1.
CONTENIDO:	Realización del LAB Assignment No. 1

SESIÓN NO. 3:	
OBJETIVO:	Definir el concepto básico de una tubería y como implementarla en Linux
CONTENIDO:	<ol style="list-style-type: none"> 1. Tuberías <ol style="list-style-type: none"> 1.1. Definición 1.2. Tipos 1.3. Funciones 1.4. Ejemplo de uso 1.5. Implementación 1.6. Otros mecanismos de comunicación <ol style="list-style-type: none"> 1.6.1. Spinlocks 1.6.2. Semáforos
SESIÓN NO. 4:	
OBJETIVO:	Practicar los conceptos aprendidos en las sesión 3.
CONTENIDO:	Realización del LAB Assignment No. 2
SESIÓN NO. 5:	
OBJETIVO:	Definir los conceptos sobre la virtualización y la instalación de Sistemas Operativos en máquinas virtuales.
CONTENIDO:	<ol style="list-style-type: none"> 1. Virtualización <ol style="list-style-type: none"> 1.1. Definición 1.2. Maquina Virtual <ol style="list-style-type: none"> 1.2.1. Xen <ol style="list-style-type: none"> 1.2.1.1. Características 1.2.1.2. Arquitectura Interna 1.2.1.3. Operación 1.2.2. KVM <ol style="list-style-type: none"> 1.2.2.1. Arquitectura Interna 1.2.2.2. Operación 1.2.3. VMware 1.2.4. Comparativa entre las diferentes tecnologías
SESIÓN NO. 6:	
OBJETIVO:	Practicar los conceptos aprendidos en las sesión 4.

CONTENIDO:	Realización del LAB Assignment No. 3
SESIÓN NO. 7:	
OBJETIVO:	Definir los conceptos sobre el funcionamiento de los Dispositivos de Entrada/Salida dentro el sistema operativo y su respectiva implementación dentro del sistema Linux.
CONTENIDO:	<ol style="list-style-type: none"> 1. Dispositivos de Entrada/Salida <ol style="list-style-type: none"> 1.1. Definición 1.2. Organización del sistema de E/S 1.3. Funciones del sistema de E/S 1.4. Principios de hardware/software 1.5. Dispositivos Udev <ol style="list-style-type: none"> 1.5.1. Conceptos 1.5.2. /dev estático y dinámico 1.5.3. Configuración 1.5.4. Reglas 1.5.5. Conflictos
SESIÓN NO. 8:	
OBJETIVO:	Definir los conceptos sobre el interbloqueo y su impacto dentro del funcionamiento del Sistema Operativo.
CONTENIDO:	<ol style="list-style-type: none"> 1. Interbloqueo <ol style="list-style-type: none"> 1.1. Definición 1.2. Recursos 1.3. Condiciones 1.4. Prevención 1.5. Algoritmo del banquero 1.6. Como evitarlo 1.7. Estrategias cuando se ha detectado 1.8. Implementación
SESIÓN NO. 9:	
OBJETIVO:	Definir los conceptos importantes sobre la administración de la memoria dentro del sistema operativo y su funcionamiento dentro del sistema Linux.

CONTENIDO:	<ol style="list-style-type: none"> 1. Administración de la Memoria <ol style="list-style-type: none"> 1.1. Conceptos 1.2. Funciones <ol style="list-style-type: none"> 1.2.1. Asignación de Memoria 1.2.2. Liberación de Memoria 1.3. Regiones de Memoria 1.4. Protección de la Memoria 1.5. Bloqueo de zonas de memoria 1.6. Proyección de archivos en memoria 1.7. Dispositivos SWAP 1.8. Páginas en memoria <ol style="list-style-type: none"> 1.8.1. Protección 1.8.2. Bloqueo 1.8.3. Sincronización 1.8.4. Operaciones <ol style="list-style-type: none"> 1.8.4.1. Segmentación 1.8.4.2. Paginación 1.8.4.3. Compartición 1.8.4.4. Descriptores 1.9. Memoria y el núcleo 1.10. Mapas de Memoria
SESIÓN NO. 10:	
OBJETIVO:	Definir los diferentes algoritmos de sustitución de páginas en un Sistema Operativo.
CONTENIDO:	<ol style="list-style-type: none"> 1. Algoritmos de Sustitución de Páginas <ol style="list-style-type: none"> 1.1. Conceptos 1.2. Sustitución de páginas 1.3. Algoritmo FIFO 1.4. Algoritmo Óptimo 1.5. Algoritmos de páginas menos recientemente utilizadas (LRU) 1.6. Algoritmos de conteo
SESIÓN NO. 11:	
OBJETIVO:	Practicar los conceptos aprendidos en las sesión 10.
CONTENIDO:	Realización del LAB Assignment No. 4

SESIÓN NO. 12:	
OBJETIVO:	Definir los conceptos básicos sobre la administración básica del sistema operativo Linux.
CONTENIDO:	<ul style="list-style-type: none"> 1. Administración del Sistema <ul style="list-style-type: none"> 1.1 Conceptos 1.2 Swappiness 1.3 Prioridad de procesos 1.4 Control de ejecución 1.5 Configuración Dinámica del Sistema 1.6 Implementación
SESIÓN NO. 13:	
OBJETIVO:	Definir los conceptos básicos sobre seguridad y enumerar políticas, principios y mecanismos para diseñar sistemas seguros.
CONTENIDO:	<ul style="list-style-type: none"> 1. Seguridad <ul style="list-style-type: none"> 1.1. Conceptos 1.2. Tipos de amenazas 1.3. Diseño de un sistema seguro <ul style="list-style-type: none"> 1.3.1. Principios 1.3.2. Técnicas 1.4. Servicios de seguridad 1.5. SELinux <ul style="list-style-type: none"> 1.5.1. Conceptos 1.5.2. Archivos de configuración 1.5.3. Control de accesos 1.5.4. Arranque
SESIÓN NO. 14:	
OBJETIVO:	Definir los conceptos básicos sobre sistemas distribuidos y enumerar los diferentes diseños y clasificaciones que se pueden implementar en sistemas distribuidos.

CONTENIDO:	<ol style="list-style-type: none"> 1. Sistemas Distribuidos <ol style="list-style-type: none"> 1.1. Definición 1.2. Ventajas y desventajas 1.3. Conceptos 1.4. Clasificación 1.5. Middleware 1.6. Diseño de procesos 1.7. Arquitectura peer-to-peer 1.8. Grids 1.9. Ldap
SESIÓN NO. 15:	
OBJETIVO:	Practicar los conceptos aprendidos en las sesión 14.
CONTENIDO:	Realización del LAB Assignment No. 5
SESIÓN NO. 16:	
OBJETIVO:	Esta sesión es por atrasos en las sesiones teóricas previas.
CONTENIDO:	
SESIÓN NO. 17:	
OBJETIVO:	Esta sesión es por atrasos en sesiones prácticas previas.
CONTENIDO:	

4.6. Detalle de tutoriales

ASSIGNMENTS:

Criterio de Evaluación:

- **Correcto** - **70**
- **Conocimiento Técnico** - **20**
- **Documentación** - **10**

Documentación a entregar: SI

- **Enunciado**

**MODULO: Sistemas Operativos Avanzados usando
Linux**

Tabla XV: Tutoriales curso Sistemas Operativos 2

TUTORIAL 1:	Evaluar los conceptos adquiridos en la sesión No. 8
OBJETIVO:	Comparar el rendimiento del algoritmo del Banquero.
Iniciada en sesión	8
Entregada en sesión	9
Viva / Demostración	NO
TUTORIAL 2:	Evaluar los conceptos adquiridos en la sesión No. 9
OBJETIVO:	Practicar los conocimientos adquiridos sobre los mapas de memoria del sistema operativo.
Iniciada en sesión	9
Entregada en sesión	10
Viva / Demostración	NO

4.7. Detalle de lab assignments

PRÁCTICAS:

Criterio de Evaluación:

- Funcionamiento - 75
- Presentación - 05
- Conocimiento Técnico - 10
- Documentación - 10

Documentación a entregar: SI

- Enunciado

MODULO: Sistemas Operativos Avanzados usando Linux

Tabla XVII: Prácticas curso Sistemas Operativos 2

LAB ASSIGNMENT 1:	Poner en práctica los conocimientos módulos.
OBJETIVO:	Implementar un módulo en el Sistema Operativo Linux.
Iniciada en sesión	1
Entregada en sesión	2
Viva / Demostración	SI
LAB ASSIGNMENT 2:	Poner en práctica los conocimientos sobre las tuberías.
OBJETIVO:	Implementar una tubería en el Sistema Operativo Linux.
Iniciada en sesión	3
Entregada en sesión	4
Viva / Demostración	SI

LAB ASSIGNMENT 3:	Poner en práctica los conocimientos sobre virtualización
OBJETIVO:	Instalar dos sistemas operativos en máquinas virtuales en distinta tecnología y hacer un análisis comparativo de rendimiento.
Iniciada en sesión	5
Entregada en sesión	6
Viva / Demostración	SI
LAB ASSIGNMENT 4:	Poner en práctica los conocimientos sobre paginación de memoria.
OBJETIVO:	Implementar los algoritmos de paginación de memoria del Sistema Operativo y realizar un análisis comparativo de rendimiento.
Iniciada en sesión	10
Entregada en sesión	11
Viva / Demostración	SI
LAB ASSIGNMENT 5:	Poner en práctica los conocimientos sobre sistemas distribuidos.
OBJETIVO:	Implementar un tipo de sistema distribuido en el Sistema Operativo Linux.
Iniciada en sesión	14
Entregada en sesión	15
Viva / Demostración	SI

4.8. Detalle del proyecto

PROYECTO:

Criterio de Evaluación:

- **Funcionamiento** - **70**
- **Conocimiento Técnico** - **10**
- **Documentación** - **20**

Documentación a entregar: SI

- **Enunciado**

MÓDULO: **Sistemas Operativos Avanzados usando Linux**

Tabla XVIII: Proyecto curso Sistemas Operativos 2

PROYECTO 1:	Poner en práctica los conocimientos del curso, así como su espíritu investigativo que le ayuda a obtener más conocimientos que le puedan servir en su vida de profesional.
OBJETIVO:	Implementar una llamada al Sistema de un comando propio del Sistema Operativo utilizando módulos en Linux.
Iniciada en sesión	10
Entregada en sesión	Examen Final
Viva / Demostración	SI

4.9. Detalle del examen

EXAMEN:

Número total de Exámenes	1
Examen realizado después de sesión	10
Temas para el examen	Sistemas Operativos Avanzados usando Linux

Duración 1 ½ hrs.

Criterio de Evaluación:

- **Objetivo** - **40%**
- **Subjetivo** - **60%**

4.10. EXAMEN DE EJEMPLO

1. ¿Cuales son los tipos de direccionamiento de un dispositivo de E/S?
2. ¿Que tipos de conversiones se realizan durante la transferencia de información?
3. ¿Cuales son los tipos de direccionamiento de un dispositivo de E/S?
4. ¿Que tipos de conversiones se realizan durante la transferencia de información?
5. ¿Como se implementa el algoritmo LRU?
6. ¿Como funciona el algoritmo de bit de referencia?
7. ¿Cuál es la diferencia entre los dos tipos de tuberías?
8. ¿Cuál es la función para crear una tubería sin nombre?
9. ¿Cuál es la función para crear una FIFO?
10. ¿Cual es la función de un modulo cargable?
11. ¿Describe los mandatos se utilizan para la carga manual de un modulo?
12. ¿Que programas se necesitan para realizar la carga dinámica?
13. ¿Describe es la llamada para configurar dinámicamente el sistema?
14. ¿Las llamadas al sistema para la modificación del nombre de la estación y su ámbito?
15. ¿Cuales son las modalidades para el cambio de estado del núcleo?
16. ¿Describe los requisitos básicos para proveer seguridad?
17. ¿Cuales son las etapas de un virus?
18. ¿Describe la estrategia que siguen los antivirus para mantener la seguridad?
19. ¿Cuales son las ventajas y desventajas de los sistemas distribuidos?
20. ¿Cuales son las ventajas y desventajas de los sistemas distribuidos?

Figura 2: Examen Ejemplo Sistemas Operativos 2

4.11. Bibliografía sugerida

1. Abraham Silberschatz & James, **Operating Systems Concepts.**
2. Francisco Manuel Marquez Addison Wesley, **Unix. Programación Avanzada**
3. Lina García Cabrera, Francisco Martínez del Rio & Miguel A. Redondo Duque, **Sistemas Operativos**
4. Andrew S. Tanenbaum, **Sistemas operativos modernos** 2a edición.

4.12. Lab handbook

Lab Assignment No. 1

Implementar un módulo en el sistema operativo.

Especificaciones:

- Desarrollar la implementación de un modulo del sistema.
- Utilizar librerías de c o c++, para la implementación del modulo.
- Realizar documentación técnica y de usuario.
- Trabajarlo sobre sistema operativo LINUX.

Lab Assignment No. 2

Implementar una tubería en el sistema operativo Linux.

Especificaciones:

- Desarrollar la implementación de una tubería
- Describir las librerías que se utilizan para implementar tuberías.

- Realizar documentación técnica y de usuario.
- Trabajarlo sobre sistema operativo Linux.

Lab Assignment No. 3

Instalar dos sistemas operativos en maquinas virtuales de distinta tecnología y hacer una análisis comparativo de rendimiento.

Especificaciones:

- Instalar un sistema operativo Linux en una máquina virtual de Windows.
- Instalar un sistema operativo Windows en una maquina virtual de Linux.
- Documentar los pasos de instalación de ambos sistemas operativos.
- Describir ventajas/desventajas de cada Maquina Virtual y compararlas.

Lab Assignment No. 4

Realizar simulación de los diferentes algoritmos de paginación de memoria del Sistema Operativo

Especificaciones:

- Implementar 3 o más algoritmos
- Realizar una comparación de rendimiento entre los diferentes algoritmos y determinar cual es el más eficiente.
- Realizar documentación técnica y de usuario.
- Trabajarlo sobre sistema operativo LINUX.
- Implementarlo en Java o QT.

Lab Assignment No. 5

Implementar un tipo de sistema distribuido en el Sistema Operativo Linux.

Especificaciones:

- Implementar gráficamente un tipo de sistema distribuido.
- Mostrar gráficamente el tipo de sistema seleccionado y describir las ventajas y desventajas del mismo.
- Realizar documentación técnica y de usuario.
- Trabajarlo sobre sistema operativo LINUX.
- Implementarlo en Java o QT.

Tutorial No. 1

Comparar el rendimiento del algoritmo del Banquero.

Especificaciones:

- Investigar el algoritmo del banquero en pseudocódigo.
- Comparar al menos dos casos de prueba de procesos utilizando este algoritmo.
- Incluir conclusiones y recomendaciones.

Tutorial No. 2

Practicar los conocimientos adquiridos sobre los mapas de memoria.

Especificaciones:

- Investigar los pasos generales para la realización de un mapa de memoria.
- Detallar como trabajan al menos dos sistemas operativos con los mapas

de memoria.

- Incluir conclusiones
- Incluir referencias bibliográficas.

Proyecto No. 1

Implementar una llamada al Sistema de un comando propio del Sistema Operativo utilizando módulos de Linux.

Especificaciones:

- Definir la llamada al sistema a capturar por parte de la aplicación.
- Simular el funcionamiento de la llamada seleccionada a través de módulos y que sea transparente para el usuario.
- Verificar el funcionamiento tanto en modo superusuario y como usuario normal.
- Realizar documentación técnica, de usuario y manual de instalación.
- Trabajarlo sobre sistema operativo LINUX con kernel 2.6.

5. GUÍA DEL INSTRUCTOR DE MANEJO E IMPLEMENTACIÓN DE ARCHIVOS

A continuación se presenta el desarrollo de la Guía del curso de Manejo e Implementación de Archivos, de acuerdo al formato sugerido en el curso de “Estructuración de laboratorios”, impartido por el señor. Mrutunjaya Panda.

5.1. Datos generales

NOMBRE DE CURSO: Manejo e Implementación de Archivos (773)

PRE- REQUISITOS: Estructuras de Datos (772)
Lenguajes Formales y de Programación (796)

POST – REQUISITOS: Sistemas de Bases de Datos I (774)
Redes de Computadoras I (970)

OBJETIVO: Este curso se estudian los elementos necesarios para que el estudiante tenga los conceptos básicos para comprender la forma en que los archivos soportan un sistema de gestión de base de datos, permitiendo, conocer la forma en que los datos deben ser almacenados para que su manejo y mantenimiento sean eficientes y óptimos. Adicionalmente, se da una introducción a los conceptos necesarios para entender la arquitectura de las bases de datos y la forma en que se administran y funcionan.

MÓDULOS: Introducción a los Sistemas de Archivos y Bases de Datos

5.2. Distribución del curso

Tabla XVIII: Distribución curso Manejo e Implementación de Archivos

MODULO	TEORÍA	PRACTICA	TOTAL	EXAMEN	LAB ASSIGNMENTS	PROYECTOS
INTRODUCCIÓN A LOS SISTEMAS DE ARCHIVOS Y BASES DE DATOS	16 horas	12 horas	28 horas	1 ½ horas	2	2
			28 horas			

5.3. Evaluación

Ponderación de tareas, prácticas y proyectos.

Tabla XIX: Evaluación curso manejo e implementación de archivos

ACTIVIDAD	NUMERO	PORCENTAJE	TOTAL
TUTORIAL	2	5.0 %	10.00 %
PRESENTACIÓN	1	5.0 %	5.00 %
LAB ASSIGNMENTS	2	10.0 %	10.00 %
PROYECTO	2	30.0 %	60.00 %
EXAMEN	1	15.0 %	15.00 %
TOTAL			100.00 %

5.4. Detalle de sesiones

Tabla XX: Sesiones curso Manejo e Implementación de Archivos

No	TEORIA	No	PRACTICA	TUTORIAL / PROYECTO / LAB ASSIGNMENT
1	Introducción a los Sistemas de Archivos y Bases de Datos - 1			Tutorial - 1
2	Introducción a los Sistemas de Archivos y Bases de Datos – 2			
3	Introducción a los Sistemas de Archivos y Bases de Datos – 3		Introducción a los Sistemas de Archivos y Bases de Datos – 3	LAB Assignment – 1
4	Introducción a los Sistemas de Archivos y Bases de Datos – 4		Introducción a los Sistemas de Archivos y Bases de Datos – 4	Proyecto – 1
5	Introducción a los Sistemas de Archivos y Bases de Datos – 5		Introducción a los Sistemas de Archivos y Bases de Datos – 5	
6	Introducción a los Sistemas de Archivos y Bases de Datos – 6		Introducción a los Sistemas de Archivos y Bases de Datos – 6	Tutorial – 2
7	Introducción a los Sistemas de Archivos y Bases de Datos – 7		Introducción a los Sistemas de Archivos y Bases de Datos – 7	LAB Assignment – 2
8	Introducción a los Sistemas de Archivos y Bases de Datos – 8		Introducción a los Sistemas de Archivos y Bases de Datos – 8	Proyecto – 2

5.5. Distribución de sesiones

Tabla XXI: Sesiones curso manejo e implementación de archivos

SESIÓN NO. 1:	
OBJETIVO:	Definir los diferentes dispositivos de almacenamiento más comunes y las características que estos poseen.
CONTENIDO:	1. Dispositivos de Almacenamiento 1.1. Definición

	<ul style="list-style-type: none"> 1.1.1. Almacenamiento 1.1.2. Dispositivos de almacenamiento 1.2. Tipos 1.3. Disco Duro <ul style="list-style-type: none"> 1.3.1. Características 1.3.2. Componentes 1.4. Memorias USB 1.5. DVD / CD 1.6. Otros Dispositivos de Almacenamiento <ul style="list-style-type: none"> 1.6.1. Diskette 1.6.2. Cintas magnéticas
SESIÓN NO. 2:	
OBJETIVO:	Definir el concepto de un sistema de archivos y enumerar los diferentes tipos que hay.
CONTENIDO:	<ul style="list-style-type: none"> 1. Sistemas de Archivos <ul style="list-style-type: none"> 1.1. Definición 1.2. FAT <ul style="list-style-type: none"> 1.2.1. Fat32 1.2.2. Fat16 1.2.3. VFat 1.3. NTFS 1.4. EXT2 1.5. Sistema de archivos basados en bitácoras <ul style="list-style-type: none"> 1.5.1. Ext3 1.5.2. RaiserFS 1.5.3. XFS 1.5.4. JFS
SESIÓN NO. 3:	
OBJETIVO:	Definir que es RAID e identificar los diferentes niveles y tipos que hay.
CONTENIDO:	<ul style="list-style-type: none"> 1. Manejo de Información <ul style="list-style-type: none"> 1.1. RAID (Redundant Array of Inexpensive Disks) <ul style="list-style-type: none"> 1.1.1. Definición 1.1.2. Tipo <ul style="list-style-type: none"> 1.1.2.1. Basados en Hardware 1.1.2.2. Basados en Software 1.1.2.3. Diferencias 1.1.2.4. Ventajas y Desventajas 1.1.3. Niveles

SESIÓN NO. 4:	
OBJETIVO:	Identificar los diferentes tipos de bases de datos, así como de DBMS y el funcionamiento de los mismos.
CONTENIDO:	<ol style="list-style-type: none"> 1. Introducción a Bases de Datos <ol style="list-style-type: none"> 1.1. Bases de Datos <ol style="list-style-type: none"> 1.1.1. Definición 1.1.2. Arquitectura 1.1.3. Modelos de Datos <ol style="list-style-type: none"> 1.1.3.1. Jerárquico 1.1.3.2. Red 1.1.3.3. Relacional 1.1.3.4. Objetos 1.1.4. Tipos 1.2. DBMS <ol style="list-style-type: none"> 1.2.1. Definición 1.2.2. Características 1.2.3. Tipos 1.2.4. Objetivos 1.2.5. Beneficios 1.2.6. Desventajas 1.2.7. Componentes 1.3. API's para el manejo de Base de Datos
SESIÓN NO. 5:	
OBJETIVO:	Identificar el lenguaje de definición de datos y la manipulación de las tablas
CONTENIDO:	<ol style="list-style-type: none"> 1. Tablas y lenguaje de definición de datos <ol style="list-style-type: none"> 1.1. Concepto 1.2. Elementos <ol style="list-style-type: none"> 1.2.1. Dominio 1.2.2. Relaciones 1.2.3. Llaves 1.3. Lenguaje de definición de Datos
SESIÓN NO. 6:	
OBJETIVO:	Identificar el lenguaje de manipulación de datos y sus funciones principales.

CONTENIDO:	<ul style="list-style-type: none"> 1. Lenguaje de Manipulación de Datos <ul style="list-style-type: none"> 1.1. Select <ul style="list-style-type: none"> 1.1.1. Distinct 1.1.2. Where <ul style="list-style-type: none"> 1.1.2.1. between 1.2. Insert 1.3. Update 1.4. Delete 1.5. Manejo de subconsultas <ul style="list-style-type: none"> 1.5.1. In 1.5.2. Exists 1.6. Strings 1.7. Funciones de agregación
SESIÓN NO. 7:	
OBJETIVO:	Definir que es algebra relacional y sus distintas operaciones.
CONTENIDO:	<ul style="list-style-type: none"> 1. Algebra Relacional <ul style="list-style-type: none"> 1.1. Selección 1.2. Proyección 1.3. Asignación 1.4. Unión 1.5. Diferencia 1.6. Intersección 1.7. Producto Cartesiano 1.8. Reunión 1.9. División
SESIÓN NO. 8:	
OBJETIVO:	Definir que es un modelo relacional y como diseñarlo.
CONTENIDO:	<ul style="list-style-type: none"> 1. Modelo Relacional <ul style="list-style-type: none"> 1.1. Introducción 1.2. Definición 1.3. Elementos <ul style="list-style-type: none"> 1.3.1. Entidades 1.3.2. Propiedades 1.3.3. Relaciones 1.3.4. Ejemplos

5.6. Detalle de tutoriales

ASSIGNMENTS:

Criterio de Evaluación:

- Correcto - 70
- Conocimiento Técnico - 20
- Documentación - 10

Documentación a entregar: SI

- Enunciado

MÓDULO: Introducción a los Sistemas de Archivos y Bases de Datos

Tabla XXII: Tutoriales curso manejo e implementación de archivos

TUTORIAL 1:	Evaluar los conceptos adquiridos en la sesión No. 1
OBJETIVO:	Comparar entre los diferentes métodos de indexación de acuerdo sus características principales.
Iniciada en sesión	1
Entregada en sesión	2
Viva / Demostración	NO
TUTORIAL 2:	
OBJETIVO:	Investigar acerca de los conceptos de bases de datos distribuidas.
Iniciada en sesión	6
Entregada en sesión	7
Viva / Demostración	NO

5.7. Detalle de lab assignments

PRÁCTICAS:

Criterio de Evaluación:

- **Funcionamiento** - **75**
- **Presentación** - **05**
- **Conocimiento Técnico** - **10**
- **Documentación** - **10**

Documentación a entregar: SI

- **Enunciado**

MODULO: Introducción a los Sistemas de Archivos y Bases de Datos

Tabla XXIII: Prácticas curso Manejo e Implementación de Archivos

LAB ASSIGNMENT 1:	Poner en práctica los conocimientos adquiridos a lo largo del módulo.
OBJETIVO:	Visualizar los i-nodos en el sistema y la estructura de la que están compuestos.
Iniciada en sesión	2
Entregada en sesión	3
Viva / Demostración	SI
LAB ASSIGNMENT 2:	Poner en práctica los conocimientos los modelos relacionales.
OBJETIVO:	Practicar la realización de modelos relaciones complejos.
Iniciada en sesión	8
Entregada en sesión	Examen Final
Viva / Demostración	SI

5.8. Detalle del proyecto

PROYECTO:

Criterio de Evaluación:

- Funcionamiento - 70
- Conocimiento Técnico - 10
- Documentación - 20

Documentación a entregar: SI

- Enunciado

MÓDULO: Introducción a los Sistemas de Archivos y Bases de Datos

Tabla XXIV: Proyecto curso Manejo e Implementación de Archivos

PROYECTO 1:	Poner en práctica los conocimientos del curso, así como su espíritu investigativo que le ayuda a obtener más conocimientos que le puedan servir en su vida de profesional.
OBJETIVO:	Desarrollar un sistema administrador de base datos, aplicando los conceptos adquiridos en las sesiones posteriores.
Iniciada en sesión	4
Entregada en sesión	5
Viva / Demostración	SI
PROYECTO 2:	Poner en práctica los conocimientos del curso, así como su espíritu investigativo que le ayuda a obtener más conocimientos que le puedan servir en su vida de profesional.
OBJETIVO:	Resolver un problema de modelos relacionales aplicando los conceptos de algebra relacional aprendidos en las sesiones posteriores.

Iniciada en sesión	8
Entregada en sesión	Examen Final
Viva / Demostración	SI

5.9. Detalle del examen

EXAMEN:

Número total de Exámenes	1
Examen realizado después de sesión	10
Temas para el examen	Introducción a los Sistemas de Archivos y Bases de Datos
Duración	1 ½ hrs.
Criterio de Evaluación:	
• Objetivo	- 40%
• Subjetivo	- 60%

5.10. Examen de ejemplo

1. ¿Cuál es la definición de almacenamiento?
2. ¿Cuál es la definición de disco duro?
3. ¿Cómo está estructurado el sistema ext2?
4. ¿Ventajas de los sistemas de archivos basados en bitácoras?
5. ¿Cuáles son las ventajas del RAID 5?
6. ¿Cuáles son los beneficios y las desventajas de un DBMS?
7. ¿Describa los componentes de un DBMS?
8. ¿Cuál es la sentencia para la definición del esquema de una tabla?
9. ¿Cuál es la sentencia para la definición del dominio?
10. ¿Cuáles son las sentencias para subconsultas?
11. ¿Cuáles son las sentencias para el manejo de strings?
12. ¿Cuáles la definición de álgebra relacional?
13. ¿Cuál es la clasificación de las operaciones de álgebra relacional?
14. ¿Qué incorpora el modelo relacional?
15. ¿Cuál es la definición de entidad?

Figura 3: Examen ejemplo manejo e implementación de Archivos

5.11. Bibliografía sugerida

1. C.J. Date, **An introduction to Database Systems**, Fifth edition
2. Folk, Michael J., Zoellick, Bill (1,992). **Estructuras de archivos un conjunto de herramientas conceptuales** (Edición en Español). Editorial Addison-Wesley Iberoamericana. USA.
3. Loomis, Mary E.S. (1,989) **Estructuras de datos y organización de archivos** (Segunda Edición). Editorial Prentice Hall.
4. Wiederhold, Gio (1,987). **Diseño de base de datos** (Segunda Edición). Editorial McGraw Hill. México.

5.12. Lab handbook

Lab Assignment No. 1

Visualizar los i-nodos en el sistema y la estructura de la que están compuestos.

Especificaciones:

- Definir las estructuras que se utilizan para guardar la información de los i-nodos.
- Utilizar librerías del sistema de c ó c++, para la utilización de los i-nodos.
- Desarrollar una interfaz gráfica para poder interactuar con los i-nodos.
- Trabajarlo sobre sistema operativo LINUX.
- Implementarlo en Java o QT.

Lab Assignment No. 2

Practica la realización de modelos relacionales complejos.

