

INSTITUTO TECNOLÓGICO Y DE ESTUDIOS
SUPERIORES DE MONTERREY

CAMPUS MONTERREY

PROGRAMA DE GRADUADOS EN ELECTRÓNICA,
COMPUTACIÓN, INFORMACIÓN Y
COMUNICACIONES



Una Herramienta de Plataforma Abierta de
Configuración de Conmutadores
TESIS

PRESENTADA COMO REQUISITO PARCIAL PARA OBTENER EL GRADO
ACADÉMICO DE:

MAESTRÍA EN CIENCIAS EN TECNOLOGÍA INFORMÁTICA

POR:

MARCO ANTONIO SAUCEDO VÉRTIZ

MONTERREY, NUEVO LEÓN, MÉXICO

MAYO 2004.

**INSTITUTO TECNOLÓGICO Y DE ESTUDIOS
SUPERIORES DE MONTERREY**

CAMPUS MONTERREY

**PROGRAMA DE GRADUADOS EN ELECTRÓNICA,
COMPUTACIÓN, INFORMACIÓN Y
COMUNICACIONES**



**Una Herramienta de Plataforma Abierta de
Configuración de Conmutadores
TESIS**

**PRESENTADA COMO REQUISITO PARCIAL PARA OBTENER EL GRADO
ACADEMICO DE:**

MAESTRÍA EN CIENCIAS EN TECNOLOGÍA INFORMÁTICA

POR:

MARCO ANTONIO SAUCEDO VÉRTIZ

MONTERREY, NUEVO LEÓN, MÉXICO.

MAYO 2004.

Una Herramienta de Plataforma Abierta de Configuración de Conmutadores

POR:

MARCO ANTONIO SAUCEDO VÉRTIZ

TESIS

**Presentada al Programa de Graduados en Electrónica, Computación,
Información y Comunicación.**

**Este trabajo es requisito parcial para obtener el grado de Maestro en
Tecnología Informática**

**INSTITUTO TECNOLÓGICO Y DE ESTUDIOS
SUPERIORES DE MONTERREY**

MAYO 2004

Dedicatoria

*A Dios, por todo lo que hasta este momento me ha dado,
por permitirme llegar a este momento de mi vida y poder superarme.*

*A mis padres, por enseñarme a ser la persona que soy,
por estar ahí en todo momento que los necesito
por todo eso y más, muchas gracias, los amo.*

*A mi novia Perla, por apoyarme en este paso importante de mi vida,
por estar conmigo en las buenas y en las malas, te amo.*

*A mis amigos, gracias por estar en todo momento a mi lado,
por las bromas, soy muy afortunado de tenerlos.*

Agradecimientos

*Agradezco a mi asesor por su apoyo en el desarrollo de mi tesis, Dr. Raúl Ramírez,
por su tiempo dedicado en este trabajo.*

*Agradezco infinitamente la ayuda de mis sinodales, Dr. Raúl Pérez,
por sus comentarios para lograr una tesis digna de maestría.
M.C. Moraima Cambell, por todo su tiempo dedicado a mí,
Muchas gracias.*

Resumen

La propuesta de tesis es realizada para la maestría en Ciencias de la Tecnología Informática. Las empresas que dependen fuertemente de las tecnologías de información se encuentran en la necesidad de contar con personal con suficientes habilidades profesionales para poder responder al reto que representan las demandas constantes sobre TIC(FALTA).

La constante evolución de estas tecnologías, implica que las empresas necesitan que sus profesionistas actualicen sus conocimientos de una manera más rápida. Para cumplir con su compromiso con la sociedad, las universidades tienen la necesidad de forjar estudiantes que posean habilidades técnicas y experiencia en el manejo de las nuevas tecnologías.

Es muy difícil para los profesionistas en informática poder someterse a un proceso de entrenamiento constante y así como también es difícil para los estudiantes de las carreras de informática, electrónica y telecomunicaciones adquirir esos conocimientos y habilidades que los prepare para ser productivos dentro de las empresas. Es por eso que se opta por la realización de una herramienta de simulación capaz de ofrecer entrenamiento realista sobre interconexión de redes, que sea accesible, haciendo que los estudiantes y profesionistas en el área de informática se vean beneficiados en su aprendizaje.