Especificaciones:

- Diseñar el modelo relacional de al menos 3 sistemas de información con un nivel de complejidad alto, es decir que su número de entidades y relaciones sea considerable.
- Implementar los modelos diseñados en un sistema de bases de datos (MySQL o Oracle)
- Realizar de consultas y reportes sobre información ingresada a los modelos implementados.
- Realizar documentación.

Tutorial No. 1

Comparar entre los diferentes métodos de indexación de acuerdo a sus características principales.

Especificaciones:

- Investigar al menos 3 métodos de indexación.
- Realizar un análisis comparativo entre los diferentes métodos.
- Incluir conclusiones.
- Incluir referencias bibliográficas

Tutorial No. 2

Investigar acerca de los conceptos de Bases de Datos Distribuidas.

Especificaciones

- Investigar términos básicos de una base distribuida.
- Describir ventajas/desventajas de las bases distribuidas.
- Describir el futuro de las bases de datos distribuidas.
- Incluir conclusiones.

- Incluir referencias bibliográficas.

Proyecto No. 1

Desarrollar un sistema administrador de base de datos, aplicaciones los conceptos adquiridos en las sesiones recibidas.

Especificaciones

- Definir los componentes principales a implementar de un sistema administrador de base de datos.
- Realizar documentación técnica, de usuario y manual de instalación.
- Trabajarlo sobre sistema operativo LINUX.
- Implementarlo en Java o QT.

Proyecto No. 2

Resolver un problema de modelos relacionales aplicando los conceptos de algebra relacional aprendidos en las sesiones recibidas.

Especificaciones

- Diseñar un modelo relacional con aplicación practica.
- Integrar el manejo de usuarios y roles.
- Mostrar reportes según definición de consultas específicas.
- Realizar documentación técnica, de usuario y manual de instalación.
- Trabajarlo sobre sistema operativo LINUX.
- Implementarlo en Java o QT.

6. Documentación de Apoyo de “Sistemas Operativos 1”

A continuación se presenta el desarrollo de la documentación de apoyo del curso de Sistemas Operativos 1, el cual se realizó en formato de un libro conteniendo las unidades más importantes impartidas a lo largo del curso.

Datos generales

NOMBRE DE CURSO: Sistemas Operativos I (281)

PRE- REQUISITOS: Organización de Lenguajes y Compiladores 2 (781)

Arquitectura de Computadores y Ensambladores 1 (778)

POST – REQUISITOS: Sistemas Operativos 2 (285)

OBJETIVO: Este libro esta destinado para los estudiantes del curso de Sistemas Operativos 1, contiendo teoría, ejemplos, diagramas, resumen, glosario y artículos de interés relacionados a cada una de las unidades del curso.

MODULOS: Introducción a los Sistemas Operativos

Figura 4 – Páginas 1-4 Libro “Introducción a los Sistemas Operativos”



Figura 5 – Páginas 5-8 Libro “Introducción a los Sistemas Operativos”

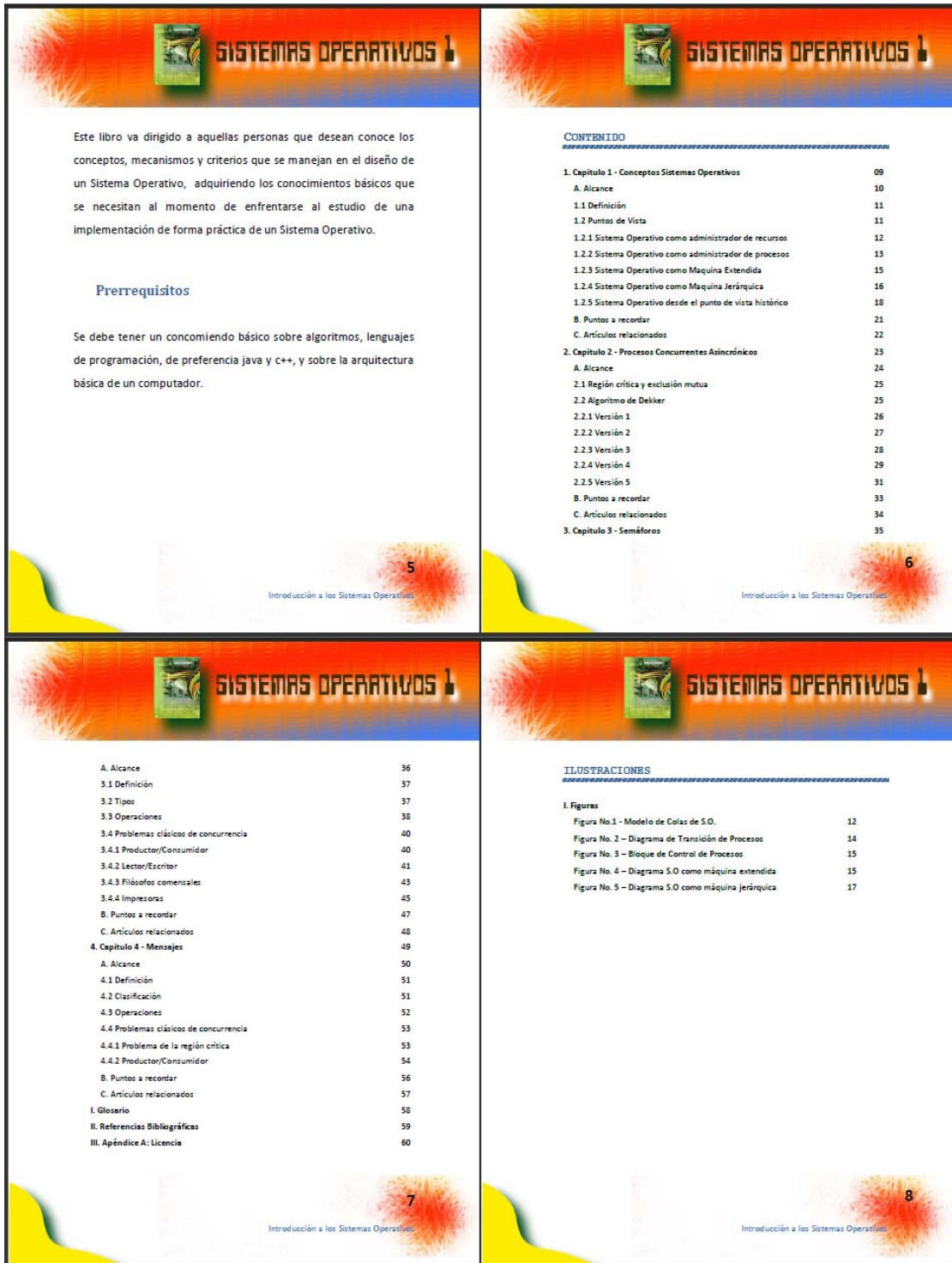


Figura 6 – Páginas 9-12 Libro “Introducción a los Sistemas Operativos”

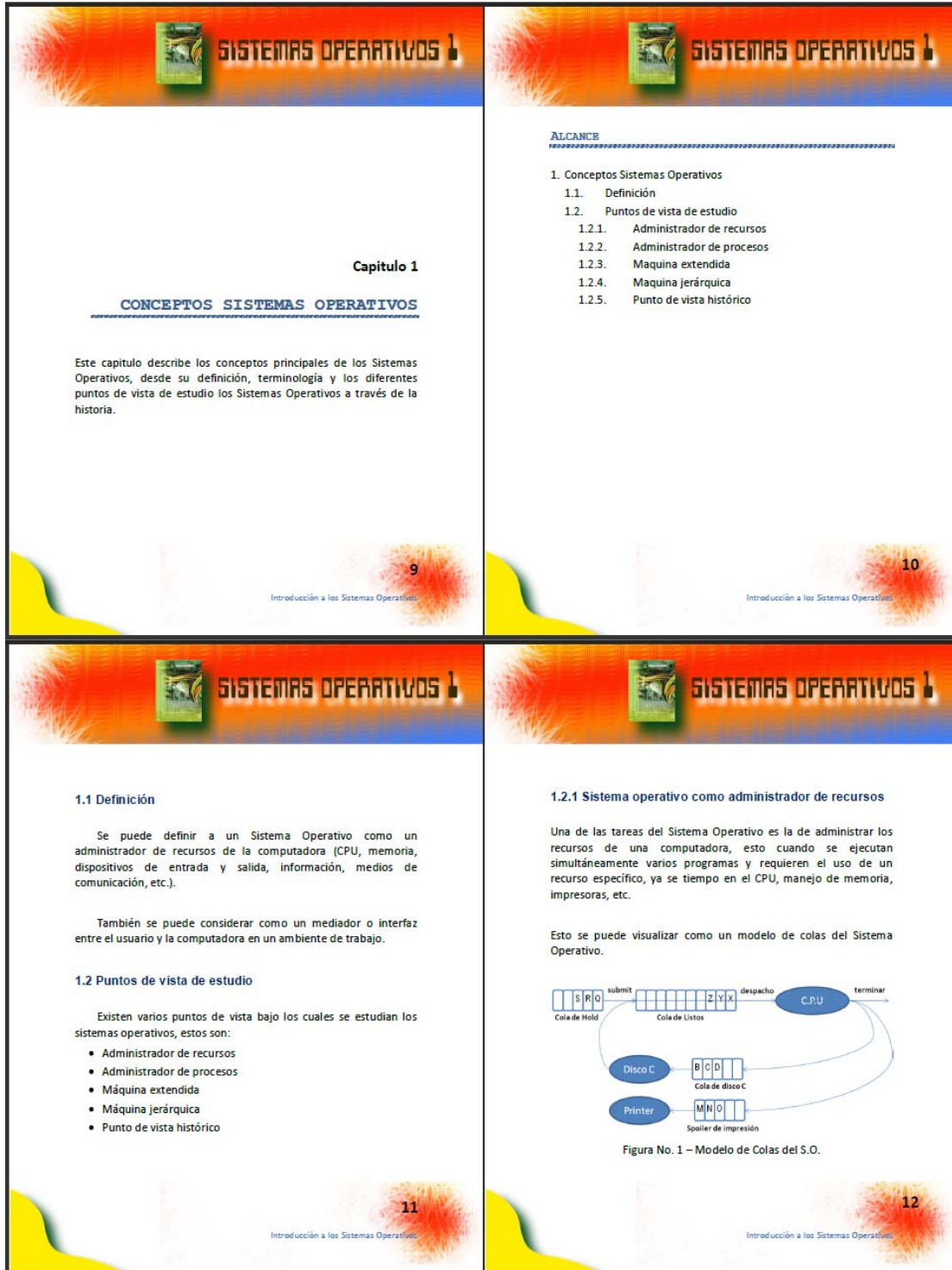


Figura 7 – Páginas 13-16 Libro “Introducción a los Sistemas Operativos”

SISTEMAS OPERATIVOS

1.2.2 Sistema operativo como administrador de procesos

Para estudiar este punto de vista, es necesario definir varios conceptos importantes:

Programa
Se puede definir a un programa con un archivo que contiene un conjunto de instrucciones que efectúan una tarea, como tal, es una entidad pasiva, es decir que no tiene recursos asignados.

Proceso
Un proceso se puede definir como un programa en ejecución que tiene recursos asignados, es una entidad activa es decir que esta viva.

Paralelismo
El paralelismo se refiere a varios procesadores trabajando simultáneamente en la realización de tareas.

Concurrencia
La concurrencia se refiere a una sola unidad funcional, es decir que trabaja con un solo proceso a la vez.

13

Introducción a los Sistemas Operativos

SISTEMAS OPERATIVOS

Figura No. 2 – Diagrama de Transición de Procesos

Bloque de Control de Procesos (PCB)
Se puede definir al bloque de control de procesos como una estructura de datos propia del Sistema Operativo que tiene la función de llevar el control de los procesos y los recursos asignados que cada uno de ellos tiene.

14

Introducción a los Sistemas Operativos

SISTEMAS OPERATIVOS

PID	Nombre	Estado	Prioridad	Memoria
A	Tesis	W	2	
B	Planilla	R	2	
C	-----	L	1	

Figura No. 3 – Bloque de Control de Procesos

1.2.3 Sistema Operativo como Máquina Extendida

Figura No. 4 – Diagrama S.O como máquina extendida

15

Introducción a los Sistemas Operativos

SISTEMAS OPERATIVOS

Kernel
Se define el Kernel como el conjunto de instrucciones del sistema Operativo que interactúan directamente sobre el hardware.

Sistema Operativo Externo
Se define como Sistema Operativo externo al conjunto e instrucciones del Sistema Operativo que interactúan sobre el kernel y no sobre el hardware.

Shell
Por Shell se refiere al conjunto de instrucciones del sistema operativo que sirve de interfaz entre la computadora (hardware + sistema operativo) con el usuario.

1.2.4 Sistema Operativo como Máquina Jerárquica

16

Introducción a los Sistemas Operativos

Figura 8 – Páginas 17-20 Libro “Introducción a los Sistemas Operativos”

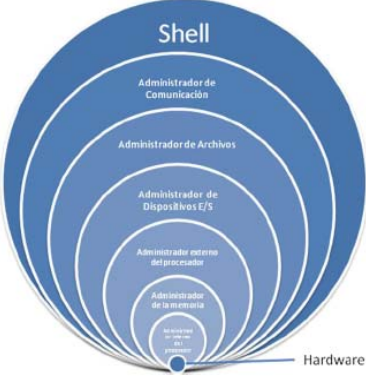
<p style="text-align: center;">SISTEMAS OPERATIVOS</p>  <p style="text-align: center;">Figura No. 5 – Diagrama S.O. como máquina jerárquica</p> <p style="text-align: right;">17</p> <p style="text-align: center;">Introducción a los Sistemas Operativos</p>	<p style="text-align: center;">SISTEMAS OPERATIVOS</p> <h3>1.2.5 Sistema Operativo desde el punto de vista histórico</h3> <p>Los Sistemas Operativos han evolucionado de la siguiente forma:</p> <ul style="list-style-type: none"> • 1940 – Primeras Computadoras <ul style="list-style-type: none"> ○ En esta etapa los computadores eran grandes máquinas que se ejecutaban desde consola, solo era para ser usada por un usuario a la vez, se manejaban tarjetas perforadas o cintas programadas y la programación y depuración se realizaba por medio de switches y focos de luz. ○ El software estaba distribuido en ensambladores, cargadores, compiladores, drivers y librerías de subrutinas comunes. • 1950 – Sistemas Batch <ul style="list-style-type: none"> ○ En esta etapa se contiene un lector de tarjetas, las cuales contienen un job a ejecutarse. ○ Se define el primer sistema operativo rudimentario, que no es más que un monitor residente que ejecuta los jobs en una secuencia específica, este contiene un controlador que lo transfiere al job y cuando este termina de ejecutar es regresado al monitor. <p style="text-align: right;">18</p> <p style="text-align: center;">Introducción a los Sistemas Operativos</p>
<p style="text-align: center;">SISTEMAS OPERATIVOS</p> <ul style="list-style-type: none"> • 1960 – Timesharing y Multiprogramación <ul style="list-style-type: none"> ○ En esta etapa se manejan varios jobs, los cuales se conservan en memoria al mismo tiempo y la CPU es compartida entre ellos. ○ Se manejan rutinas de E/S, administración de memoria administración de los dispositivos y del CPU para el control de la ejecución de los jobs. • 1970 – Microprocesadores y minicomputadoras <ul style="list-style-type: none"> ○ En esta etapa los computadores han disminuido considerablemente su tamaño, y se inicia el desarrollo de Sistemas Operativos como UNIX, DOS y CP/M. ○ Los sistemas operativos tienen una mejora en su interfaz de usuario, se introducen los microprocesadores y se desarrollan los lenguajes de programación. • Finales de los 70's y 80's – Computadoras personales, sistema de redes, distribuidos y tiempo real. <ul style="list-style-type: none"> ○ En esta etapa se aparecen las computadoras personales, que son dedicados para un solo usuario, se mejora la velocidad de respuesta y el manejo de los dispositivos de E/S como el teclado, mouse, etc. ○ También aparecen los Sistemas Distribuidos, los cuales se distribuyen entre varios computadores geográficamente dispersos. <p style="text-align: right;">19</p> <p style="text-align: center;">Introducción a los Sistemas Operativos</p>	<p style="text-align: center;">SISTEMAS OPERATIVOS</p> <ul style="list-style-type: none"> ○ También los Sistemas de Redes, como la organización cliente/servidor, servicios y protocolos de comunicación, seguridad y consistencia de la información. ○ Por último los Sistemas Paralelos, que son sistemas con múltiples procesadores que se comunican entre ellos. • 1990 – La web WWW, sistemas móviles, pc's. <ul style="list-style-type: none"> ○ En esta etapa la red de comunicación se ha extendido a distancia mundial con la aparición del World Wide Web. ○ También el apareamiento de las Notebooks, con tecnología de comunicación inalámbrica, dando origen al Computo Móvil. <p style="text-align: right;">20</p> <p style="text-align: center;">Introducción a los Sistemas Operativos</p>

Figura 9 – Páginas 21-24 Libro “Introducción a los Sistemas Operativos”

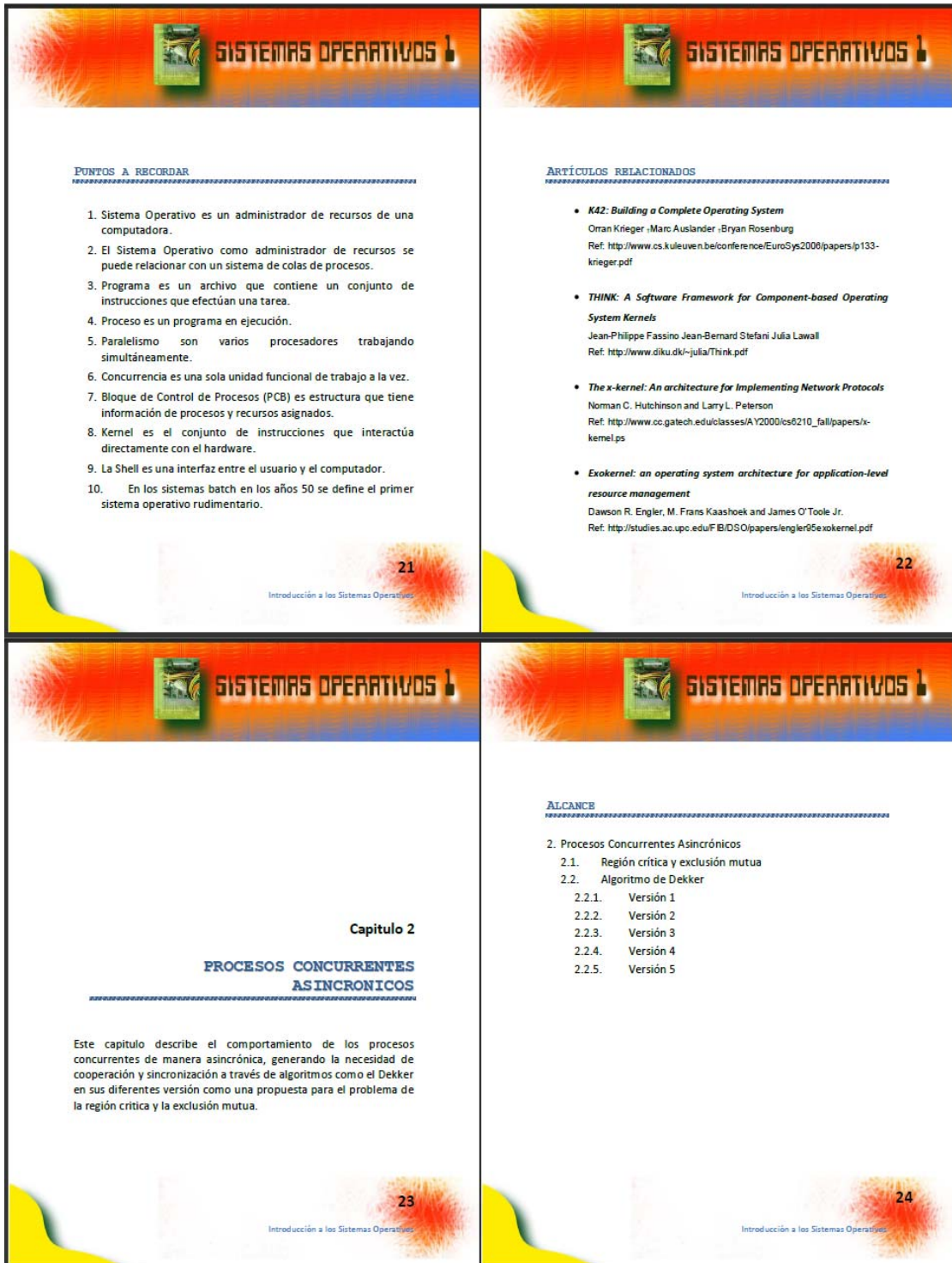


Figura 10 – Páginas 25-28 Libro “Introducción a los Sistemas Operativos”

SISTEMAS OPERATIVOS

2.1 Región crítica y exclusión mutua

Quando se habla de un proceso concurrente, se refiere a procesos que funcionan de manera independiente, unos de otros pero al mismo tiempo, es decir de manera asíncrona, por lo que es necesaria cierta cooperación o sincronización.

Por lo tanto cuando dos o más procesos llegan al ejecutarse al mismo tiempo, se le llaman concurrencia de procesos.

Para que dos o más procesos sean concurrentes, deben tener alguna relación entre ellos, como por ejemplo la cooperación para un determinado trabajo, uso de la misma información o de otro recurso de manera compartida.

Se puede definir como región crítica como un conjunto de recursos (Lógicos o Físicos).

Quando se habla de Exclusión Mutua, se refiere a garantizar que si un proceso está utilizando algún recurso compartido, los demás procesos no podrán utilizarlo en ese tiempo.

2.2 Algoritmo de Dekker

Este algoritmo de programación concurrente fue implementado por Edsger Dijkstra para resolver la exclusión mutua, su funcionalidad era permitir que dos procesos que estuvieran en

25

Introducción a los Sistemas Operativos

SISTEMAS OPERATIVOS

ejecución simultáneamente, pudieran compartir un recurso sin tener conflictos entre ellos.

De este algoritmo fueron desarrolladas 5 versiones, siendo la última la que trabaja más eficientemente al resolver los fallos de las primeras cuatro.

2.2.1 Versión 1

```

Proceso UNO
while TRUE do
  hace_cosas
  while Turno = 2 do:
    REGION CRITICA
    Turno = 1
  hace_mas_cosas
            
```

```

Proceso DOS
while TRUE do
  hace_algo
  while Turno = 1 do:
    REGION CRITICA
    Turno = 2
  hace_mas_de_algo
            
```

26

Introducción a los Sistemas Operativos

SISTEMAS OPERATIVOS

ⓘ

Problema: Alternancia Estricta

En esta versión se garantiza la exclusión mutua, pero tiene una desventaja en que los procesos son fuertemente acoplados, es decir que los procesos lentos atrasan a los que son más rápidos.

2.2.2 Versión 2

```

var P1QE, P2QE: boolean

Proceso UNO
while TRUE do
  hace_cosas
  P1QE = TRUE
  while P2QE do:
    REGION CRITICA
  P1QE = FALSE
  hace_mas_cosas
            
```

```

Proceso DOS
while TRUE do
  hace_algo
  P2QE = TRUE
  while P1QE do:
    REGION CRITICA
  P2QE = FALSE
  hace_mas_de_algo
            
```

27

Introducción a los Sistemas Operativos

SISTEMAS OPERATIVOS

ⓘ

Problema: Interbloqueo

En esta versión se evita el problema de la alternancia, pero existe la posibilidad que los procesos que se están ejecutando caigan en un mismo estado y no salgan de allí.

2.2.3 Versión 3

```

var P1QE, P2QE: boolean

Proceso UNO
while TRUE do
  hace_cosas
  while P2QE do:
    P1QE = TRUE
  REGION CRITICA
  P1QE = FALSE
  hace_mas_cosas
            
```

```

Proceso DOS
while TRUE do
  hace_algo
  while P1QE do:
    P2QE = TRUE
  REGION CRITICA
  P2QE = FALSE
  hace_mas_de_algo
            
```

28

Introducción a los Sistemas Operativos

Figura 11 – Páginas 29-32 Libro “Introducción a los Sistemas Operativos”

SISTEMAS OPERATIVOS

Problema: No se garantiza la exclusión mutua
 El problema con esta versión es que no puede evitar que en algún momento dos procesos accedan al mismo tiempo a la región crítica.

2.2.4 Versión 4

```

Proceso UNO
while TRUE do
  hace_casos
  P1QE=TRUE
  while P2QE do
    begin
      P1QE=FALSE
      delay(random)
      P1QE=TRUE
    end
  REGION CRÍTICA
  P1QE=FALSE
  hace_mas_casos
    
```

SISTEMAS OPERATIVOS

```

Proceso DOS
while TRUE do
  hace_algo
  P2QE=TRUE
  while P2QE do
    begin
      P2QE=FALSE
      delay(random)
      P2QE=TRUE
    end
  REGION CRÍTICA
  P2QE=FALSE
  hace_mas_de_algo
    
```

Problema: Postergación indefinida
 Este problema se refiere a que uno o varios procesos se quedan a la espera de algún evento, el cuanto puede que nunca ocurra.

SISTEMAS OPERATIVOS

2.2.5 Versión 5

```

Proceso UNO
while TRUE do
  hace_casos
  P1QE=TRUE
  while P2QE do
    if TURNO = 2 then
      begin
        P1QE=FALSE
        delay(random)
        P1QE=TRUE
      end
    REGION CRÍTICA
    P1QE=FALSE
    TURNO = 2
    hace_mas_casos
    
```

```

Proceso DOS
while TRUE do
  hace_algo
  P2QE=TRUE
  while P1QE do
    if TURNO = 1 then
      begin
        P2QE=FALSE
        delay(random)
        P2QE=TRUE
      end
    REGION CRÍTICA
    P2QE=FALSE
    TURNO = 1
    hace_mas_de_algo
    
```

El algoritmo de Dekker en su quinta versión resuelve el problema de la exclusión mutua con el estado de “Espera Ocupada”.

Figura 12 – Páginas 33-36 Libro “Introducción a los Sistemas Operativos”

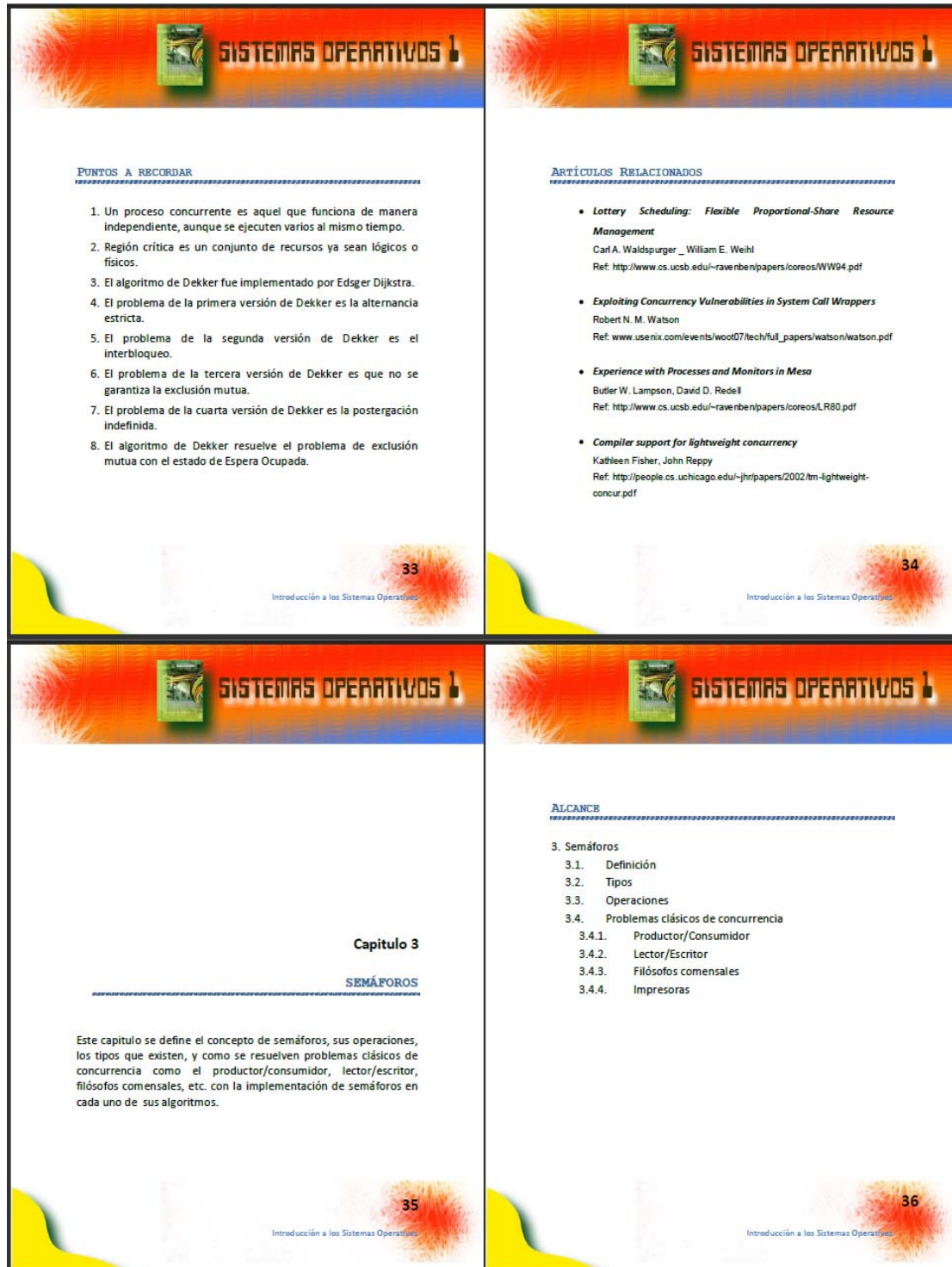


Figura 13 – Páginas 37-40 Libro “Introducción a los Sistemas Operativos”

SISTEMAS OPERATIVOS

3.1 Definición

Se puede definir a un semáforo como una variable protegida que solamente puede ser accedida por las operaciones “is”, “up”, “down” y que administra la concurrencia sobre un recurso compartido.

Semáforo: S

3.2 Tipos

Se pueden identificar dos tipos:

<p>Binarios</p> <p>0 → rojo</p> <p>1 → verde</p>	<p>Contadores</p> <p>0 → rojo</p> <p>1 → verde</p> <p>2 → verde</p> <p>...</p> <p>n → verde</p>
---	--

37

Introducción a los Sistemas Operativos

SISTEMAS OPERATIVOS

3.3 Operaciones

Con los semáforos se identifican las siguientes operaciones:

IS → inicializa el semáforo

is(s,n) → s = n

down(s)

CLI (se deshabilitan las interrupciones)

if s > 0 then

s = s - 1

else

BLOQUEAR_PROCESO

STI (se habilitan las interrupciones)

up(s)

CLI

If HAY_PROCESO_DORMIDO then

DESPERTAR

else

s = s + 1

STI

38

Introducción a los Sistemas Operativos

SISTEMAS OPERATIVOS

Resolución del problema de la región crítica con semáforos:

```

IS(S, 1)

Proceso UNO
while TRUE do
  hace_cosas
  down(s)
  REGION CRÍTICA
  up(s)
  hace_mas_cosas

Proceso DOS
while TRUE do
  hace_algo
  down(s)
  REGION CRÍTICA
  up(s)
  hace_mas_de_algo
    
```

39

Introducción a los Sistemas Operativos

SISTEMAS OPERATIVOS

3.4 Problemas clásicos de concurrencia

3.4.1 Algoritmo del productor/consumidor

Se utilizan los siguientes semáforos:

<p>Contadores:</p> <p>Is(vacio,1)</p> <p>Is(lleno,0)</p>	<p>Binario:</p> <p>Is(excmut,1)</p>
---	--

```

Productor()
while TRUE do
  hace_cosas
  x := PRODUCE_ALGO
  down(vacio)
  down(excmut)
  push(x)
  up(excmut)
  up(lleno)
  hacer_mas_cosas
    
```

40

Introducción a los Sistemas Operativos

Figura 14 – Páginas 41-44 Libro “Introducción a los Sistemas Operativos”

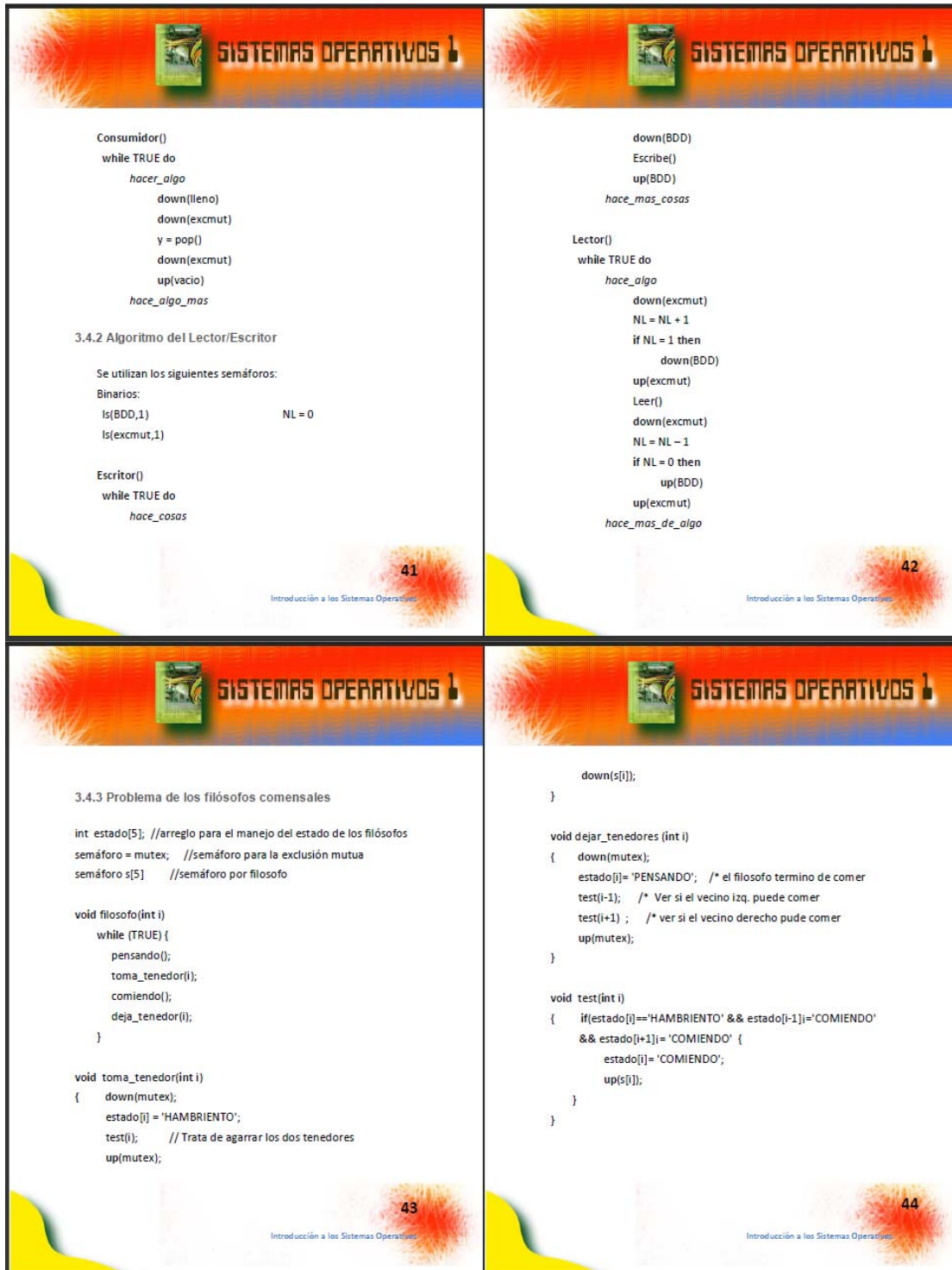


Figura 15 – Páginas 45-48 Libro “Introducción a los Sistemas Operativos”





<p style="text-align: center;"> SISTEMAS OPERATIVOS</p> <p>3.4.4 Problema de las impresoras</p> <p>Se tienen 3 impresoras y hay cierta cantidad de procesos a las impresoras.</p> <table border="0"> <tr> <td>Estatus</td> <td style="text-align: center;">Sem</td> </tr> <tr> <td>Q → Quiere entrar</td> <td style="text-align: center;">□</td> </tr> <tr> <td>I → Imprimiendo</td> <td style="text-align: center;">□</td> </tr> <tr> <td>C → hace_cosas</td> <td style="text-align: center;">□</td> </tr> <tr> <td>O → hace_otras_cosas</td> <td style="text-align: center;">□</td> </tr> </table> <pre> Dejar_impresora(i) down(excmut) ESTATUS[i] = 'O' N = N - 1 K = 0 while(K < max_proc and (N < max_imp) { Test(K) K = K + 1 } </pre> <p style="text-align: right;">45</p> <p style="text-align: center; font-size: small;">Introducción a los Sistemas Operativos</p>	Estatus	Sem	Q → Quiere entrar	□	I → Imprimiendo	□	C → hace_cosas	□	O → hace_otras_cosas	□	<p style="text-align: center;"> SISTEMAS OPERATIVOS</p> <pre> up(excmut) Test(i) if(N < max_imp and ESTATUS[i] = 'Q' { N = N + 1 asigna_imp() up(sem[i]) ESTATUS[i] = 'I' } Tomar_impresora(i) down(excmut) ESTATUS[i] = 'H' Test(i) up(excmut) down(sem[i]) </pre> <p style="text-align: right;">46</p> <p style="text-align: center; font-size: small;">Introducción a los Sistemas Operativos</p>
Estatus	Sem										
Q → Quiere entrar	□										
I → Imprimiendo	□										
C → hace_cosas	□										
O → hace_otras_cosas	□										
<p style="text-align: center;"> SISTEMAS OPERATIVOS</p> <p><u>PUNTOS A RECORDAR</u></p> <ol style="list-style-type: none"> 1. Un semáforo es una variable protegida que solo puede ser accedida por sus operaciones definidas. 2. Las operaciones de un semáforo son: inicialización (is), encendido (up), apagado (down). 3. Un semáforo administra la concurrencia sobre un registro compartido. 4. Los tipos de semáforo son binarios y contadores. 5. Un semáforo binario solo puede tener los valores 1(verde) y 0(rojo). <p style="text-align: right;">47</p> <p style="text-align: center; font-size: small;">Introducción a los Sistemas Operativos</p>	<p style="text-align: center;"> SISTEMAS OPERATIVOS</p> <p><u>ARTÍCULOS RELACIONADOS</u></p> <ul style="list-style-type: none"> • <i>Towards a Model of Fair and Unfair Semaphores in MoDeST</i> Jens Happe Ref: http://pastaworkshop.org/2007/proceedings/fair-semaphores.pdf • <i>An Efficient Semaphore Implementation Scheme for Small-Memory Embedded Systems</i> Khawar M. Zuberi and Kang G. Shin Ref: kabru.eecs.umnich.edu/papers/publications/1997/zuberi_rtas97.ps • <i>Analyzing Semaphores Using "Smart" Models</i> Thaddeus Westerson, Jeffrey Hargrave and Holden Peebles Ref: http://hargrave.csail.mit.edu/smart.pdf • <i>Process Scheduling and UNIX Semaphores</i> NEIL DUNSTAN AND IVAN FRIS Ref: www.cs.ubc.ca/local/reading/proceedings/spe91-95/spe/vol25/issue10/spe983nd.pdf <p style="text-align: right;">48</p> <p style="text-align: center; font-size: small;">Introducción a los Sistemas Operativos</p>										

Figura 16 – Páginas 49-52 Libro “Introducción a los Sistemas Operativos”

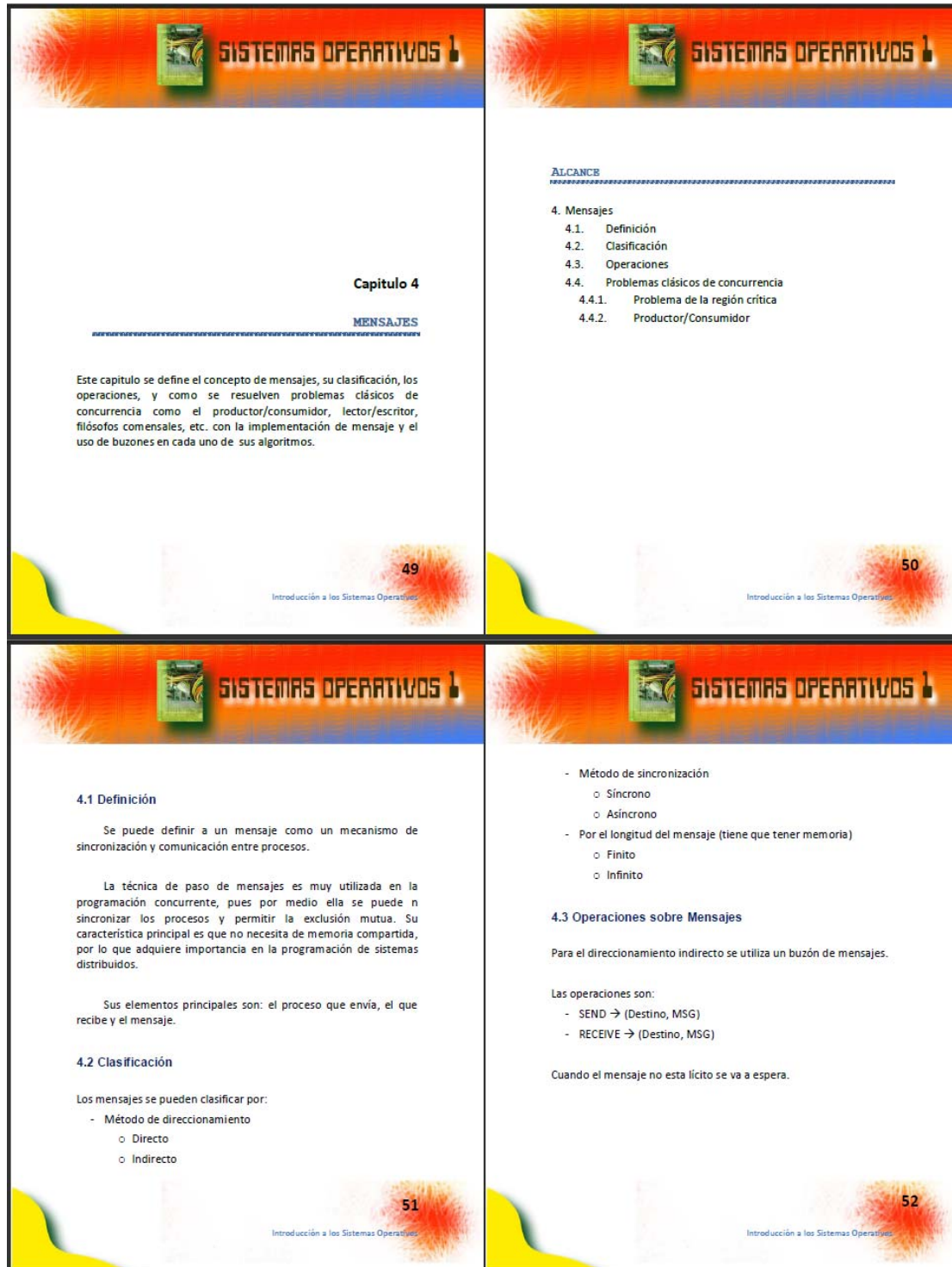


Figura 17 – Páginas 53-56 Libro “Introducción a los Sistemas Operativos”

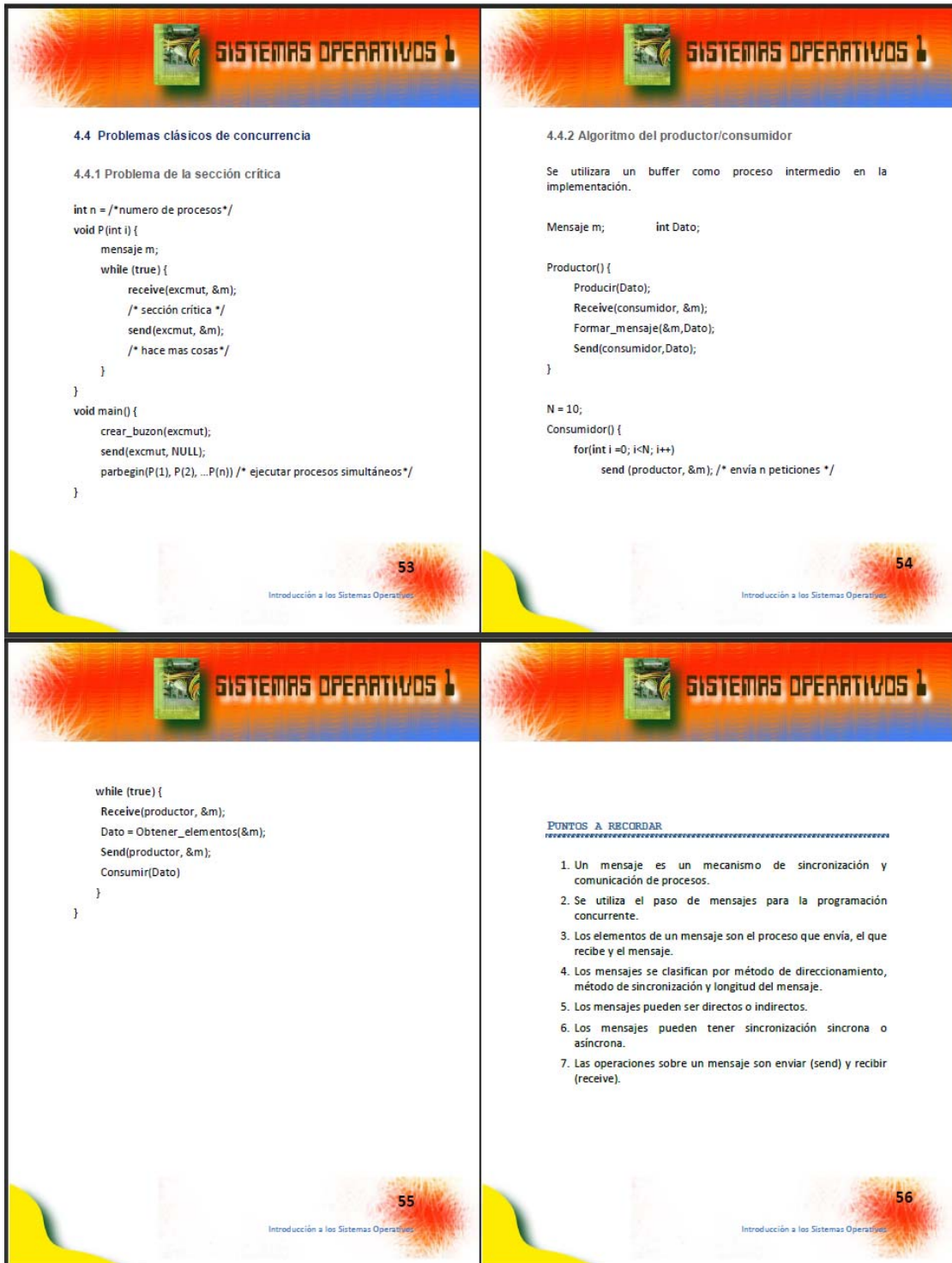


Figura 18 – Páginas 57-60 Libro “Introducción a los Sistemas Operativos”

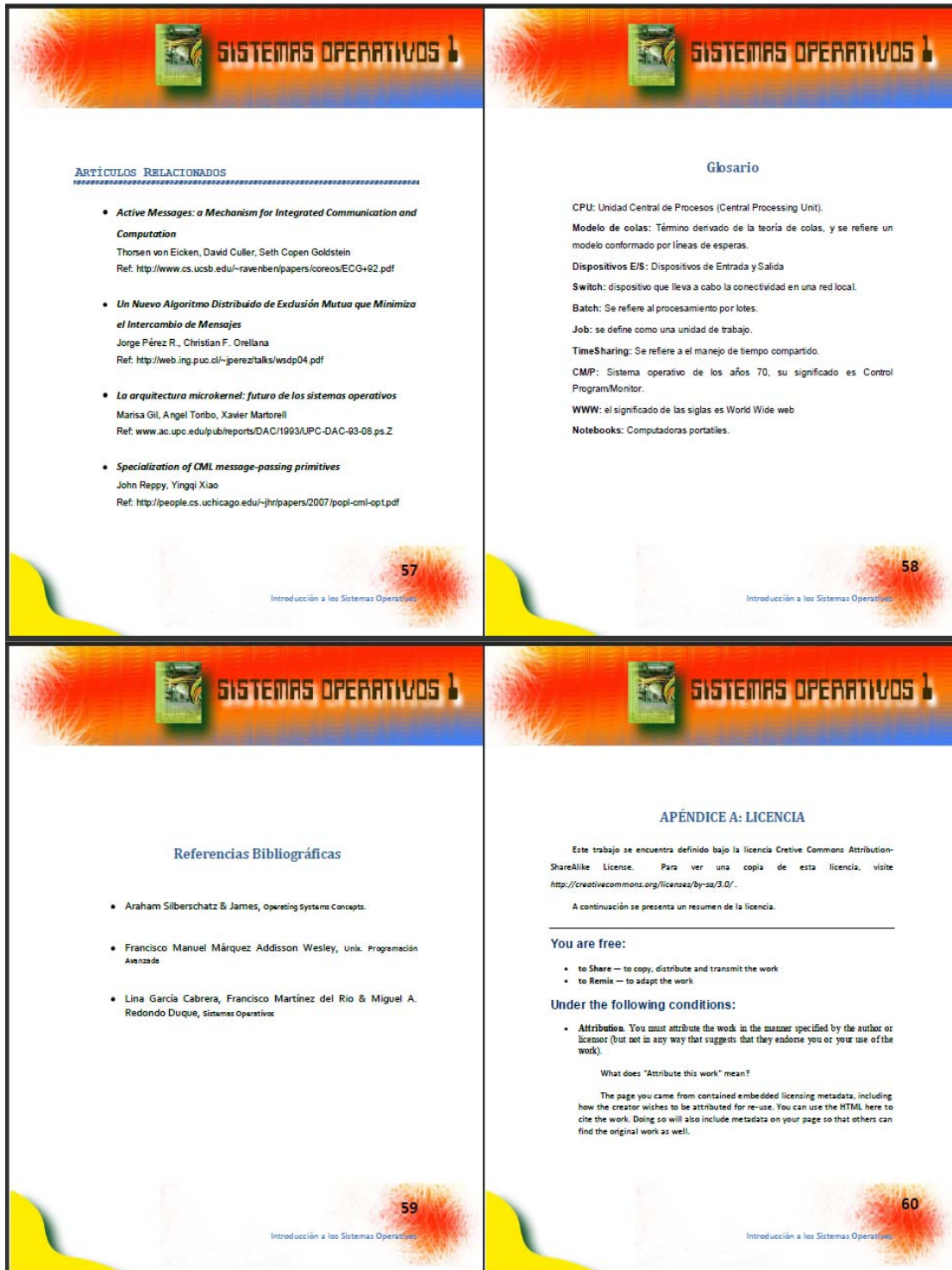


Figura 19 – Páginas 61-64 Libro “Introducción a los Sistemas Operativos”

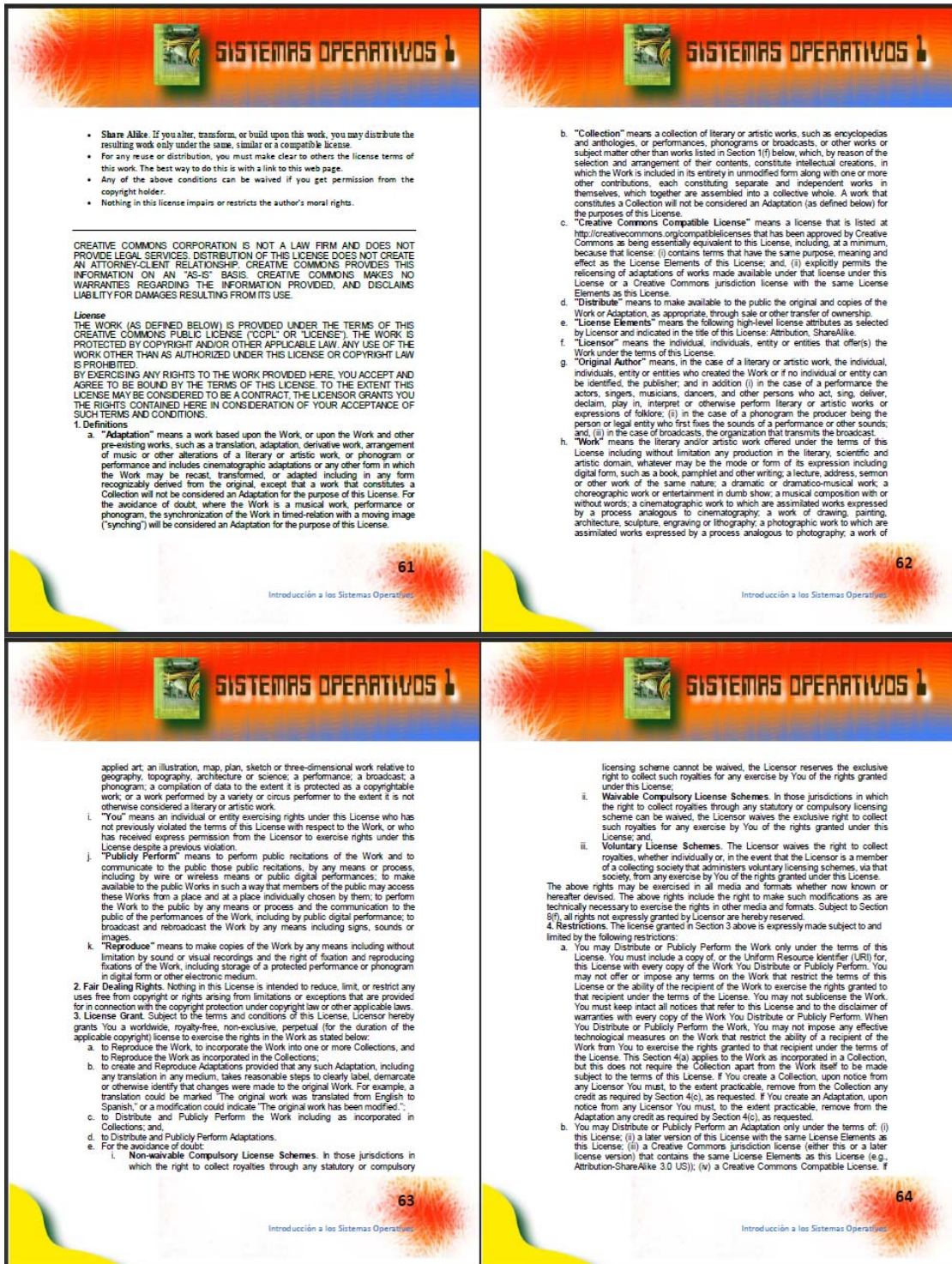
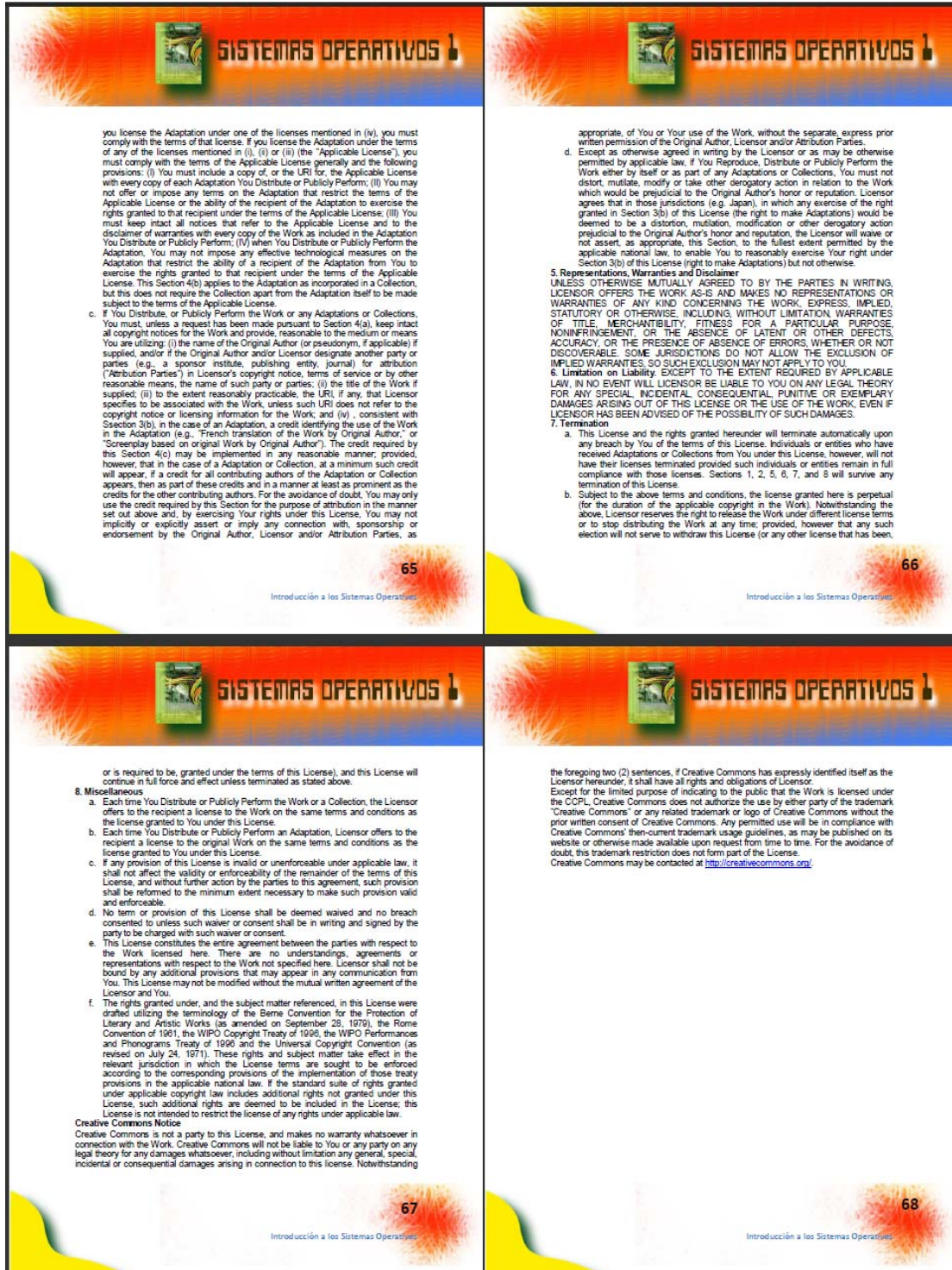


Figura 20 – Páginas 65-68 Libro “Introducción a los Sistemas Operativos”



7. DOCUMENTACIÓN DE APOYO DE “SISTEMAS OPERATIVOS 2”

A continuación se presenta el desarrollo de la documentación de apoyo del curso de Sistemas Operativos 1, el cual se realizó en formato de un libro conteniendo las unidades más importantes impartidas a lo largo del curso.

Datos generales

NOMBRE DE CURSO: Sistemas Operativos 2 (285)

PRE- REQUISITOS: Sistemas Operativos 1 (281)

POST – REQUISITOS:

OBJETIVO: Este libro esta destinado para los estudiantes del curso de Sistemas Operativos 2, contiendo teoría, ejemplos, diagramas, resumen, glosario y artículos de interés relacionados a cada una de las unidades del curso.

MÓDULOS: Introducción a los Sistemas Operativos

Figura 21 – Páginas 1-4 Libro “Sistemas Operativos Avanzados”



Figura 22 – Páginas 5-8 Libro “Sistemas Operativos Avanzados”

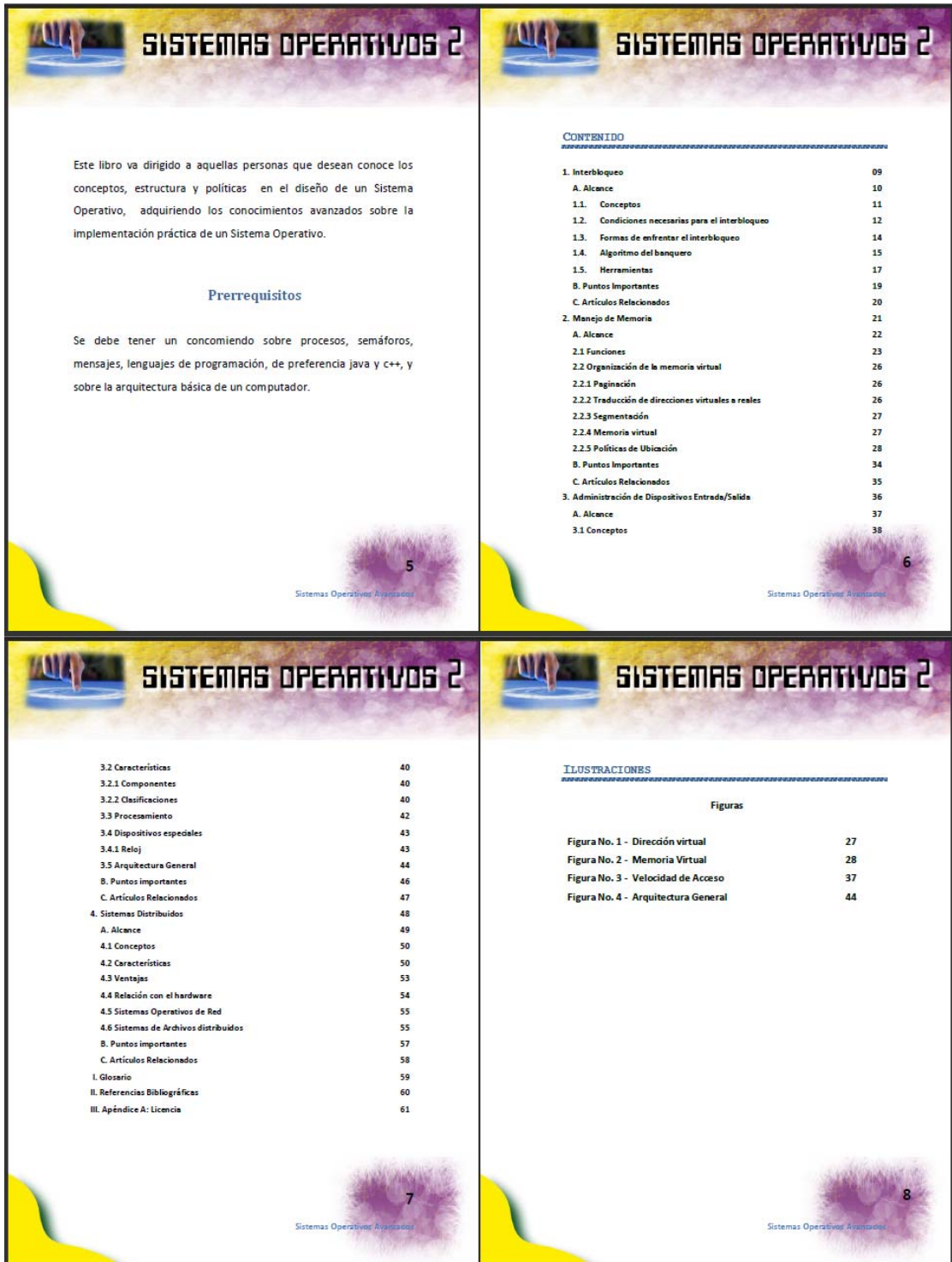


Figura 23 – Páginas 9-12 Libro “Sistemas Operativos Avanzados”

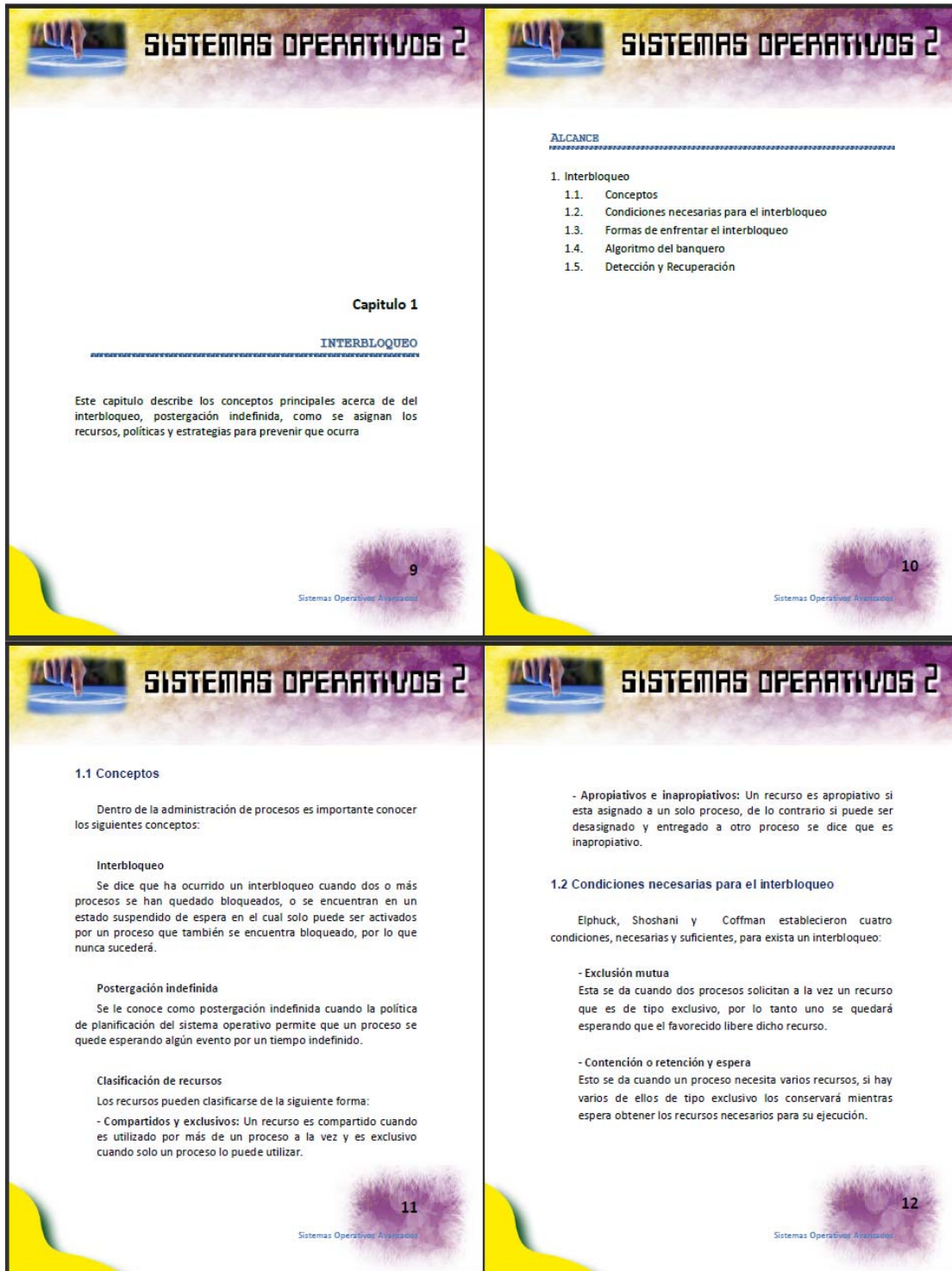


Figura 24 – Páginas 13-16 Libro “Sistemas Operativos Avanzados”

SISTEMAS OPERATIVOS 2

- Inapropiatiividad
Esto se da debido a los recursos apropiativos, en la que solo el proceso puede liberar los recursos que le han sido asignados, sin poder el sistema desasignárselo si no los esta utilizando porque esta en espera y dárselos a otro proceso.

- Espera circular
Suponiendo que un proceso PA esta suspendido, esperando algún recurso exclusivo que tiene un proceso PB, entonces se dice que PA depende de PB (PA <= PB). De la misma forma se puede ir creando una cadena circular de dependencias entre procesos PA <= PB <= PC <= ... <= PZ <= PA, que mantendrá a todos en espera.

1.3 Políticas de tratamiento del interbloqueo

Para poder tratar con el interbloqueo el sistema operativo debe implementar ciertas políticas y estrategias, entre las que se encuentran:

Indiferencia: El sistema operativo no se hace responsable y deja que el usuario y el programador sean quienes eviten que se de.

13

Sistemas Operativos Avanzados

SISTEMAS OPERATIVOS 2

Prevención: Dicen que es mejor prevenir que lamentar, con esta estrategia se trata de colocar en el sistema una serie de reglas de forma estática que lo condicionen de tal manera que aunque sea una de las condiciones del interbloqueo no se de y así poder evitarlo.

Evitación o predicción: En esta estrategia se deja que las condiciones para el interbloqueo se den, pero una vez son identificadas y existe el riesgo que el interbloqueo se de denegar la asignación del recurso que puede generar el interbloqueo.

Detección y recuperación: En esta política, se deja que el interbloqueo ocurra, pero se deben implementar procesos que se encarguen de revisar el estado de asignación de los procesos para detectar los bloqueados, y aplicar políticas de recuperación, como matar el proceso, para resolverlo.

1.4 Algoritmo del Banquero

Algoritmo desarrollado por Dijkstra y Habermann para evitar el interbloqueo, funciona de la siguiente forma:

1. Necesita que cada proceso indique el número máximo de recursos que va a utilizar.

14

Sistemas Operativos Avanzados

SISTEMAS OPERATIVOS 2

2. En cada requerimiento de un recurso se determina si el asignar los recursos que se han pedido deja un estado inseguro de asignación de recursos, entonces se pospone el requerimiento, de lo contrario se asignan los recursos que se solicitaron.

```

Estructuras de datos:
Siendo m el número de tipos de recursos y n el número de procesos
int disponible[m]; // unidades de recursos disponibles
int ma[m]; // número máximo de requerimientos del proceso P del recurso Rj
int asignado[m][n]; // asignación actual en el proceso P del recurso Rj
int recurso[m]; // recurso Rj
int requerimiento[m][n]; // vector para el requerimiento de cada petición de recursos

void requerimiento_de_recursos (int requerimiento[], int proceso) {
    if (requerimiento >= recurso[m][proceso])
        error (); // máximo de recursos excedidos a peticion
    if (requerimiento <= disponible)
        suspender (proceso); // recursos que no estan no disponibles
    disponible -= requerimiento;
    asignado[proceso] += requerimiento;
    recurso[proceso] -= requerimiento;
    if (estado_seguro []) // no hay quejas al estado anterior
        disponible += requerimiento;
    asignado[proceso] -= requerimiento;
    recurso[proceso] += requerimiento;
    suspender (proceso);
}

int es_estado_seguro () {
    int disp_temp = disponible;
    bool terminado[n] = {FALSE, ..., FALSE};
    int i;
    while (terminado == TRUE) {
        for (i=0; i < n; i++) {
            if (terminado[i] == TRUE)
                continue;
            if (disp_temp <= ma[i][i])
                terminado[i] = TRUE;
            disp_temp -= ma[i][i];
        }
    }
    return TRUE;
}
                
```

16

Sistemas Operativos Avanzados

SISTEMAS OPERATIVOS 2

Desventajas

Entre las desventajas de este algoritmo están:

- La necesidad de saber el número máximo de recursos que utilizará cada proceso desde el inicio.
- Se genera un retraso cada vez que se asigna un recurso, lo que puede ser considerablemente problemático si se trabajan con muchos recursos y/o procesos.
- Se necesita una garantía en la devolución, al momento de que un proceso devuelva sus recursos, o podría generar que un proceso no termine y por consiguiente los demás tampoco.

1.5 Herramientas para el interbloqueo en lo Sistemas Operativos existentes

CAT (Crash Análisis Tool)

Esta herramienta es utilizada por Solaris y sirve para analizar los archivos centrales del sistema, es decir que analizan las salidas o errores que se dan generando archivos de reporte describiendo las causas que ocasionaron el estado actual del sistema.

16

Sistemas Operativos Avanzados

Figura 25 – Páginas 17-20 Libro “Sistemas Operativos Avanzados”

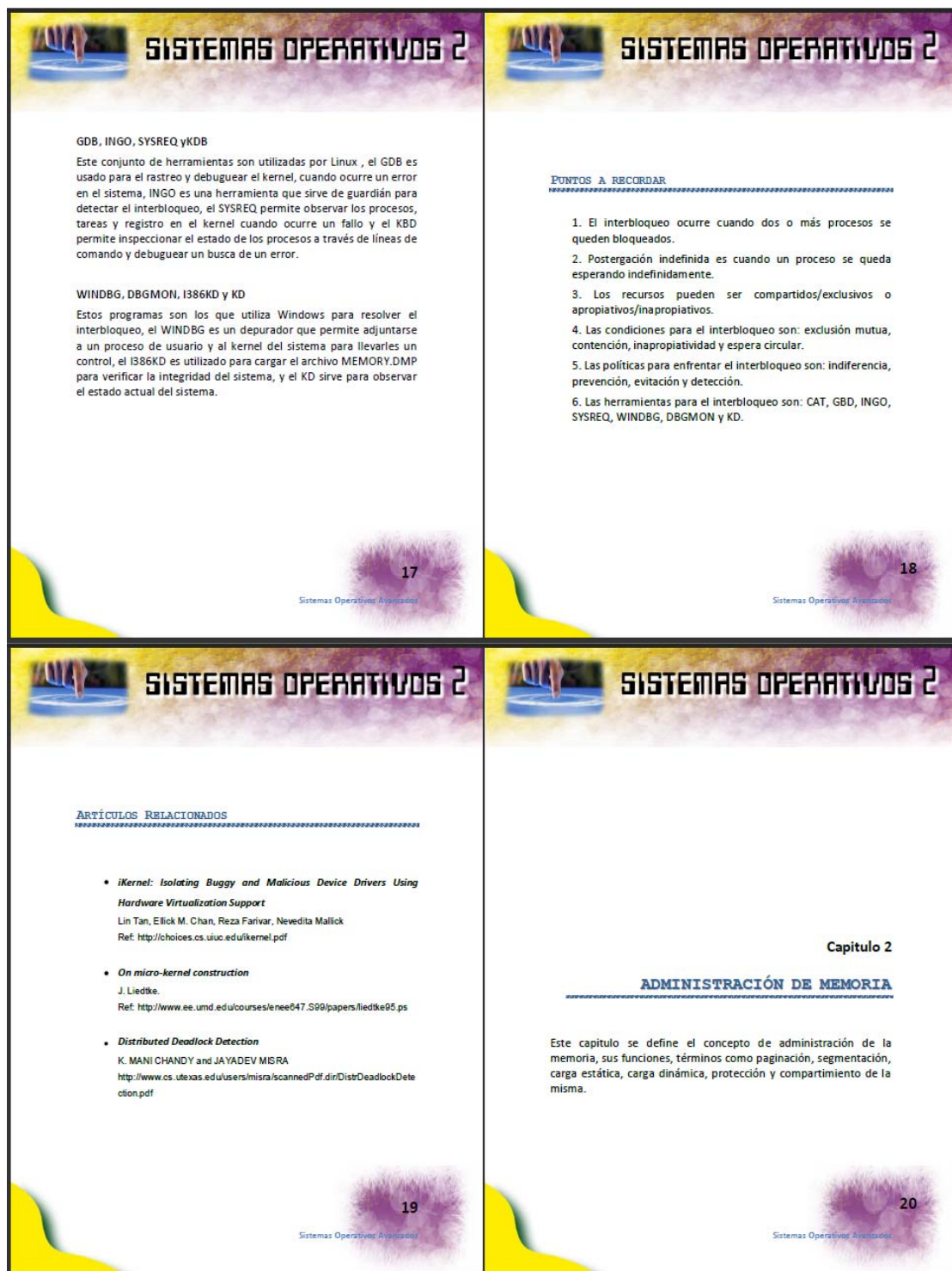


Figura 26 – Páginas 21-23 Libro “Sistemas Operativos Avanzados”

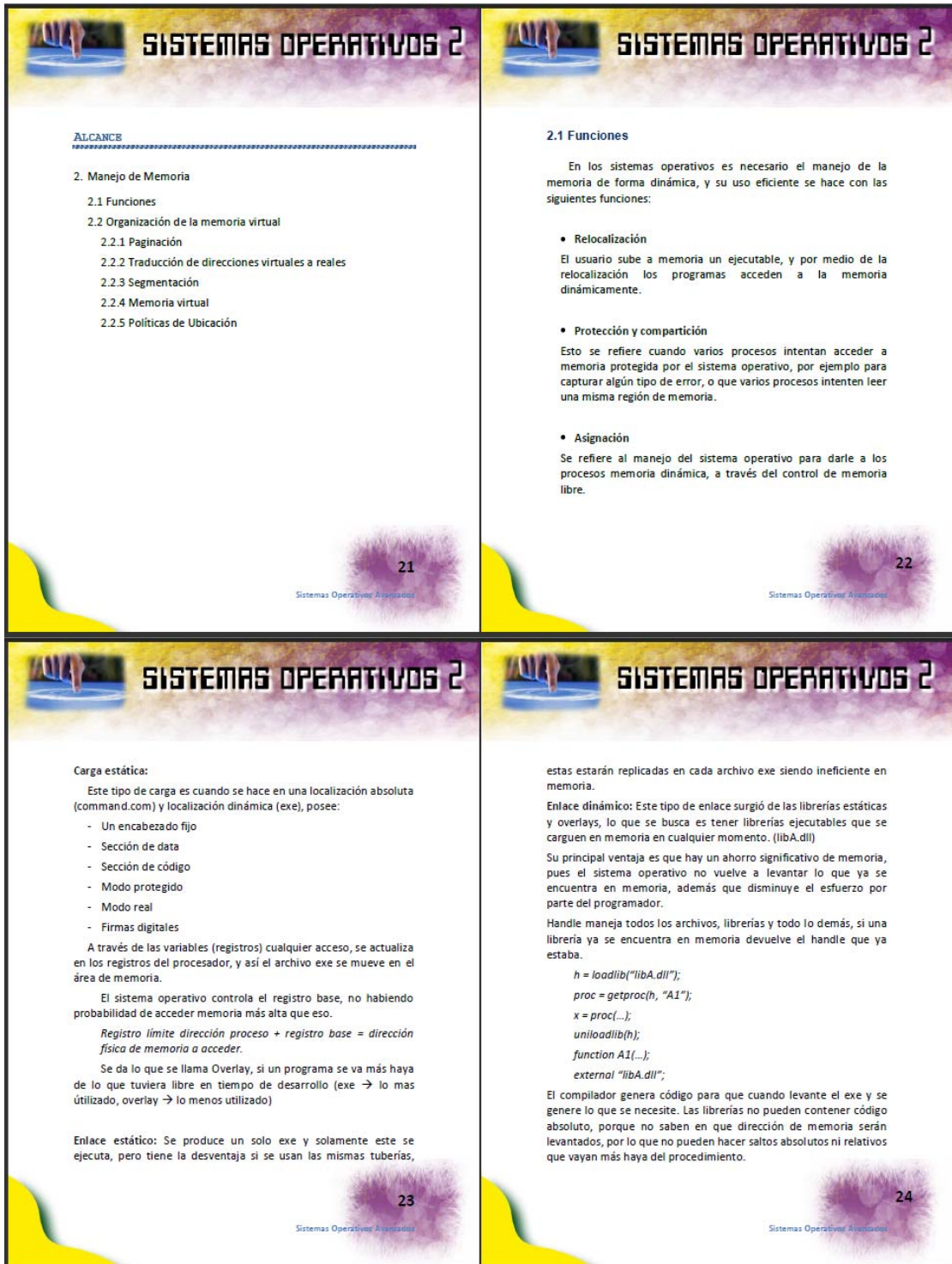


Figura 27 – Páginas 25-28 Libro “Sistemas Operativos Avanzados”

SISTEMAS OPERATIVOS 2

2.2 Organización de la memoria virtual

2.2.1 Paginación

- Espacios de dirección virtual
- Las páginas son de tamaño fijo
- Se usan marcos de página
- Se da fragmentación interna

2.2.2 Traducción de direcciones virtuales a reales

Figura No. 1 - Dirección virtual

25

Sistemas Operativos Avanzados

SISTEMAS OPERATIVOS 2

2.2.3 Segmentación

- Punto de vista del usuario: la memoria es un conjunto de segmentos, tales como datos, código, pila, etc.
- Se realiza un mapeo similar a la paginación
- La tabla de segmentos tiene que incluir el tamaño de cada segmento.
- Tiene relación con las políticas de asignación de memoria por particiones variables
- Se da fragmentación externa

2.2.4 Memoria virtual

Se debe hacer una jerarquía en el uso de la memoria

Figura No. 2 - Memoria Virtual

Quando se habla de memoria virtual, se refiere a utilizar un espacio en disco duro (partición o archivo del sistema), que se utilizará para almacenar porciones ó paginas, para de esa forma expandir la memoria real.

26

Sistemas Operativos Avanzados

SISTEMAS OPERATIVOS 2

Los segmentos tienen la información de su tamaño, se llenan los marcos de páginas, entonces se debe esperar a que el marco de página sea liberado.

Swapping: Se refiere a cuando se toma una página y se baja su información a disco.

Fallas de página: Se refiere al acceso de una página que no se encuentra en memoria, sino que en el disco.

Hiperpaginación: Exceso de fallos de página.

Desperdicio (trashing): Uso excesivo de memoria virtual.

2.2.5 Políticas de Ubicación

Estrategias de reposición

Se ocupa del aspecto de qué página se debe reemplazar cuando se produce una falla página

27

Sistemas Operativos Avanzados

SISTEMAS OPERATIVOS 2

Principio de optimización

Debe reemplazarse la página que no se va a utilizar durante el tiempo más largo, es decir el que tiene la menor tasa de fallos de páginas de todos los algoritmos.

Objetivos

- * Minimizar la hiperpaginación (trashing o desperdicio) del procesador

Al azar

- * Simple
- * ineficiente

PEPS

- * Se reemplaza la primera más vieja en memoria
- * Desfavorecer a páginas de uso constante
- * Anomalia PEPS

28

Sistemas Operativos Avanzados

Figura 28 – Páginas 29-32 Libro “Sistemas Operativos Avanzados”

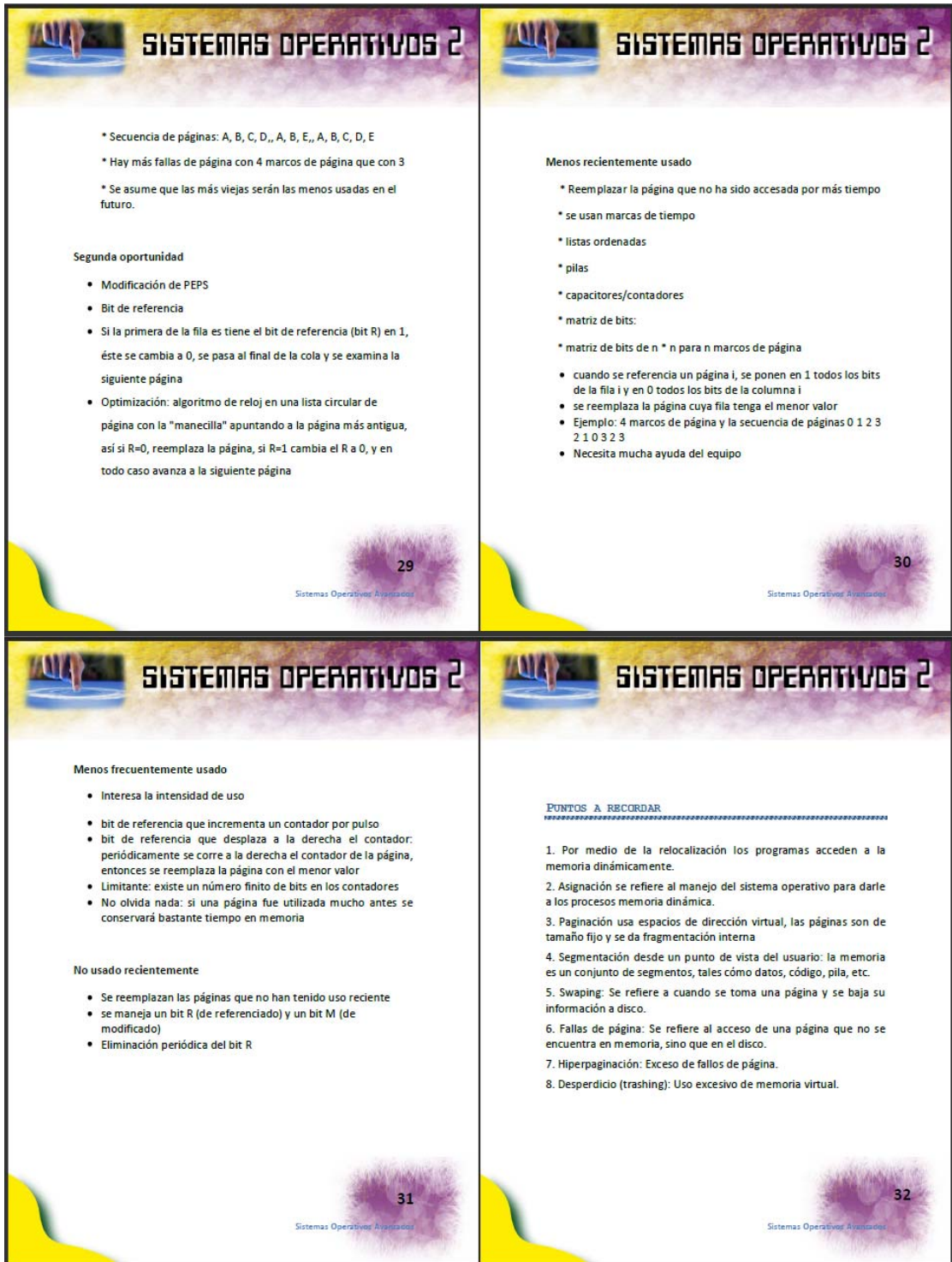


Figura 29 – Páginas 33-36 Libro “Sistemas Operativos Avanzados”

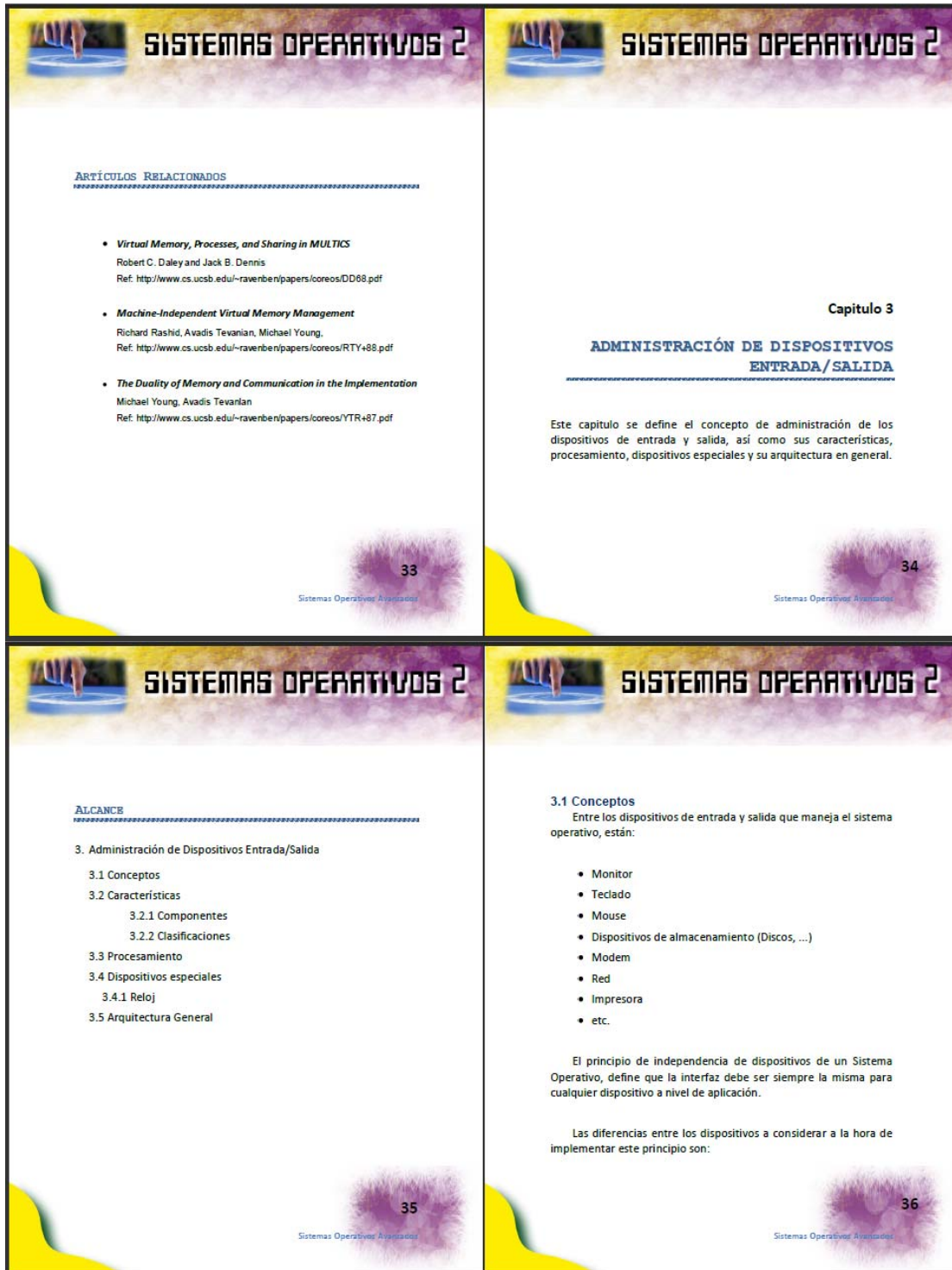


Figura 30 – Páginas 37-40 Libro “Sistemas Operativos Avanzados”

SISTEMAS OPERATIVOS 2

- Velocidad de datos
- Aplicaciones
- Unidad de transferencia (por bloques/por caracteres)
- Complejidad del control
- Codificación en la representación de los datos
- Condiciones de error

La velocidad de los dispositivos se mide en el orden de milisegundos, y el deseo de su ejecución interactiva genera cuellos de botella.

Figura 3 - Velocidad de acceso

37

Sistemas Operativos Avanzados

SISTEMAS OPERATIVOS 2

Por lo tanto los objetivos de un Sistema Operativo respecto a los dispositivos de E/S deben ser:

- Proporcionar una interfaz sencilla y fácil de usar
- Optimizar la ejecución
- Evitar el rearmar el Sistema E/S o el Sistema Operativo al conectar un nuevo dispositivo.

3.2 Características

3.2.1 Componentes

Un dispositivo consta de tres componentes principales:

- **Periférico:** Es el componente mecánico que forma el dispositivo en sí.
- **Consolador:** Se refiere al componente electrónico que funciona como un adaptador que realiza la interacción entre el dispositivo y el procesador por medio del bus de datos.
- **Driver:** Se refiere a la componente lógico, es decir el software que realiza la conversión entre los comandos propios del sistema operativo y los adapta a instrucciones específicas para el dispositivo.

38

Sistemas Operativos Avanzados

SISTEMAS OPERATIVOS 2

3.2.2 Clasificaciones

Los dispositivos se pueden clasificar de acuerdo a la función que tienen:

- Dispositivos de interfaz de usuarios
- Dispositivos de almacenamiento
- Dispositivos de comunicaciones

Según la transferencia de información los dispositivos se clasifican en:

- **Dispositivos de bloques**
Este tipo de dispositivos reciben y entregan la información en forma de bloques de tamaño fijo, en los cuales se puede leer/escribir de manera independiente a través de sus propiedades de localización.
- **Dispositivos de caracteres**
Este tipo de dispositivos reciben y entregan la información en un flujo de caracteres que no son direccionables, por lo tanto no poseen ninguna función de localización.

39

Sistemas Operativos Avanzados

SISTEMAS OPERATIVOS 2

Según su direccionamiento los dispositivos se clasifican en:

- **Dispositivos proyectados en memoria**
Este tipo de dispositivo asigna un rango de direcciones a cada controlador por medio de las cuales se puede acceder a sus registros, programando instrucciones de acceso a memoria. Reserva en memoria física una zona específica para el manejo de los controladores.
- **Dispositivos proyectados en puertos**
Este tipo de dispositivo asigna a cada controlador de E/S un puerto específico, en el que se indica que registro se debe manipular y la forma de hacerlo.

3.3 Procesamiento

La interacción que existe entre los controladores de cada dispositivo y el procesador se de las formas:

40

Sistemas Operativos Avanzados

Figura 31 – Páginas 41-44 Libro “Sistemas Operativos Avanzados”

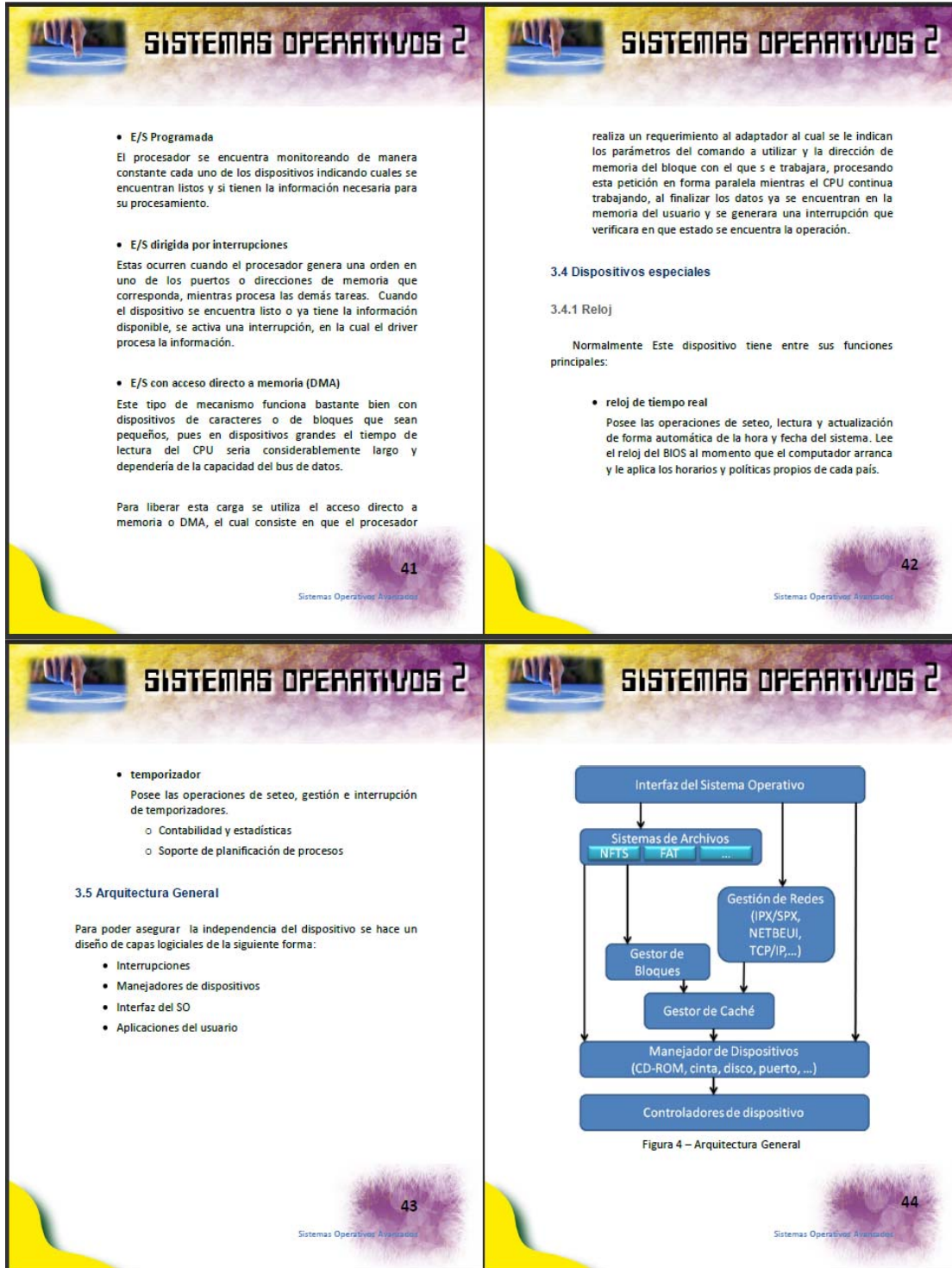


Figura 32 – Páginas 45-48 Libro “Sistemas Operativos Avanzados”

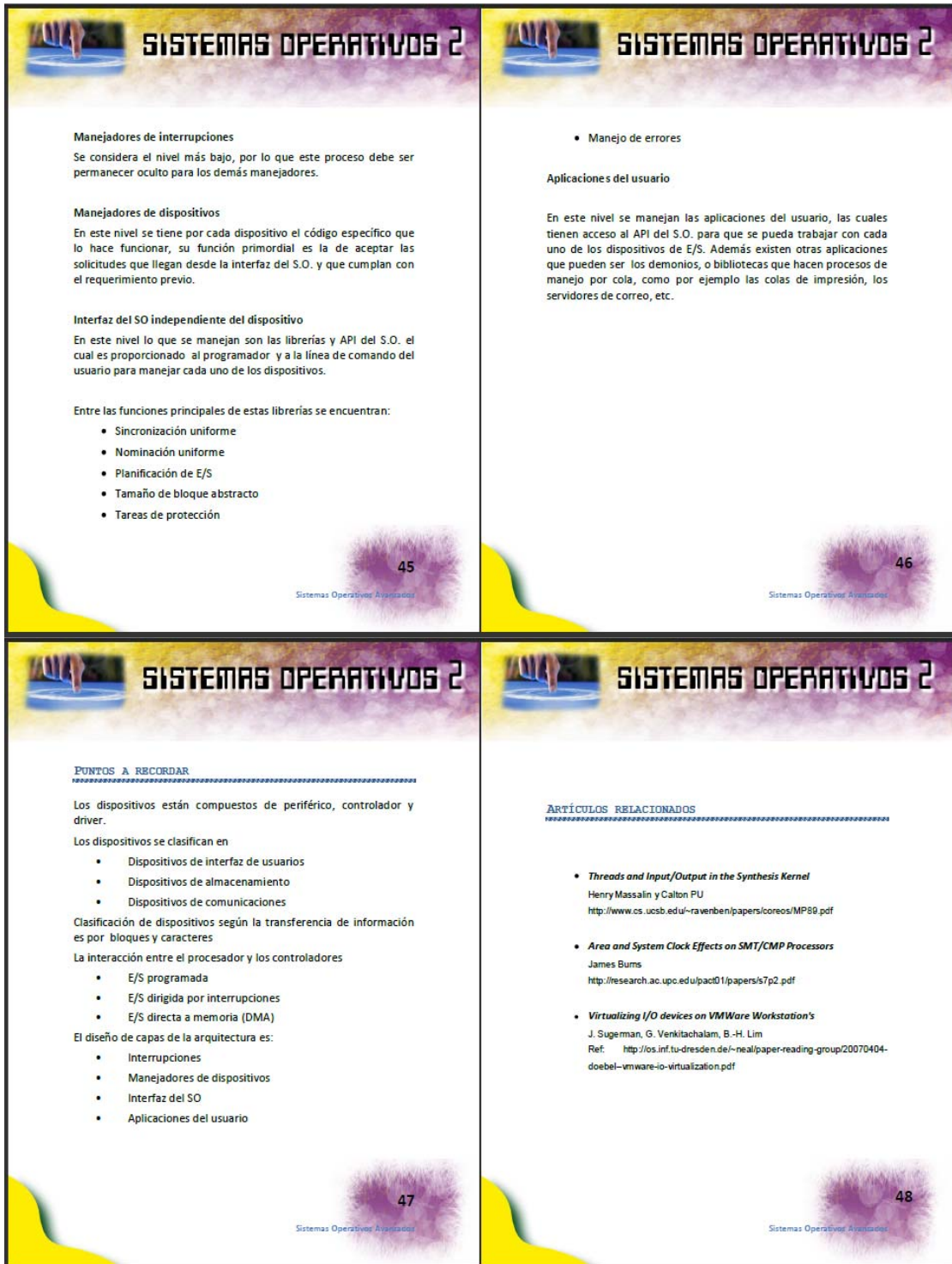


Figura 33 – Páginas 49-52 Libro “Sistemas Operativos Avanzados”

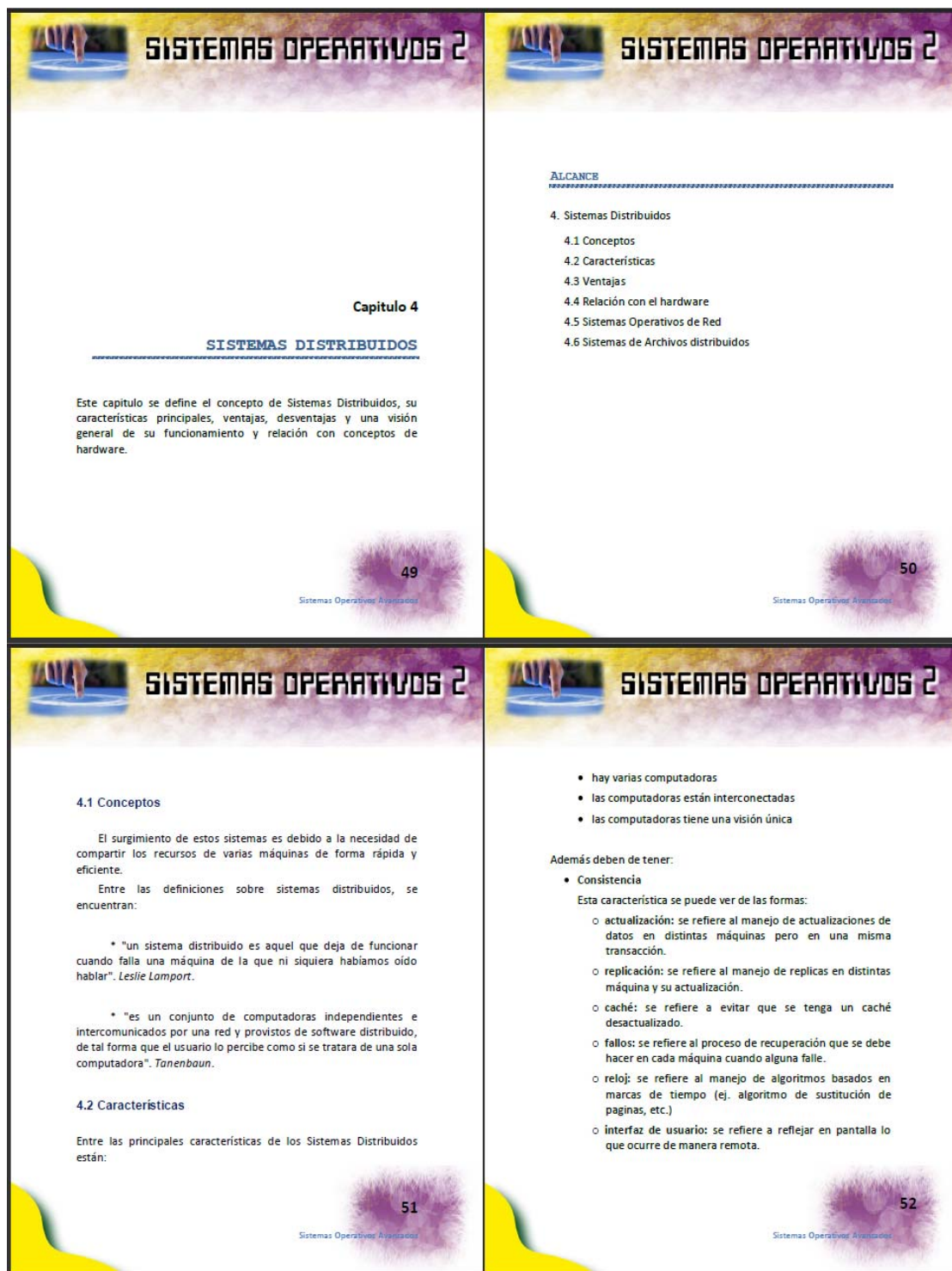


Figura 34 – Páginas 53-56 Libro “Sistemas Operativos Avanzados”

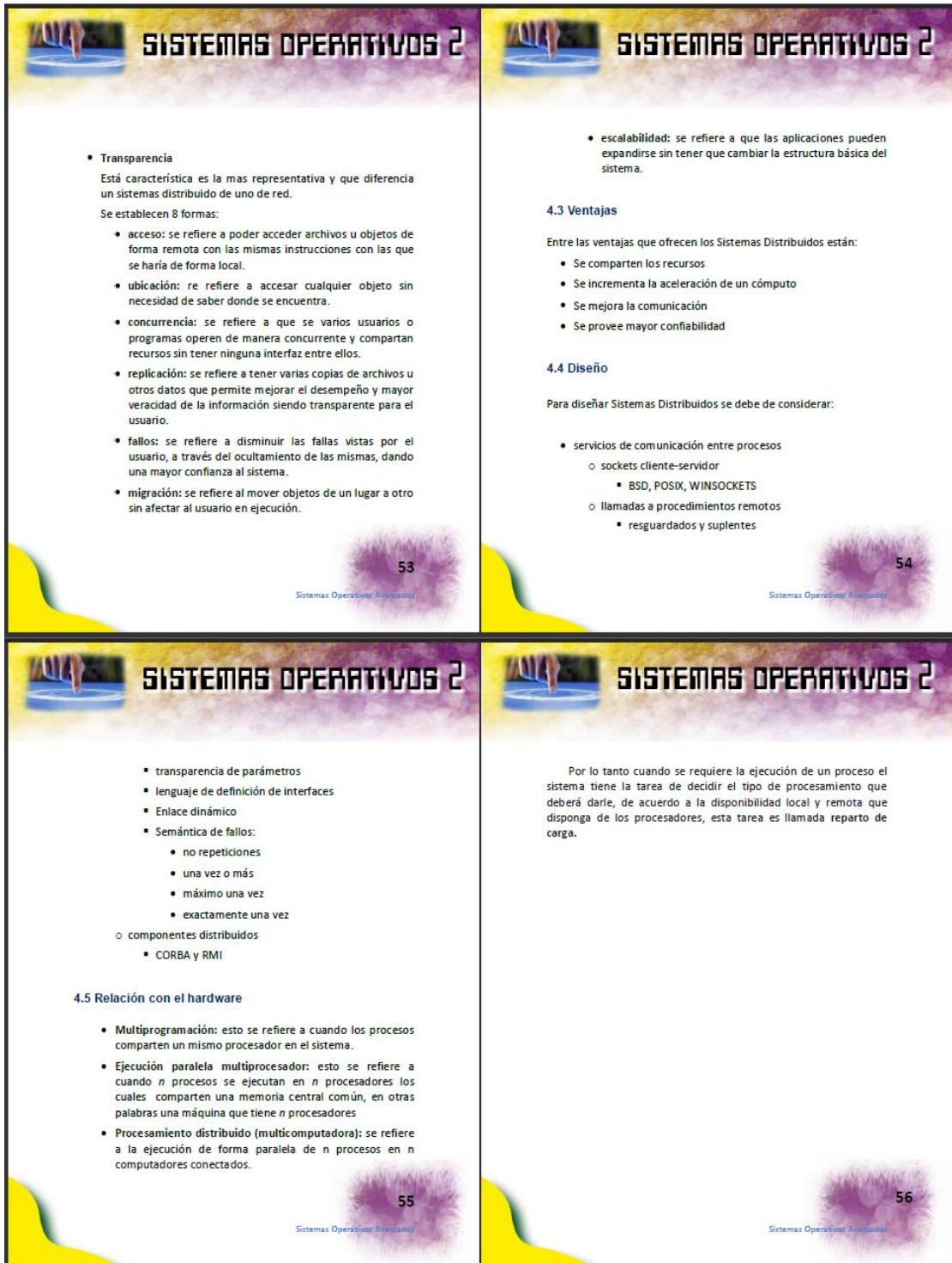


Figura 35 – Páginas 57-60 Libro “Sistemas Operativos Avanzados”

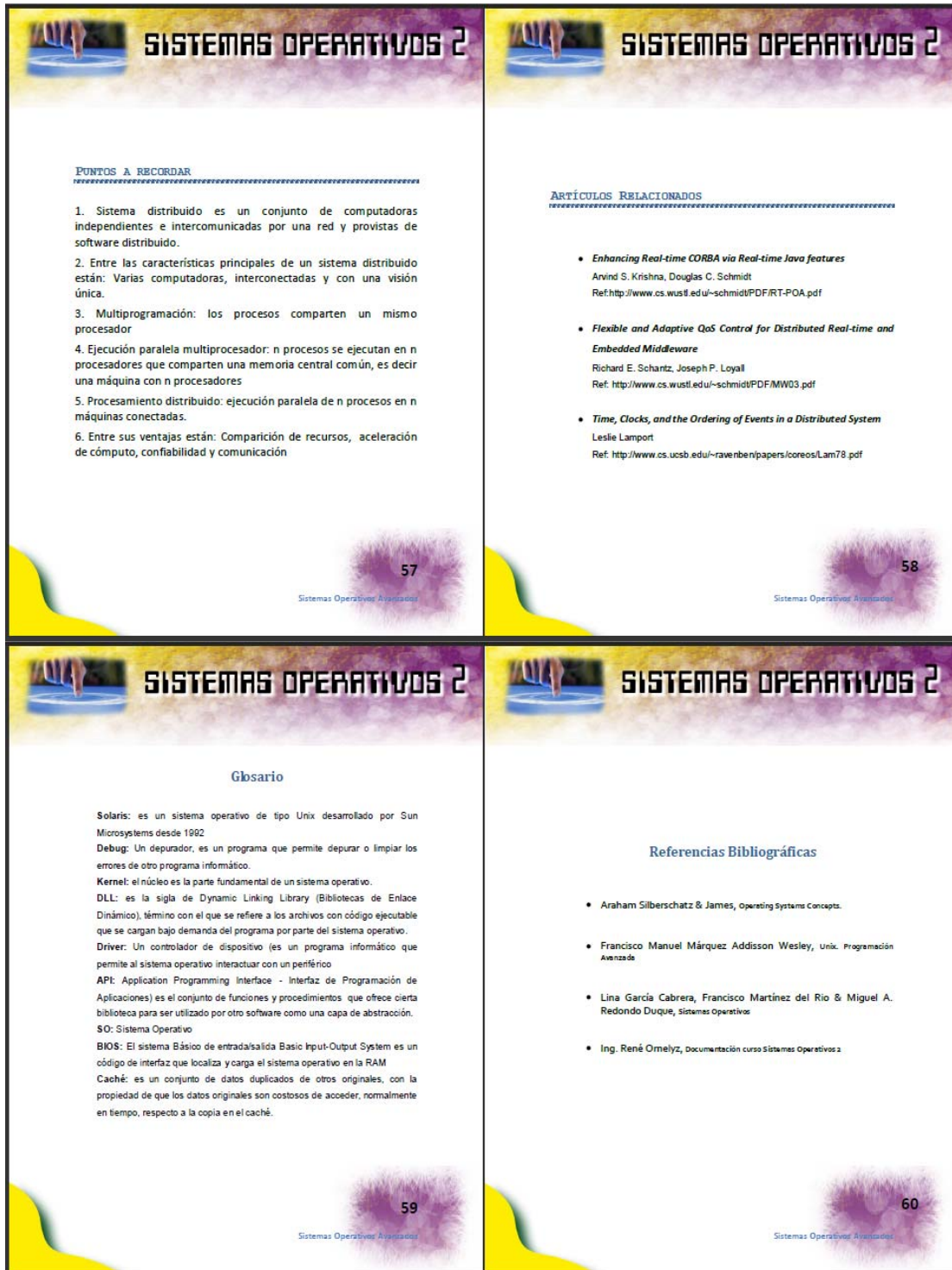


Figura 36 – Páginas 61-64 Libro “Sistemas Operativos Avanzados”

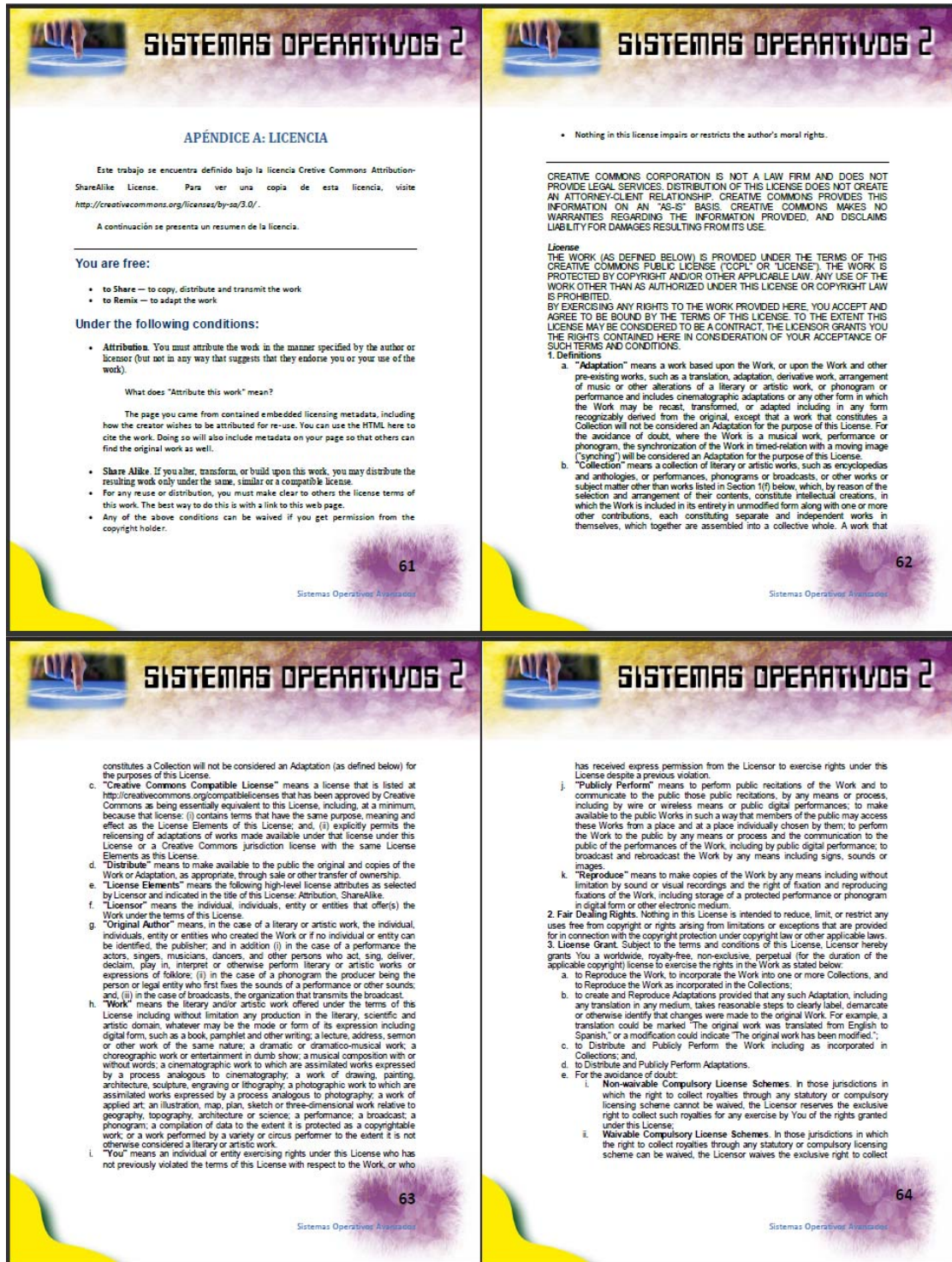


Figura 37 – Páginas 65-68 Libro “Sistemas Operativos Avanzados”

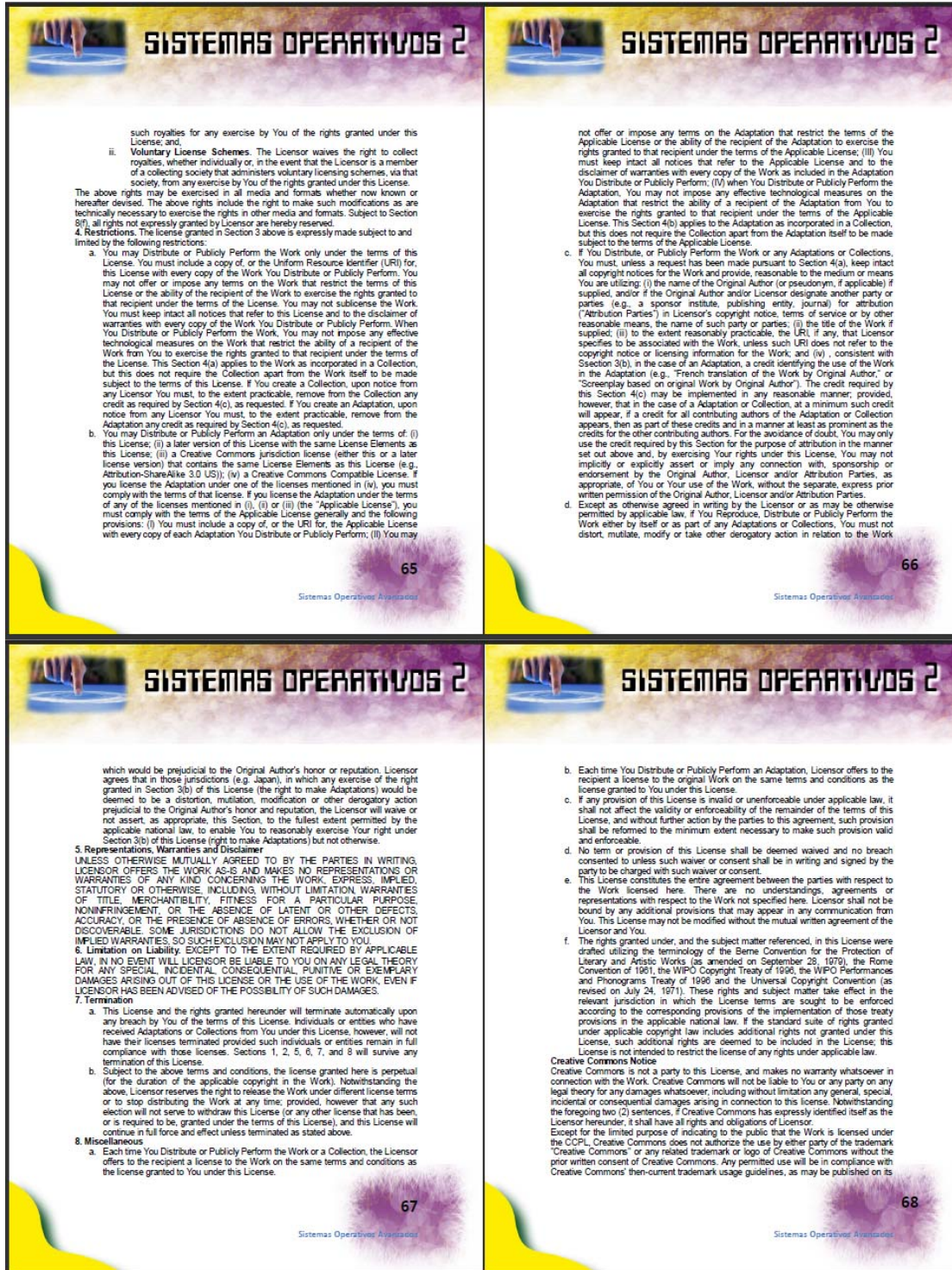
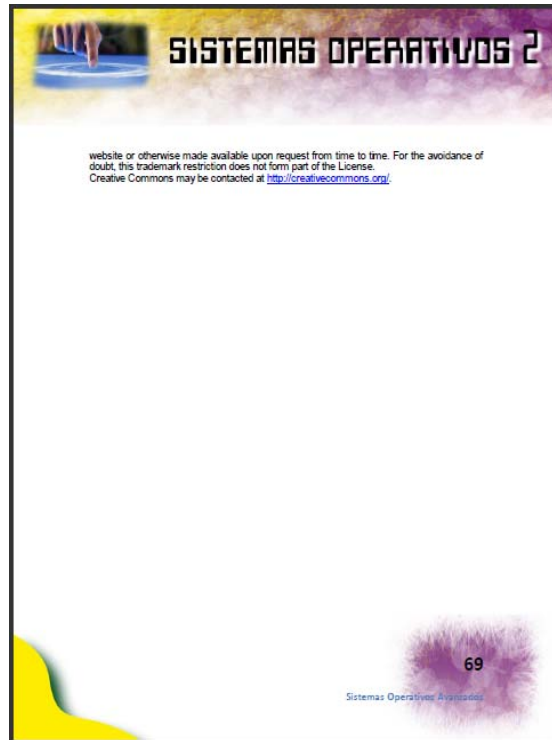


Figura 38 – Página 69 Libro “Sistemas Operativos Avanzados”



8. DOCUMENTACIÓN DE APOYO DE “MANEJO E IMPLEMENTACIÓN DE ARCHIVOS”

A continuación se presenta el desarrollo de la documentación de apoyo del curso de Manejo e Implementación de Archivos, el cual se realizó en formato de un libro conteniendo las unidades más importantes impartidas a lo largo del curso.

Datos generales

NOMBRE DE CURSO: **Manejo e Implementación de Archivos (773)**

PRE- REQUISITOS: Estructuras de Datos (772)
Lenguajes Formales y de Programación (796)

POST – REQUISITOS: Sistemas de Bases de Datos I (774)
Redes de Computadoras I (970)

OBJETIVO: Este libro está destinado para los estudiantes del curso de Manejo e Implementación de Archivos contiendo teoría, ejemplos, diagramas, resumen, glosario y artículos de interés relacionados a cada una de las unidades del curso.

MÓDULOS: Manejo e Implementación de Archivos

Figura 39 – Páginas 1-4 Libro “Manejo e Implementación de Archivos”

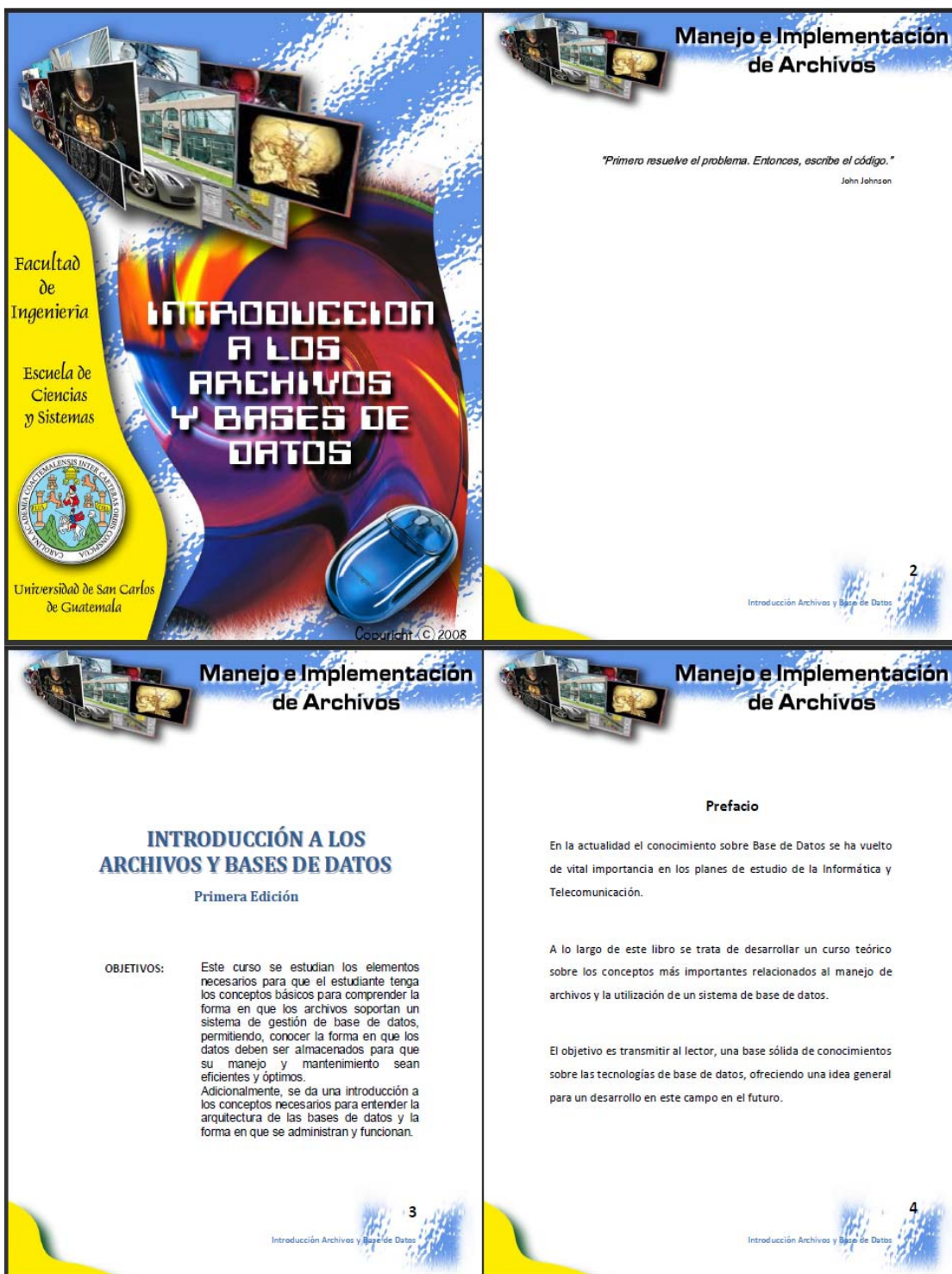


Figura 40 – Páginas 5-8 Libro “Manejo e Implementación de Archivos”

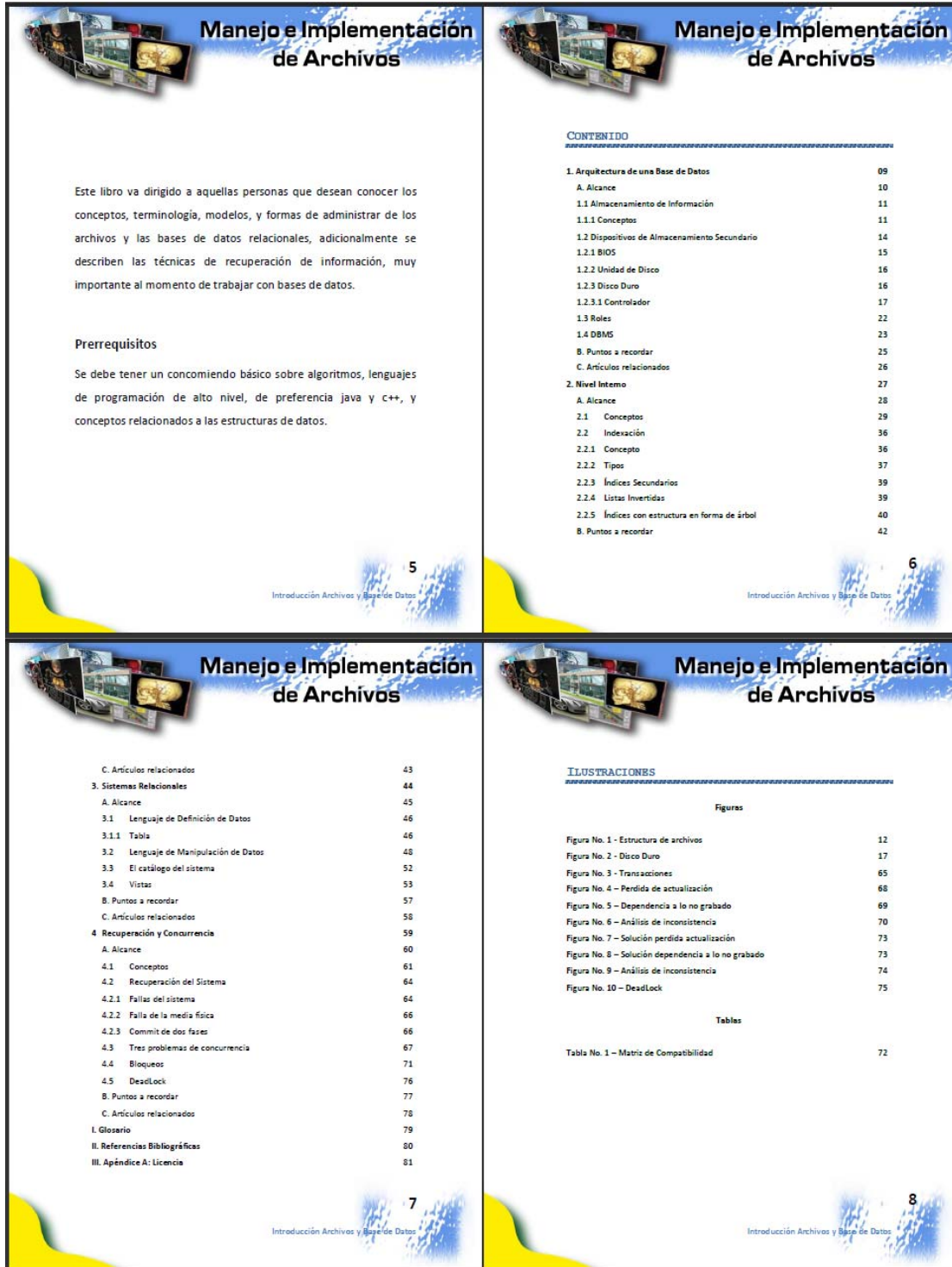


Figura 41 – Páginas 9-12 Libro “Manejo e Implementación de Archivos”

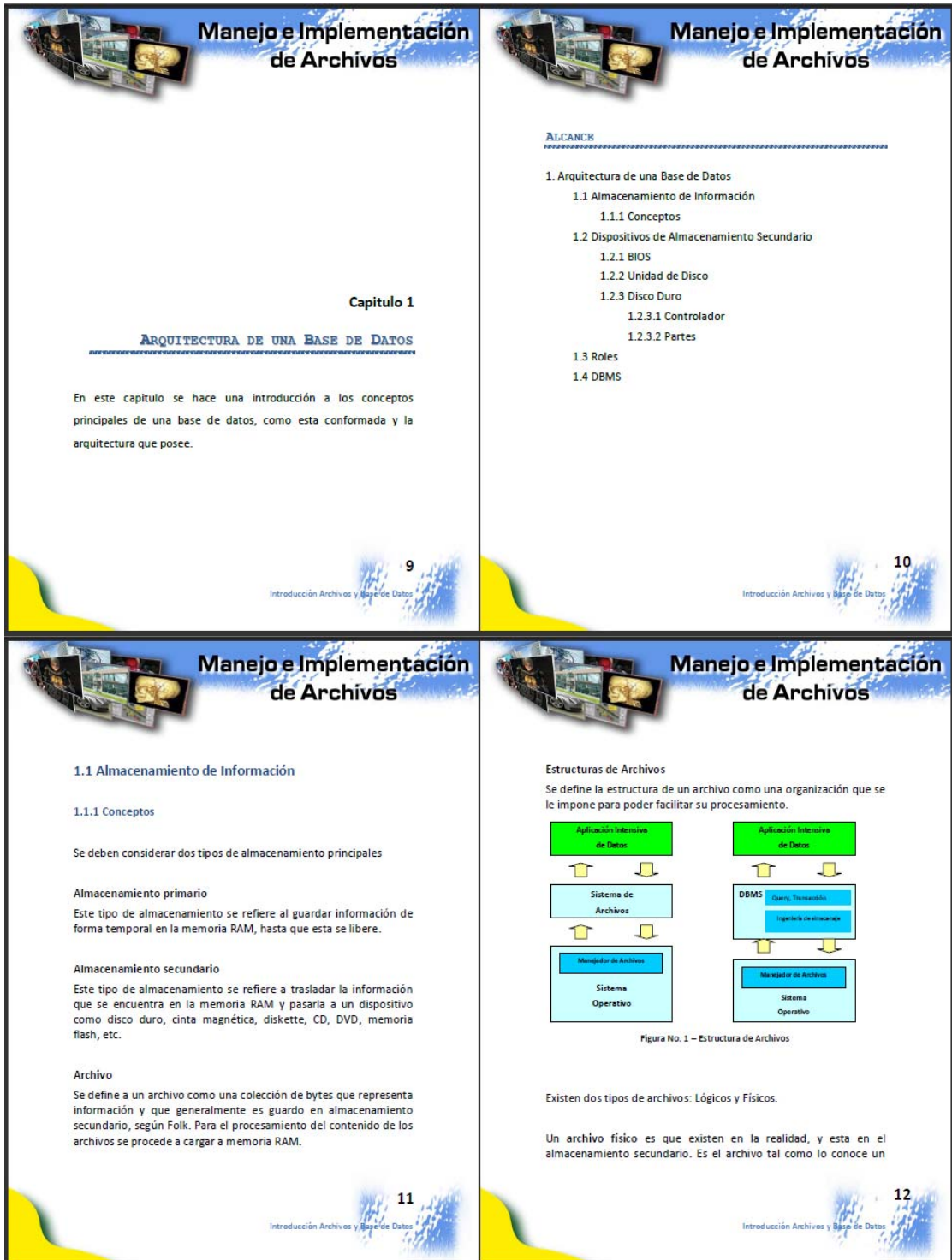


Figura 42 – Páginas 13-16 Libro “Manejo e Implementación de Archivos”

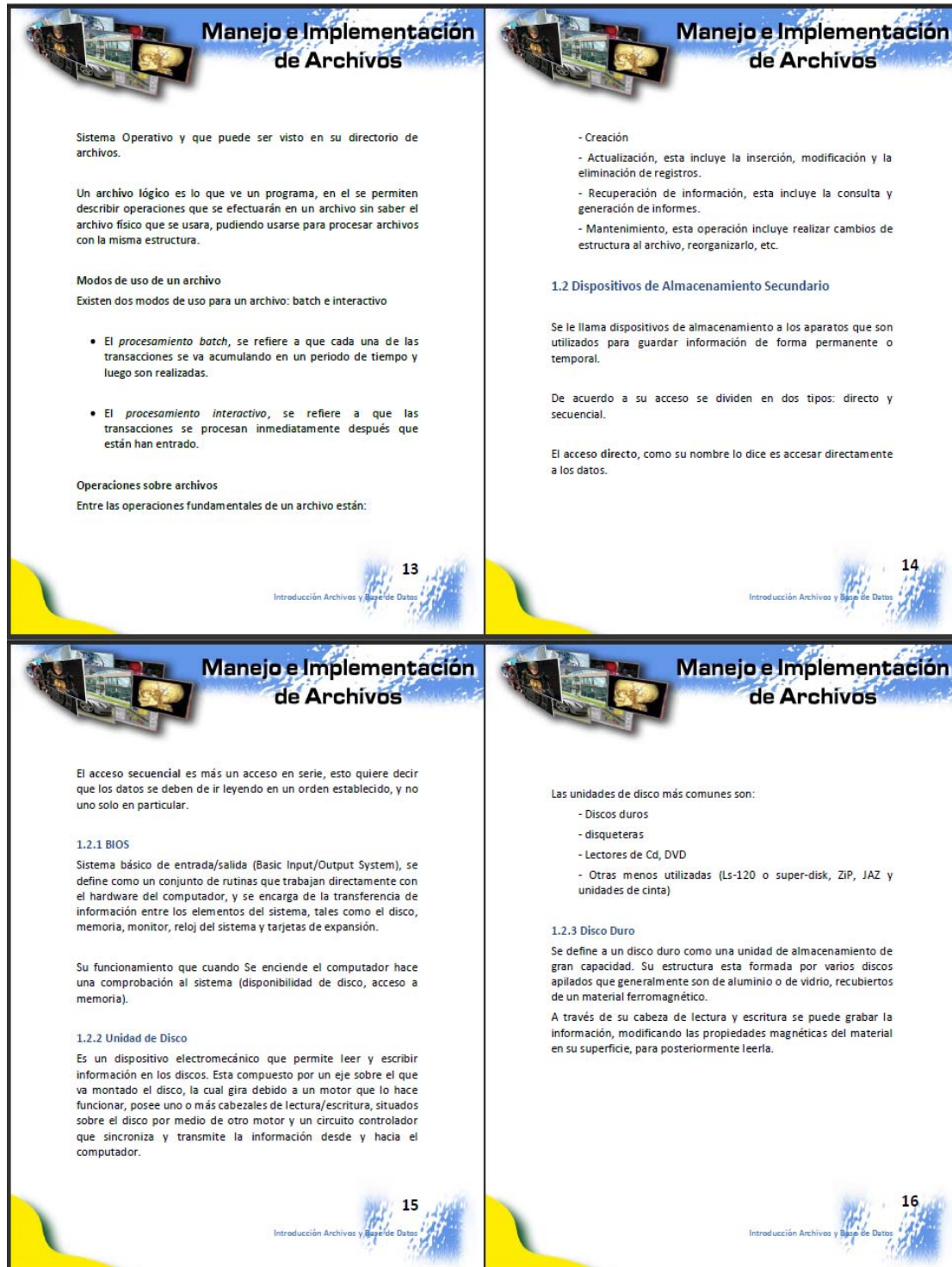


Figura 43 – Páginas 17-20 Libro “Manejo e Implementación de Archivos”

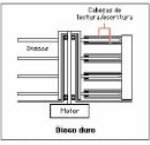
<p style="text-align: center;">Manejo e Implementación de Archivos</p>  <p style="text-align: center;">Figura No. 2 – Disco Duro</p> <p>1.2.3.1 Controlador</p> <p>El controlador de disco se define como un conjunto de circuitos los cuales se encuentran integrados en chip o tarjeta, la cual se encarga de dirigir la lectura y escritura de las unidades de disco del computador.</p> <p>Sus funciones son situar la cabeza de lectura/escritura, actuar como un intermediario entre la unidad de disco y el microprocesador y controlar la transferencia de la información desde y hacia la memoria RAM.</p> <p style="text-align: right;">17</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>	<p style="text-align: center;">Manejo e Implementación de Archivos</p> <p>Existen varios tipos:</p> <p>IDE o EIDE --> Es una tarjeta que se conecta a una ranura de expansión que se encuentra en la placa base del sistema, se tiene capacidad hasta 4 unidades de disco duro.</p> <p>SCSI --> Es una tarjeta que se conecta a una ranura de expansión que se encuentra en la placa base del sistema, se tiene capacidad hasta 7 unidades de disco duro.</p> <p>IWM --> Integrated Woz Machine, lo utiliza la marca Macintosh.</p> <p>1.2.3.2 Partes</p> <p>Un disco duro esta compuesto principalmente por:</p> <p>Pista</p> <p>Se le define así al conjunto de bytes que se encuentran en la superficie del disco, al cual es posible acceder sin mover el brazo de acceso.</p> <p>Sector</p> <p>Se define así a la porción referenciable más pequeña que tiene un disco. Una pista esta dividida en varios sectores.</p> <p style="text-align: right;">18</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>
<p style="text-align: center;">Manejo e Implementación de Archivos</p> <p>Cilindros</p> <p>Son pistas que se encuentran unas encima de otras. Su importancia radica en que se puede tener acceso a toda la información que se encuentra almacenada sin tener que mover el brazo que esta sosteniendo las cabezas de lectura/escritura.</p> <p>Cluster</p> <p>Se define así a la unidad de asignación que asigna varios sectores, es decir donde se almacenan los datos de manera organizada.</p> <p>Cada unidad solo puede utilizar un archivo, pero un archivo si puede utilizar varias unidades.</p> <p>El proceso de acceder a un byte que se encuentra en un archivo es el siguiente:</p> <ul style="list-style-type: none"> - el sistema operativo busca la superficie, pista y sector correctos - lee el sector y es colocado en un espacio reservado en memoria RAM, a este espacio se le llama buffer - después se busca dentro del buffer el byte solicitado <p>La capacidad de un disco duro se obtiene de la siguiente forma:</p> <p style="text-align: right;">19</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>	<p style="text-align: center;">Manejo e Implementación de Archivos</p> <div style="border: 1px solid black; padding: 5px; margin: 10px auto; width: fit-content;"> <p style="font-size: x-small; margin: 0;">capacidad de la unidad = #cilindros x capacidad del cilindro</p> <p style="font-size: x-small; margin: 0;">capacidad del cilindro = #pistas por cilindro x capacidad de la pista</p> <p style="font-size: x-small; margin: 0;">capacidad de la pista = #sectores por pista x bytes por sector</p> </div> <p>Referenciación de datos</p> <p>Existen dos formas para referenciar datos:</p> <p>Por bloques</p> <p>Esta se refiere a cuando un grupo de registros se son almacenados juntos en el disco y son tratados como una sola unidad. Su principal desventaja es el peligro de fragmentación interna, por el espacio que queda entre cada bloque.</p> <p>Por sectores</p> <p>Esto se refiere a acceder lógicamente adyacente, esto quiere decir que cada sector se ve como un segmento de pista adyacente de un tamaño fijo, lo cuales pueden contener un archivo.</p> <p style="text-align: right;">20</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>

Figura 44 – Páginas 21-24 Libro “Manejo e Implementación de Archivos”

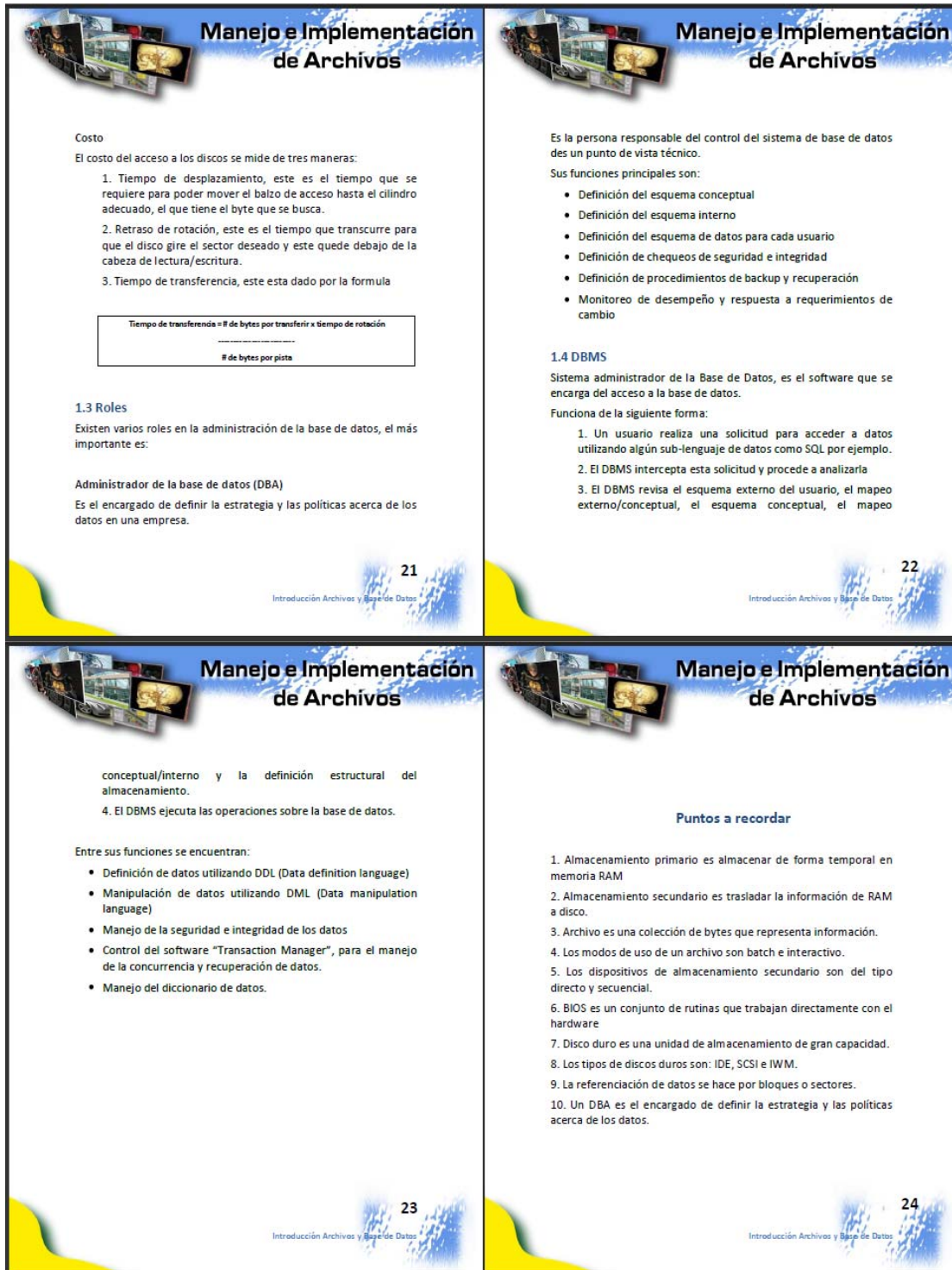


Figura 45 – Páginas 25-28 Libro “Manejo e Implementación de Archivos”

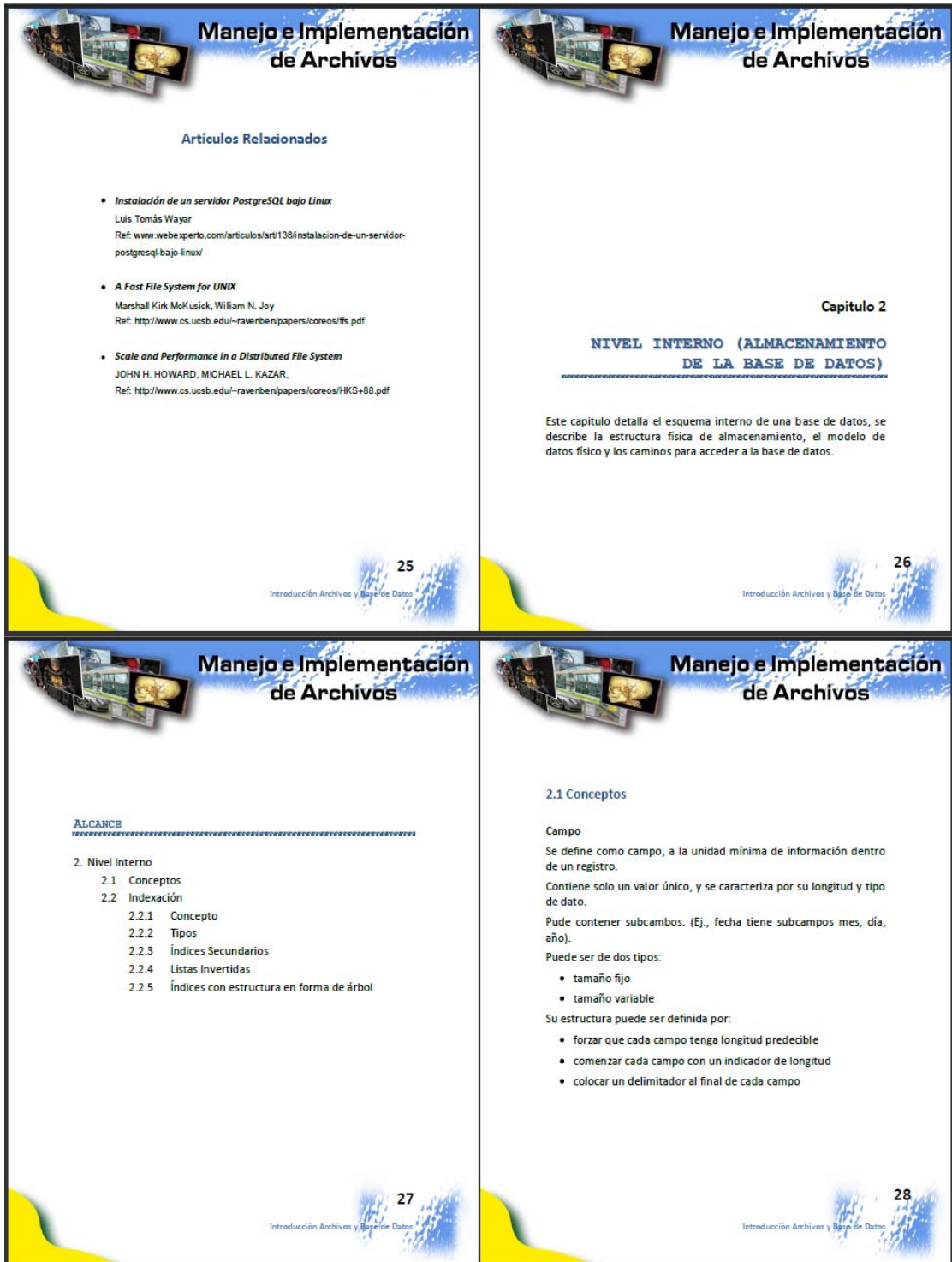


Figura 46 – Páginas 29-32 Libro “Manejo e Implementación de Archivos”

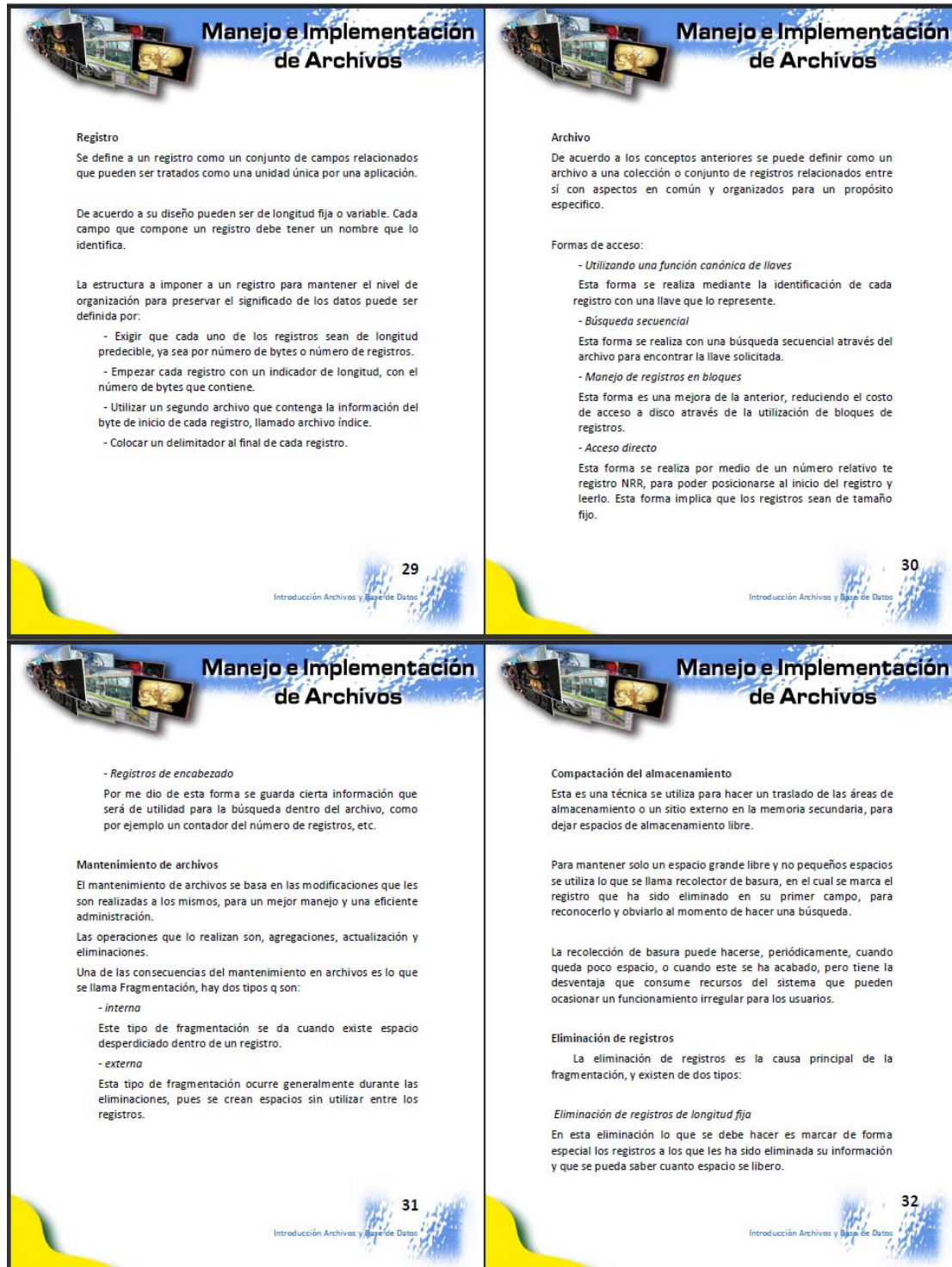


Figura 47 – Páginas 33-36 Libro “Manejo e Implementación de Archivos”





 <h2 style="text-align: center;">Manejo e Implementación de Archivos</h2> <p>Para este tipo de eliminación se usan las técnicas:</p> <ul style="list-style-type: none"> - Uso de listas ligadas: Esto se refiere a una lista la cual contiene todos los registros que se encuentran disponibles, si la lista esta compuesta por los registros que han sido eliminados, se le llama lista de disponibles, en un sistema de registros de longitud fija no hay necesidad de ordenar la lista pues cualquier registro libre es bueno. - Uso de pilas: Es el modo de manejar una lista, en donde se hacen las inserciones y eliminaciones en uno de los extremos. <p><i>Eliminación de registros de longitud variable</i></p> <ul style="list-style-type: none"> - Lista de disponibles: la diferencia con la lista de disponibles de registros de longitud fija, es que en la de longitud variable hay que asegurarse que la entrada sea de un tamaño correcto, es decir que el espacio donde se va a almacenar el registro debe ser lo suficientemente grande. <p>Estrategias para evitar la fragmentación</p> <p>Para la fragmentación interna, que es cuando el espacio se pierde dentro de un registro, esto se logra evitar desarrollando un procedimiento que divida la entrada grande de longitud variable en</p> <p style="text-align: right;">33</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>	 <h2 style="text-align: center;">Manejo e Implementación de Archivos</h2> <p>dos o más pequeñas partes, usando así solo el espacio que se necesita para el registro nuevo y el resto colocarlo en la lista de disponibles.</p> <p>El problema es que con el tiempo se va reduciendo la capacidad de espacio por estar en fragmentos demasiado pequeños, generando la fragmentación externa, para combatirla se tiene:</p> <ul style="list-style-type: none"> - Compactar el archivo, utilizando un procesamiento por lotes. - Unificar los espacios adyacentes que estén en la lista de disponibles y así tener espacios más grandes y más útiles. - Utilizar una estrategia de colocación para utilizar de mejor manera los espacios que se utilizarán. Entre esas estrategias están: <ul style="list-style-type: none"> - <i>Primer ajuste</i>: Esa estrategia consiste en si el espacio es lo suficientemente grande, ese se utiliza. - <i>Mejor ajuste</i>: En esta estrategia el nuevo registro se coloca en el espacio más pequeño pero que alcance para almacenarlo. Esta estrategia implica un costo en el rendimiento de la aplicación que lo utiliza. - <i>Peor ajuste</i>: En esta estrategia el nuevo registro se coloca en el espacio más grande posible, con el objetivo de dejar el espacio sobrante más grande posible. <p style="text-align: right;">34</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>
 <h2 style="text-align: center;">Manejo e Implementación de Archivos</h2> <h3>2.2 Indexación</h3> <h4>2.2.1 Concepto</h4> <p>Se puede definir a un índice como una estructura de datos, la cual contiene una función, donde el argumento que tiene la función es una clave y el valor de resultado de la función es un número de registro.</p> <p>En resumen, por medio de un índice se puede encontrar un número de registro que este asociado a una clave.</p> <p>Indexación consiste entonces en proporcionar varios caminos para acceder a los registro de un archivo.</p> <h4>2.2.2 Tipos</h4> <p>Entre las principales formas de indexación están:</p> <p>Índice simple con entradas de tipo secuencial</p> <p>Este tipo de indexación se realiza através de simples arreglos de estructuras las cuales contiene un campo llave y otros de referencia.</p> <p style="text-align: right;">35</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>	 <h2 style="text-align: center;">Manejo e Implementación de Archivos</h2> <p>La ventaja de esta indexación es que la utilización de la búsqueda binaria puede ser reemplazada por una más eficiente como la búsqueda por un árbol de índices.</p> <p>Entre sus operaciones básicas están:</p> <ul style="list-style-type: none"> - Creación de archivos: Se crean archivos vacíos para los índices y los datos, solo con registros de encabezados. - Carga del archivo de índices a memoria: Generalmente es definido en memoria como un arreglo, teniendo cada elemento la estructura del registro del índice. - Reescribir el archivo de índices: Esto es debido a los cambios que pudieron haberse hecho al arreglo en memoria, por lo tanto es importante considerar ciertos factores al momento de la reescritura: <ul style="list-style-type: none"> - Colocar un indicador o bandera de estado de cuando el índice no esta actualizado. - Tener un proceso reconstructor que actualice los índices cuando no lo están. - Agregar nuevos registros ya sea al archivo de datos o al de índices - Eliminación registros: En esta operación se utilizan los métodos antes mencionados (listas, pilas), también es necesario actualizar el archivo de índices. <p style="text-align: right;">36</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>

Figura 48 – Páginas 37-40 Libro “Manejo e Implementación de Archivos”





 <h2 style="text-align: center;">Manejo e Implementación de Archivos</h2> <p>- Actualización de registros: Existen dos posibles actualizaciones, la primera es cuando se actualiza el valor de llave, que conlleva a un reacomodo del archivo de índices y de datos, y cuando solo se actualiza el archivo de datos, en la cual es importante determinar el espacio de la actualización por si es necesario cambiar de lugar el registro.</p> <p>2.2.3 Índices Secundarios</p> <p>Estos índices son los que se aplican a uno o varios campos de los registros del archivo de datos, para los cuales es necesario localizar rápidamente los registros a partir de estos campos, como por ejemplo un catalogo de biblioteca.</p> <p>2.2.4 Listas Invertidas</p> <p>Se define como listas invertidas a los archivos que funcionan como índices secundarios, en la que una llave secundaria, lleva a un conjunto de una o más de una llaves primarias.</p> <p>Su principal ventaja es que en el momento de agregar una llave secundaria, solo se tiene que reacomodar la lista que esta asociada a la llave primaria.</p> <p style="text-align: right;">37</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>	 <h2 style="text-align: center;">Manejo e Implementación de Archivos</h2> <p>Su desventaja es la "localidad" de los registros, pues existe menos posibilidad que se encuentren físicamente adyacentes.</p> <p>2.2.5 Índices con estructura en forma de árbol</p> <p>Este tipo de estructura es en forma de jerarquía, la raíz bien a ser el primer nivel y este apunta al segundo nivel, y así sucesivamente hasta los niveles menores, llegando a los nodos hojas que tienen punteros al archivo de datos.</p> <p>Ofrece las ventajas:</p> <ul style="list-style-type: none"> - flexibilidad: permite el procesamiento de diferentes modos. - eficiencia de búsqueda: principalmente por el filtro que se puede hacer para localizar el elemento deseado, pasando de largo los elementos no deseados. - naturalidad en la representación: su representación se hace de forma sencilla. - poder de representación: hacen posible la representación que no podría funcionar de otro modo. <p>Existen varios tipos q son:</p> <ul style="list-style-type: none"> - Árbol binario: es un árbol en el que sus nodos no solo proporcionan dirección hacia otros nodos, sino que también <p style="text-align: right;">38</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>
 <h2 style="text-align: center;">Manejo e Implementación de Archivos</h2> <p>guardan información. Está organizado que las claves mayores del nodo padre se encuentran en el subárbol del lado derecho y las menores en el subárbol del lado izquierdo.</p> <ul style="list-style-type: none"> - Árbol AVL: Es un árbol de búsqueda binario en el que los subárboles de cada nodo tratan de mantenerse a la misma profundidad, es decir de forma equilibrada. - Árbol B: Su composición es similar a la de un bloque de registros que posee múltiples entradas. Dependiendo su número de orden es el número de registros que puede almacenar. <p>Árbol B* En este árbol B todos sus nodos son del mismo tamaño, su raíz puede contener hasta $4d+1$ registros.</p> <p>Árbol B# En este árbol B las operaciones de división e inserción se realizan de un modo en que se pospone la división, esto resulta en una mayor utilización del espacio y un rendimiento más rápido en la recuperación.</p> <p>Árbol B+ En este árbol B los datos están en los nodos terminales, y las direcciones en los no terminales, lo que lo hace menos profundo y se accede de forma más rápida al registro deseado.</p> <p style="text-align: right;">39</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>	 <h2 style="text-align: center;">Manejo e Implementación de Archivos</h2> <p style="text-align: center;">Puntos a recordar</p> <ol style="list-style-type: none"> 1. Campo es la unidad mínima de información dentro de un registro 2. Registro es un conjunto de campos relacionados 3. Archivo es una colección de registros relacionados 4. La fragmentación puede ser interna o externa 5. La eliminación puede ser para registros de longitud fija o longitud variable. 6. Las estrategias de colocación pueden ser: primer, mejor y peor ajuste. 7. Por medio de un índice se puede encontrar un registro asociado a una clave. 8. Las listas invertidas son los archivos que funcionan como índices secundarios. 9. Los tipos de índices en forma de árbol son: binario, AVL y B. 10. Los tipos de árbol B son: B*, B# y B+ <p style="text-align: right;">40</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>

Figura 49 – Páginas 41-44 Libro “Manejo e Implementación de Archivos”



Figura 50 – Páginas 45-48 Libro “Manejo e Implementación de Archivos”





 <h2 style="text-align: center;">Manejo e Implementación de Archivos</h2> <p>Para crear una tabla se hace por medio de la sentencia CREATE TABLE, esta sentencia tiene el siguiente formato:</p> <pre>CREATE TABLE tabla_base (Definicion_columna [, definicion_columna] ... [definicion_llave_primaria] [definicion_llave_foranea [, definicion_llave_foranea]...]);</pre> <p>Donde la definición de columna es:</p> <pre>Tipo_de_dato_Columna [NOT NULL]</pre> <p>Modificar la estructura de la Tabla</p> <pre>ALTER TABLE <tabla_base> ADD <nombre_columna> <tipo de dato></pre> <p>Eliminar Tabla</p> <pre>DROP TABLE <tabla_base></pre> <p>Manejo de Índices</p> <pre>CREATE [UNIQUE] INDEX <nombre_indice></pre> <p style="text-align: right;">45</p> <p style="text-align: right; font-size: small;">Introducción Archivos y Base de Datos</p>	 <h2 style="text-align: center;">Manejo e Implementación de Archivos</h2> <pre>ON <nombre_tabla_base> (<nombre_columna>[orden][,<nombre_columna>[orden]]...) [CLUSTER];</pre> <p>Order puede tomar los valores ASC para ordenamiento ascendente o DESC para descendente.</p> <p>Eliminación de un índice</p> <pre>DROP INDEX <nombre_indice></pre> <h3>3.2 Lenguaje de Manipulación de Datos</h3> <p>Provee las sentencias:</p> <pre>SELECT [DISTINCT] lista de ítems FROM lista de tablas [WHERE condicion(es)] [GROUB BY lista de campos] [HAVING condiciones] [ORDER BY lista de campos];</pre> <p style="text-align: right;">46</p> <p style="text-align: right; font-size: small;">Introducción Archivos y Base de Datos</p>
 <h2 style="text-align: center;">Manejo e Implementación de Archivos</h2> <p>La cláusula DISTINCT quita las tuplas duplicadas.</p> <p>Sentencia JOIN</p> <p>Por medio de esta función se unen una o más tablas.</p> <p>En la consulta:</p> <pre>SELECT pr.* , pt* FROM proveedor pr, parte pt WHERE pr.lugar = pt.lugar</pre> <p>La condición que se realiza entre los campos "lugar" que son iguales se le conoce como "Condición JOIN".</p> <p>Si la Condición JOIN es una igualdad es conocido como EQUJOIN. El EQUJOIN por definición debe mostrar en el resultado dos columnas con datos idénticos, cuando una de estas es eliminada del Query, se le conoce como NATURAL JOIN.</p> <p style="text-align: right;">47</p> <p style="text-align: right; font-size: small;">Introducción Archivos y Base de Datos</p>	 <h2 style="text-align: center;">Manejo e Implementación de Archivos</h2> <p>Otras funciones</p> <ul style="list-style-type: none"> • Funciones de agregación <ul style="list-style-type: none"> COUNT - devuelve la cantidad de valores que hay en la columna. SUM - devuelve la suma de los valores que hay en la columna AVG - devuelve el valor promedio de los valores que hay en la columna MAX - devuelve el valor más alto en la columna MIN - devuelve el valor más bajo en la columna • Manejo de cadenas <p>Sentencia LIKE</p> <p>Es de la forma: columna LIKE valor_cadena, donde la columna tiene que ser de tipo String.</p> <ul style="list-style-type: none"> - "_", este caracter sustituye a cualquier caracter simple. - "%", este caracter sustituye a cualquier secuencia de n caracteres. <p style="text-align: right;">48</p> <p style="text-align: right; font-size: small;">Introducción Archivos y Base de Datos</p>

Figura 51 – Páginas 49-52 Libro “Manejo e Implementación de Archivos”





 <h2 style="text-align: center;">Manejo e Implementación de Archivos</h2> <p>Ejemplo: Estudiantes que empiecen con A</p> <pre>SELECT * FROM estudiantes WHERE nombre LIKE 'A%';</pre> <p>Subconsultas</p> <p>Una subconsulta en un query de la forma SELECT - FROM - WHERE - GROUP BY - HAVING que esta anidado dentro de otra expresión del mismo tipo.</p> <p>Una subconsulta es utilizada para representar un conjunto de valores que se buscan por medio de la condición IN.</p> <p>También se utiliza la sentencia exists, con la cual se obtienen si algún valor existe entre el grupo de datos obtenidos.</p> <p>Cláusulas START WITH y CONNECT BY</p> <p>Estas cláusulas se utilizan para realizar consultas sobre datos con relaciones jerárquicas (Ej. relación padre/hijo).</p> <p>Para condiciones recursivas, se usa la sentencia PRIOR</p> <p>Ej. CONNECT BY PRIOR hijo = padre // la hace recursiva</p> <p style="text-align: right;">49</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>	 <h2 style="text-align: center;">Manejo e Implementación de Archivos</h2> <h3>3.3 El catalogo del Sistema</h3> <p>En el catalogo del sistema se tiene información de interés acerca de los objetos del sistema, como tablas base, vistas, índices, usuarios, etc.</p> <p>Su principal ventaja es que almacena la información en forma de tablas o relaciones que contenga la información específica del sistema.</p> <p>System Tables</p> <p>En esta tabla se guarda la información de cada tabla base o vista, colocando su nombre, el usuario dueño, el numero de columnas, etc.</p> <p>System Columns</p> <p>Esta tabla contiene una fila por cada columna de cada tabla que se encuentra en System Tables, con la información del nombre, nombre de la tabla a la que pertenece, tipo de dato, etc.</p> <p style="text-align: right;">50</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>
 <h2 style="text-align: center;">Manejo e Implementación de Archivos</h2> <p>System Indexes</p> <p>Esta tabla contiene toda la información de los índices del sistema. Tiene información del nombre del índice, nombre de la tabla indexada, usuario creador, etc.</p> <h3>3.4 Vistas</h3> <p>Se puede definir como una vista desde el punto de vista de SQL, como una tabla virtual, que esta nombrada y es derivada de otras tablas base.</p> <p>Se almacenan en el catalogo, en la tabla SYSVIEWS.</p> <pre>CREATE VIEW nombre_vista [(nombre_columna, nombre_columna, ...)] AS subquery [WITH CHECK OPTION] ;</pre> <p>La opción WITH CHECK OPTION, se utiliza para indicar que las operaciones INSERT y UPDATE que se realizan por medio de la vista deben ser verificadas, para poder así asegurar que se satisfaga la condición descrita en la vista.</p> <p style="text-align: right;">51</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>	 <h2 style="text-align: center;">Manejo e Implementación de Archivos</h2> <p>Ejemplo:</p> <pre>CREATE VIEW buenos_proveedores AS SELECT p#, Estatus, Ciudad From proveedores Where Estatus > 15;</pre> <p>Eliminar vistas</p> <pre>DROP VIEW nombre_vista</pre> <p>Independencia Lógica de los datos</p> <p>La independencia lógica de los datos, es el propósito por el cual existen las vistas, y es utilizada para que los usuarios no sean dependientes de la estructura física de la base de datos, y a su vez también que sean independientes de la estructura lógica de la misma, para este tipo de independencia se consideran los aspectos:</p> <ul style="list-style-type: none"> - crecimiento - En la expansión de una tabla base, por medio de la adición de un nuevo campo. - La creación de una nueva tabla base. <p>Estos cambios no deben de afectar a los usuarios existentes de algún modo.</p> <p style="text-align: right;">52</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>

Figura 52 – Páginas 53-56 Libro “Manejo e Implementación de Archivos”

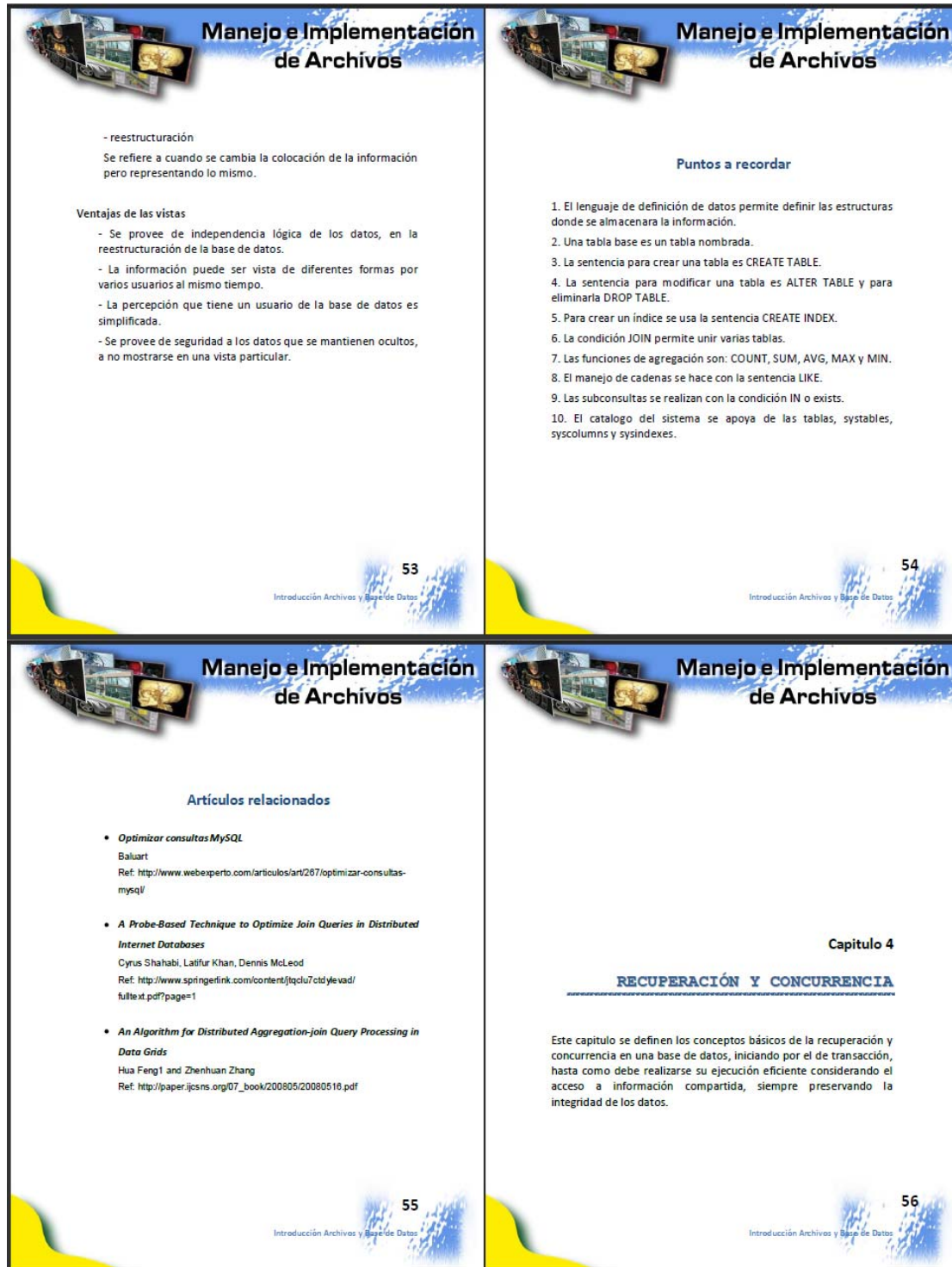


Figura 53 – Páginas 57-60 Libro “Manejo e Implementación de Archivos”

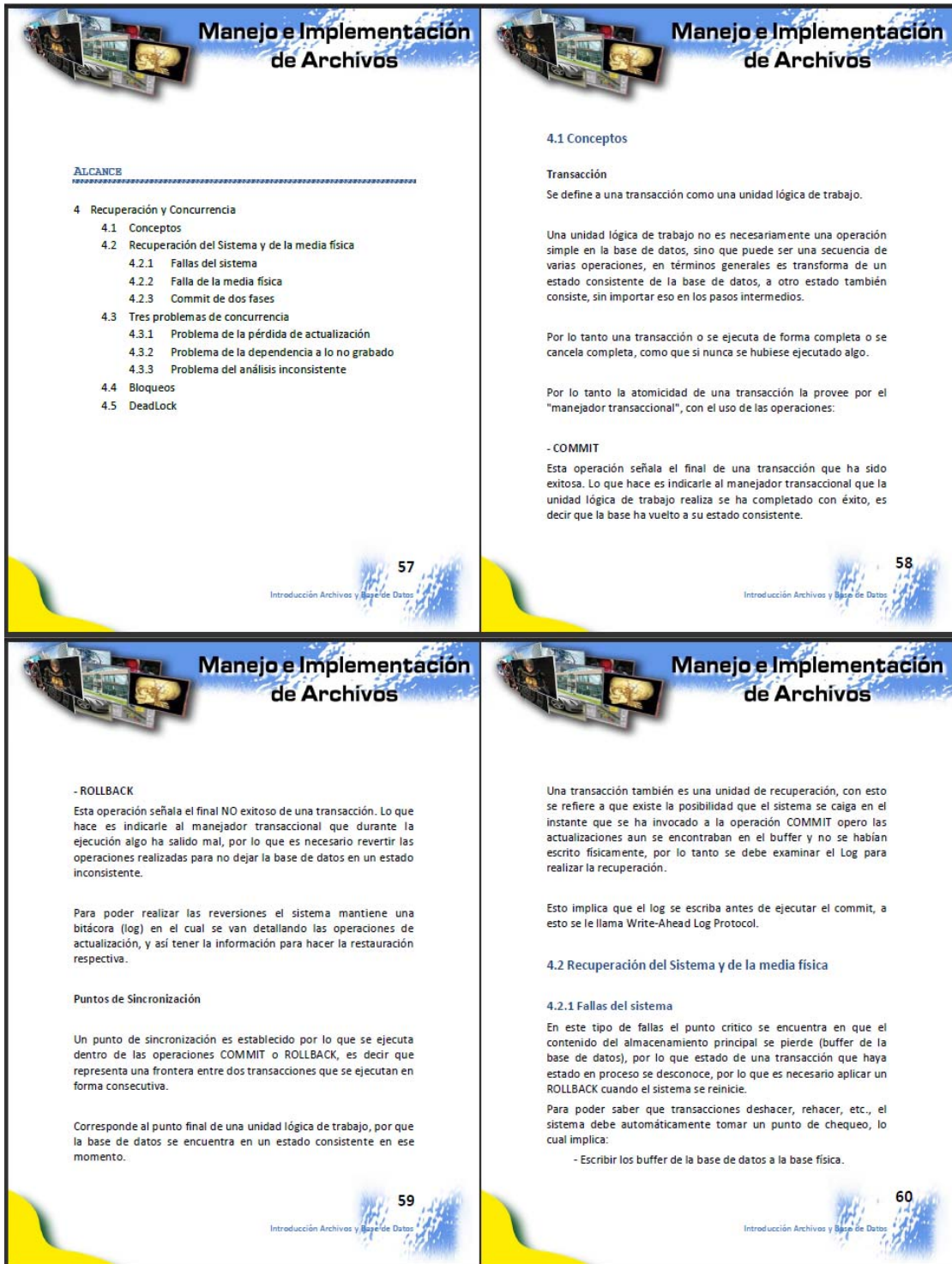


Figura 54 – Páginas 61-64 Libro “Manejo e Implementación de Archivos”

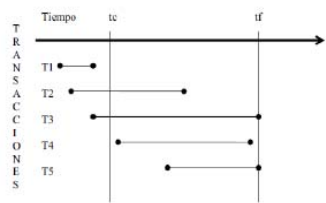
<p style="text-align: center;">Manejo e Implementación de Archivos</p> <p>- Por medio del log se obtienen una lista con las transacciones que se encontraban ejecutando en ese momento, el siguiente diagrama muestra como la información es utilizada:</p>  <p style="text-align: center;">Figura No. 3 - Transacciones</p> <ul style="list-style-type: none"> - Una falla del sistema ocurre en un tiempo t_f - El checkpoint más reciente, se realizó antes del tiempo t_f y se ejecutó en el tiempo t_c - Transacciones del tipo T1 se completaron antes del tiempo t_c. - Transacciones del tipo T2 comenzaron antes del tiempo t_c y terminaron después del tiempo t_c y antes del tiempo t_f <p style="text-align: right;">61</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>	<p style="text-align: center;">Manejo e Implementación de Archivos</p> <ul style="list-style-type: none"> - Transacciones del tipo T3 también comenzaron antes del tiempo t_c pero no fueron completadas en el tiempo t_f. - Transacciones del tipo T4 comenzaron después del tiempo t_c y terminaron antes del tiempo t_f. - Transacciones del tipo T5 también comenzaron después del tiempo t_c, pero no se habían completado durante el tiempo t_f. <p>Por lo tanto, al reiniciar el sistema, las transacciones del tipo T3 y T5 deben revertirse, y transacciones del tipo T2 y T4 deben volver a realizarse.</p> <p>4.2.2 Falla de la media física</p> <p>Estas fallas se dan cuando la base de datos física es destruida o dañada.</p> <p>La recuperación de este tipo de fallas involucra la restauración de la base de datos por medio de un backup, y con el archivo Log deshacer las transacciones que se realizaban en ese momento.</p> <p>4.2.3 Commit de dos fases</p> <p>Es un algoritmo de forma distribuida que permite que los nodos del sistema se pongan de acuerdo al hacer el commit de una transacción. Es decir que o todos hacen el commit de una transacción o todos fallan.</p> <p style="text-align: right;">62</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>															
<p style="text-align: center;">Manejo e Implementación de Archivos</p> <p>Las fases en la que se divide son:</p> <ul style="list-style-type: none"> - Petición del commit: en esta fase un coordinador se encarga de preparar a todos los nodos. - Fase de commit: donde el coordinador se encarga de velar que se completen las transacciones de todos los nodos. <p>4.3 Tres problemas de concurrencia</p> <p>En un sistema multiusuario, es necesario mantener el control en el acceso simultáneo a la base de datos, por lo tanto es necesario llevar un control que no permita que ocurran los siguientes problemas:</p> <p>4.3.1 Problema de la pérdida de actualización</p> <p style="text-align: right;">63</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>	<p style="text-align: center;">Manejo e Implementación de Archivos</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 30%;">Transacción A</th> <th style="width: 40%;">Tiempo</th> <th style="width: 30%;">Transacción B</th> </tr> </thead> <tbody> <tr> <td>FETCH R</td> <td style="text-align: center;">t_1</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">t_2</td> <td style="text-align: center;">FETCH R</td> </tr> <tr> <td>UPDATE R</td> <td style="text-align: center;">t_3</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">t_4</td> <td style="text-align: center;">UPDATE R</td> </tr> </tbody> </table> <p style="text-align: center;">Figura No. 4 – Pérdida de actualización</p> <p>Como se observa en la figura, la transacción A obtiene el valor de R en el tiempo t_1, luego la transacción B obtiene el valor del mismo registro r en t_2, TA actualiza en el t_3, y TB actualiza en el t_4, pero con el valor obtenido antes de la actualización de TA, por lo tanto dicha actualización se pierde.</p> <p>4.3.2 Problema de la dependencia a lo no grabado</p> <p>Este problema se da cuando se permite que una transacción pueda obtener o actualizar un registro sin que se le haya dado COMMIT, por parte de la otra transacción, pudiendo esta fallar y realizar un</p> <p style="text-align: right;">64</p> <p style="text-align: center; font-size: small;">Introducción Archivos y Base de Datos</p>	Transacción A	Tiempo	Transacción B	FETCH R	t_1			t_2	FETCH R	UPDATE R	t_3			t_4	UPDATE R
Transacción A	Tiempo	Transacción B														
FETCH R	t_1															
	t_2	FETCH R														
UPDATE R	t_3															
	t_4	UPDATE R														

Figura 55 – Páginas 65-68 Libro “Manejo e Implementación de Archivos”

Manejo e Implementación de Archivos

ROLLBACK, quedando la transacción anterior con información que ahora ya no existe.

Transacción A	Tiempo	Transacción B
	t1	UPDATE R
	t2	
FETCH R	t3	ROLLBACK R

Figura No. 5 – Dependencia a lo no grabado

4.3.3 Problema del análisis inconsistente

65

Introducción Archivos y Base de Datos

Manejo e Implementación de Archivos

ACC 1	ACC 2	ACC 3
40	50	30

Transacción A	Tiempo	Transacción B
	t1	...
FETCH ACC1 (40)	t1	...
Sum = 40	t1	...
	t2	...
FETCH ACC2 (50)	t2	...
Sum = 50	t2	...
	t3	FETCH ACC3 (30)
	t3	...
	t4	UPDATE ACC3
	t4	30 → 20
	t5	...
	t5	FETCH ACC1 (40)
	t5	...
	t6	UPDATE ACC1
	t6	40 → 20
	t7	...
	t7	COMMIT
FETCH ACC3 (20)	t8	...
Sum = 110 - no 120	t8	...

Figura No. 6 – Análisis de inconsistencia

De acuerdo a la imagen se esta operando sobre un registro de cuenta ACC, la transacción A suma los balances de cuenta, la transacción B transfiere una cantidad 10 desde la cuenta 3 a la 1. Se observa que el resultado que el resultado final (110) queda inconsistente y se diferencia del problema anterior a que el cambio

66

Introducción Archivos y Base de Datos

Manejo e Implementación de Archivos

no es almacenado sino que viene de la actualización hecha por la transacción B.

4.4 Bloqueos

La utilización de los bloqueos es para garantizar que un objeto será utilizado por una transacción a la vez.

De esta forma se tiene la certeza que durante la primera transacción que tome el objeto este se mantendrá en un estado estable por el tiempo en que este en uso por la transacción.

Existen dos tipos de bloqueos:

- exclusivos (X)

Cuando una transacción coloca un bloqueo exclusivo (X) sobre un registro, entonces si otra transacción solicita el uso de ese registro, la coloca en un estado de espera hasta que la primera transacción libere el bloqueo.

- compartidos (S)

Si una transacción coloca un bloqueo compartido sobre un registro, y luego una transacción trata de poner un bloqueo de tipo X, esto hará que la transacción se quede esperando hasta que la primera transacción libere el bloqueo, pero si el bloqueo que trata de colocar es de tipo S sobre el registro si será permitido.

67

Introducción Archivos y Base de Datos

Manejo e Implementación de Archivos

A continuación se presenta la matriz de compatibilidad de los bloqueos:

	X	S	-
X	N	N	Y
S	N	Y	Y
-	Y	Y	Y

Tabla No. 1 – Matriz de Compatibilidad

- el guión representa que no existe un bloqueo
- la letra N, que ocurre un conflicto y se queda en espera para poder colocar el bloqueo
- la letra Y, indica que se puede colocar el bloqueo

68

Introducción Archivos y Base de Datos

Figura 56 – Páginas 69-72 Libro “Manejo e Implementación de Archivos”

Manejo e Implementación de Archivos

Problemas resueltos con el bloqueo:

Transacción A	Tiempo	Transacción B
FETCH R S-lock sobre R	t1	
	t2	FETCH R S-lock sobre R
UPDATE R Requiere X-lock sobre R Esperar	t3	
	t4	UPDATE R Requiere X-lock sobre R Esperar

Figura No. 7 – Solución perdida actualización

Transacción A	Tiempo	Transacción B
	t1	UPDATE R (obtiene X-lock sobre R)
FETCH R (solicita S-lock)	t2	
Esperar Esperar (Obtiene registro R S-lock sobre R)	t3	ROLLBACK R (Suspende) Libera X-lock sobre R

Figura No. 8 – Solución dependencia a lo no grabado

69
Introducción Archivos y Base de Datos

Manejo e Implementación de Archivos

Transacción A	Tiempo	Transacción B
	t1	...
FETCH ACC1 (40) (obtiene S-Lock sobre ACC1) Sum = 40	t2	...
...	t3	FETCH ACC3 (30) (obtiene S-Lock sobre ACC3)
FETCH ACC2 (50) (obtiene S-Lock sobre ACC2) Sum = 50	t4	UPDATE ACC3 (obtiene X-Lock sobre ACC3) 30 → 20
...	t5	...
...	t6	FETCH ACC1 (40) (Obtiene S-Lock sobre ACC1)
...	t7	UPDATE ACC1 (requiere X-Lock sobre ACC1) Esperar
FETCH ACC3 (20) (requiere S-Lock sobre ACC3) Esperar	t8	...

Figura No. 9 – Análisis de inconsistencia

70
Introducción Archivos y Base de Datos

Manejo e Implementación de Archivos

4.5 DeadLock

A pesar de que el uso de bloqueos resuelve los problemas básicos de concurrencia, puede generar a su vez un problema, que es el deadlock.

En esta situación dos o más transacciones se quedan simultáneamente en el estado de espera, cada una esperando a la otra libere el bloqueo para poder proceder.

Transacción A	Tiempo	Transacción B
Requiere un X-Lock sobre R1	t1	...
...	t2	Requiere un X-Lock sobre R2
...	t3	...
Requiere X-Lock sobre R2 Esperar Esperar	t4	Requiere X-Lock sobre R1 Esperar

Figura No. 10 – DeadLock

Generalmente este problema se resuelve, haciendo que el sistema detecte el deadlock y las transacciones que se encuentren involucradas, revirtiendo una para que el resto continúe su proceso normal.

71
Introducción Archivos y Base de Datos

Manejo e Implementación de Archivos

Puntos a recordar

1. Transacción es una unidad lógica de trabajo.
2. Las operaciones de manejador son: COMMIT y ROLLBACK
3. Un punto de sincronización representa una frontera entre la ejecución dos transacciones consecutivas.
4. COMMIT señala el final exitoso de una transacción.
5. ROLLBACK señala que ha ocurrido un fallo y que las operaciones deben ser revertidas.
6. El COMMIT de dos fases, permite que los nodos de un sistema se pongan de acuerdo para hacer el COMMIT de una transacción.
7. El uso de un bloqueo es para garantizar que un objeto será utilizado por una transacción a la vez.
8. Los tipos de bloqueos son: exclusivos y compartidos.
9. El deadlock es el problema que se genera del uso de los bloqueos.

72
Introducción Archivos y Base de Datos

Figura 57 – Páginas 73-76 Libro “Manejo e Implementación de Archivos”

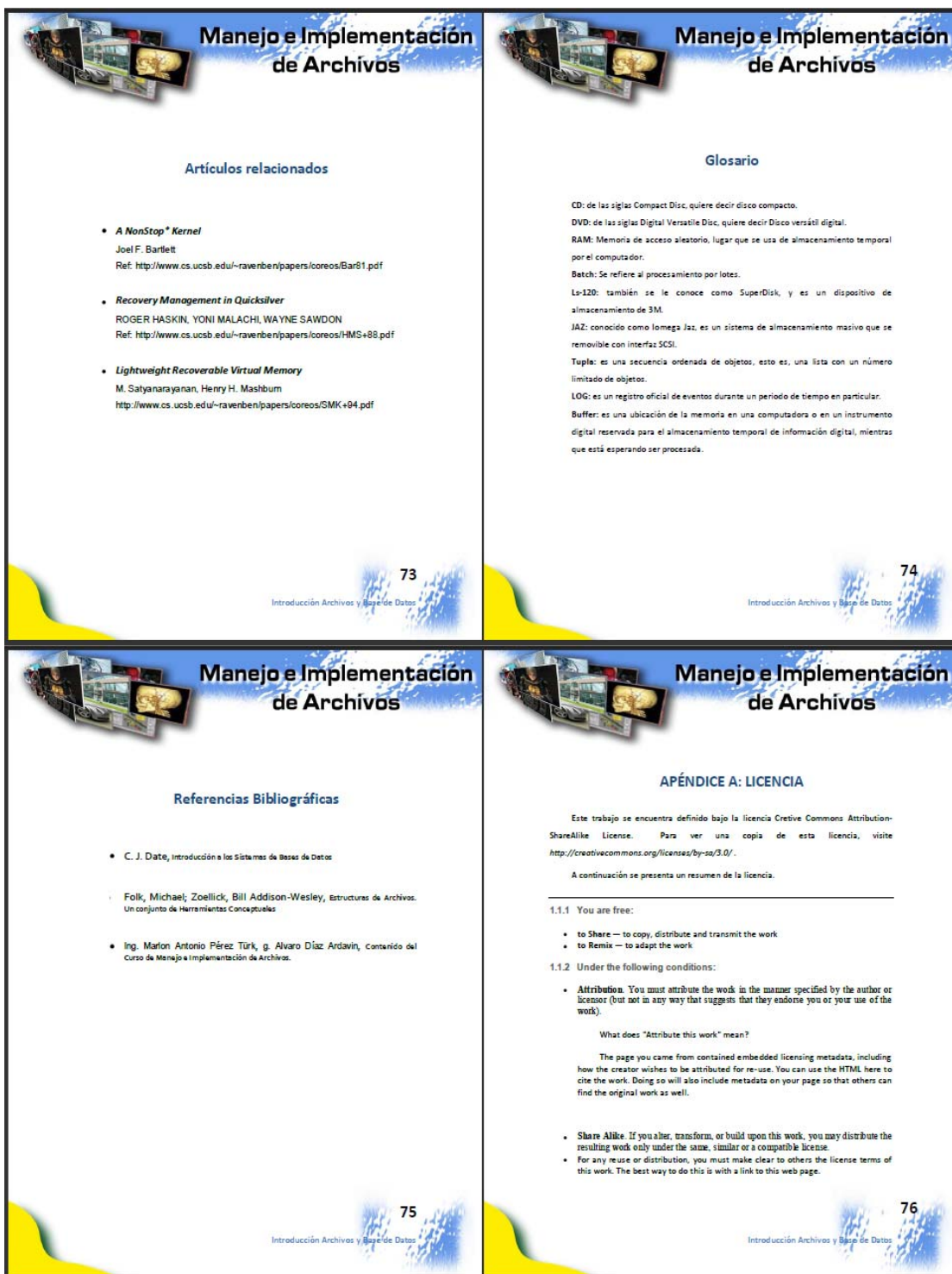


Figura 58 – Páginas 77-80 Libro “Manejo e Implementación de Archivos”

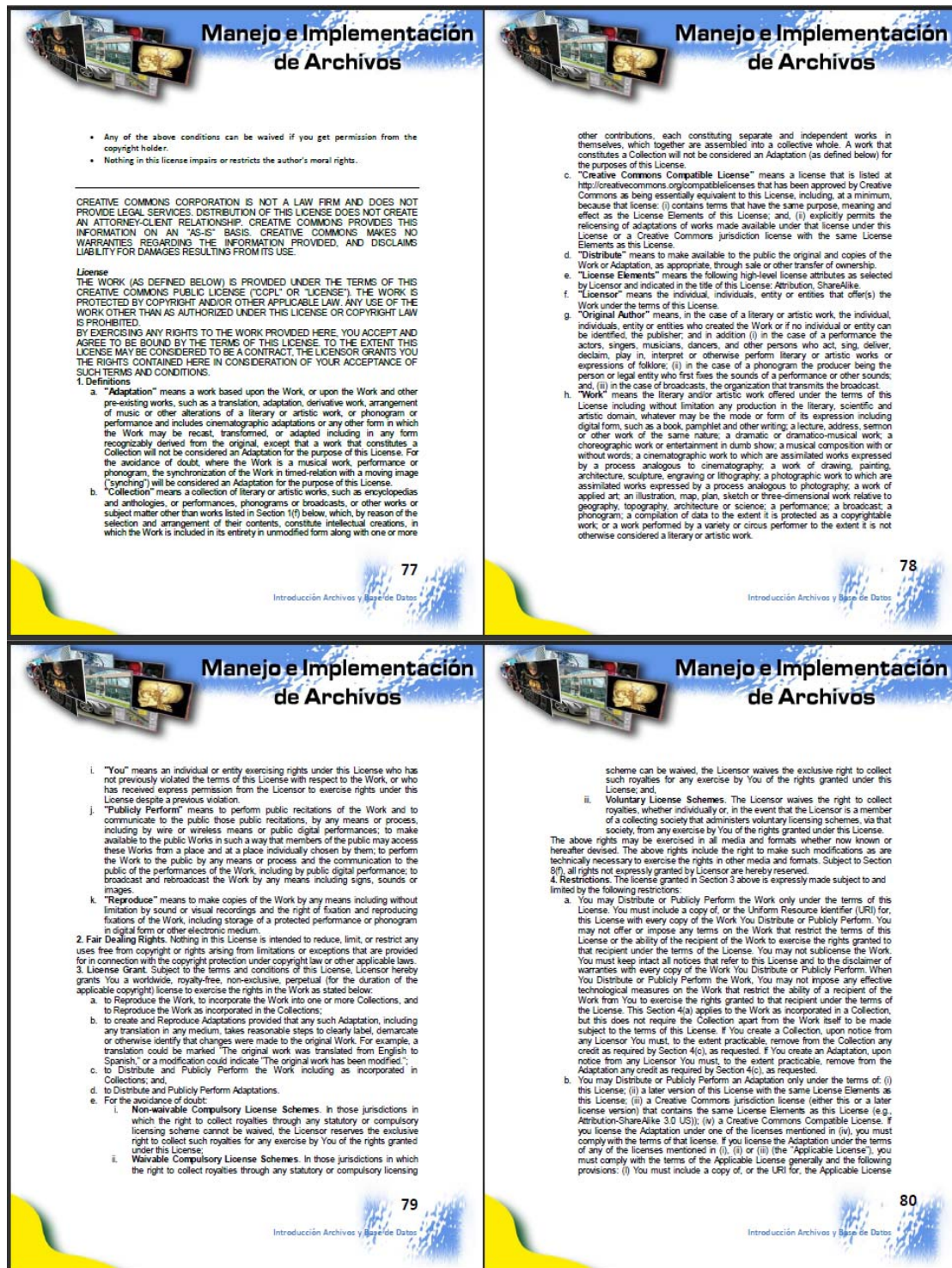
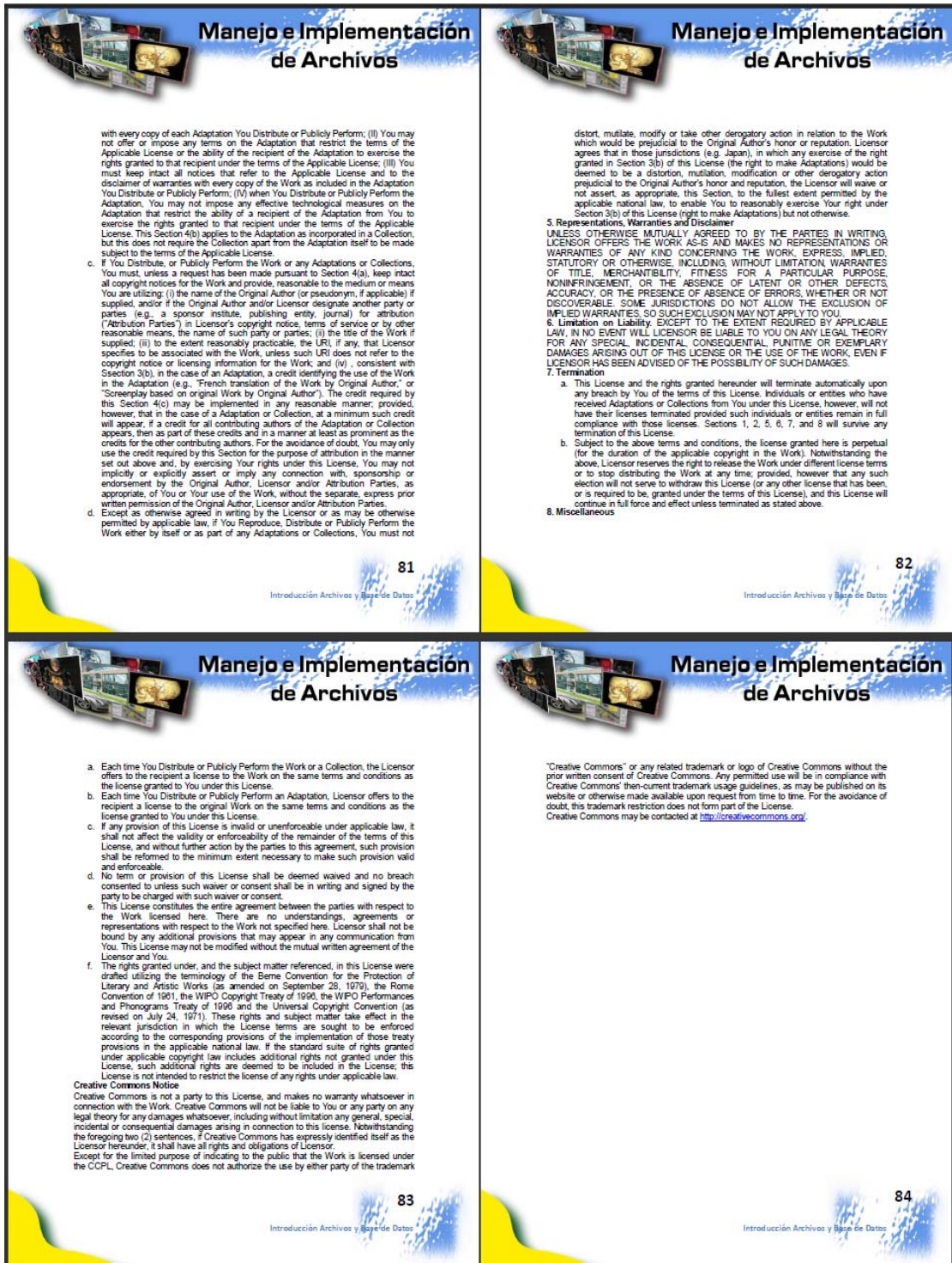


Figura 59 – Páginas 81-84 Libro “Manejo e Implementación de Archivos”



CONCLUSIONES

1. Se realizó la estructuración de los laboratorios de los cursos Sistemas Operativos 1, Sistemas Operativos 2 y Manejo e Implementación de Archivos, con base a la información sobre los contenidos actuales, contenidos de cursos que se imparten en los laboratorios de India-Guatemala IT Education Centre of Excellence, así como los propuestos y aceptos por los catedráticos titulares de cada uno de los cursos.
2. Se estructuró los puntos importantes a realizar para la definición de los proyectos, prácticas y tareas del laboratorio.
3. Se desarrolló un material sobre cada una de las sesiones de cada uno de los cursos.
4. Se desarrolló la documentación de apoyo de los cursos mencionados anteriormente, de acuerdos a los apuntes dados en clase, y a la documentación utilizada por los catedráticos de cada uno de los recursos respectivamente.
5. Se definieron las propuestas de los exámenes a realizar en cada uno de los laboratorios.
6. Se documentó una serie de artículos de interés relacionados a los contenidos específicos de cada una de las unidades de los cursos.

7. Con el desarrollo de estas herramientas para la enseñanza se podrá mejorar la forma de impartir los laboratorios, pues se realizará de forma ordenada, estandarizada y fácil de administrar por parte de la Escuela de Ciencias y Sistemas.

RECOMENDACIONES

1. Se sugiere a los instructores, auxiliares y encargados de la Escuela de Ciencias y Sistemas que se mantenga actualizados los contenidos, de preferencia en un período de seis meses, para garantizar que los temas que se están impartiendo sean los más adecuados.
2. También se recomienda llevar un control de versiones de los nuevos documentos que se generen a partir de cambios, para tener un registro histórico que pueda proporcionar información valiosa para analizar si se están cumpliendo los objetivos propios de cada “Instructor Guideline”.

BIBLIOGRAFÍA

1. Mrutunjaya Panda, **Curso de Planificación Educativa a través de *Instructor Guidelines***. India-Guatemala IT Education Centre of Excellence. Guatemala. 2008.
2. Araham Silberschatz & James, **Operating Systems Concepts**.
3. Marquez, Francisco Manuel Addison Wesley, **Unix. Programación Avanzada**.
4. Cabrera, Lina García, Francisco Marínez del Río & Redondo Duque, Miguel A. , **Sistemas Operativos**.
5. C.J. Date, **An introduction to Database Systems**, Fifth edition
6. Folk, Michael; Zoellick, Bill Addison-Wesley, **Estructuras de Archivos. Un conjunto de Herramientas Conceptuales**, Editorial Addison-Wesley Iberoamericana. USA
7. Andrew S. Tanenbaum, **Sistemas operativos modernos** 2a edición.

ANEXOS

En lo que se refiere a la estructuración de los laboratorios y el contenido de la documentación de apoyo se contó con el soporte y revisión de cada ingeniero responsable de cada uno cursos en la Escuela de Ciencias y Sistemas.

Por lo cual, en el anexo se adjuntan las cartas autorizando el contenido de la estructuración por parte de los ingenieros:

- Ing. Francisco Guevara – Sistemas Operativos 1
- Ing. René Ornelyz – Sistemas Operativos 2
- Ing. Álvaro Ardavin – Manejo e Implementación de Archivos

Donde se revisó todos los documentos necesarios y a criterio de cada ingeniero estos cumplen con el objetivo y contenido planteado para el laboratorio y documentación de cada curso.



Guatemala 26 de julio de 2008

Ingeniero
Jorge Armin Mazariegos
Asesor de EPS
Facultad de Ingeniería
USAC

Por este medio me dirijo a usted para informarle que he revisado el **Guideline de "Sistemas Operativos 1"**, del Proyecto "Estructuración de los laboratorios y documentación de apoyo de los cursos Sistemas Operativos 1, Sistema Operativos 2 y Manejo e Implementación de Archivos, y a mi parecer cumple con los objetivos de guía de laboratorio.

Agradeciendo su atención a la presente,

Atentamente,

Ing. Francisco Guevara
Catedrático de la Escuela de Ciencias y Sistemas
Facultad de Ingeniería, USAC



Guatemala 26 de julio de 2008

Ingeniero
Jorge Armin Mazariegos
Asesor de EPS
Facultad de Ingeniería
USAC

Por este medio me dirijo a usted para informarle que he revisado el **Guideline de "Sistemas Operativos 2"**, del Proyecto "Estructuración de los laboratorios y documentación de apoyo de los cursos Sistemas Operativos 1, Sistema Operativos 2 y Manejo e Implementación de Archivos, y a mi parecer cumple con los objetivos de guía de laboratorio.

Agradeciendo su atención a la presente,

Atentamente,



Ing. René Ornelyz
Catedrático de la Escuela de Ciencias y Sistemas
Facultad de Ingeniería, USAC


Vo. Bo.



Guatemala 30 de julio de 2008

Ingeniero
Jorge Armin Mazariegos
Asesor de EPS
Facultad de Ingeniería
USAC

Por este medio me dirijo a usted para informarle que he revisado el **Guideline de "Manejo e Implementación de Archivos"**, del Proyecto "Estructuración de los laboratorios y documentación de apoyo de los cursos Sistemas Operativos 1, Sistemas Operativos 2 y Manejo e Implementación de Archivos, y a mi parecer cumple con los objetivos de guía de laboratorio.

Agradeciendo su atención a la presente,

Atentamente,

Ing. Juan Álvaro Díaz Ardaín
Catedrático de la Escuela de Ciencias y Sistemas
Facultad de Ingeniería, USAC