Lista de Contenido

Dedicatoria.....	iv
Agradecimientos.....	v
Resume.....	vi
Lista de Contenido.....	vii
Lista de Figuras.....	x
CAPÍTULO 1. INTRODUCCIÓN.....	1
1.1. Introducción.....	1
1.2. Metodología científica.....	2
1.2.1. Hipótesis.....	2
1.2.2. Prueba de hipótesis.....	2
1.3. Objetivo de la tesis.....	2
1.4. Definición del problema.....	3
1.5. Alcance de la tesis.....	3
1.6. Motivación.....	4
1.7. Organización de la tesis.....	4
CAPÍTULO 2. MARCO TEORICO.....	6
2.1. Ciclo de vida.....	6
2.1.1. Introducción.....	6
2.1.2. ¿Qué es el ciclo de vida?.....	6
2.1.3. Estructura de Datos.....	7
2.1.3.1. ¿Qué es una estructura de datos?.....	7
2.2. Conmutadores.....	8
2.2.1. Introducción.....	8
2.2.2. Tipos de conmutadores.....	9
2.2.2.1. Conmutadores LAN.....	10
2.2.3. Operaciones básicas de un conmutador.....	11
2.2.4. Segmentación de LAN.....	11
2.2.4.1. Ventajas y desventajas de la segmentación de LAN con conmutadores.....	12
2.2.4.2. Problemas de comunicación en las LAN.....	13
2.2.5. Conmutación en la capa 2 y VLAN.....	15
2.2.5.1. Configuración de una VLAN.....	17
2.2.5.2. Diferencias entre las LAN tradicionales y las VLAN.....	17
2.2.5.3. Uso de tramas en las VLAN.....	19
2.2.5.4. VLAN estáticas y VLAN dinámicas.....	21
2.2.5.5. VLAN mejoran la seguridad en la red.....	23
CAPÍTULO 3. SISTEMAS DE ENTRENAMIENTO.....	24
3.1. Introducción.....	24
3.2. Historia de los sistemas de entrenamiento.....	25
3.3. Interacción humano-computadora.....	27
3.3.1. ¿Qué es la interacción humano-computadora.....	28
3.3.2. ¿Quién está involucrado en el diseño de la interacción humano-computadora?.....	29

3.4. Interfaces de usuario.....	29
3.5. Diseño educativo	30
3.5.1. ¿Qué es el diseño educativo?.....	31
3.5.2. Aprendizaje individual	32
3.6. Ventajas e inconvenientes del software de aprendizaje.....	33
3.6.1. Ventajas.....	33
3.6.2. Desventajas	33
3.7. Características de un sistema de entrenamiento de administración de conmutadores CISCO.....	34
CAPÍTULO 4. TRABAJO PROPUESTO	35
4.1. Descripción del proyecto	35
4.1.1. Introducción	35
4.1.2. Comparación entre conmutador real y la herramienta de simulación.....	36
4.1.2.1. Economía.....	37
4.1.2.2. Facilidad de obtención e instalación.....	37
4.1.2.3. Movilidad.....	37
4.1.2.2. Portabilidad.....	38
4.1.3. Herramientas utilizadas.....	38
4.1.3.1. Visual Basic .NET.....	38
4.1.3. Funciones de la herramienta de simulación.....	39
4.1.4. Descripción de los comandos de la herramienta de simulación	43
4.1.4.1. Lectura de caracteres.....	43
4.2. Estructura del diseño	48
4.2.1. Análisis del diseño.....	48
4.2.2. Descripción del diseño	48
4.2.3. Estructura de la aplicación	49
4.2.3.1. Capa Modelo	50
4.2.3.2. Capa Vista.....	54
4.2.3.3. Capa Control.....	54
4.3. Conclusiones	48
CAPÍTULO 5. PRUEBAS REALIZADAS	57
5.1. Introducción.....	57
5.1.1. Resultados de la encuesta del Anexo A.....	57
5.1.2. Resultados de la encuesta del Anexo B.....	62
5.1.2.1. Funcionalidad	62
5.1.2.2. Usabilidad.....	63
5.1.2.3. Evaluación del aspecto de funcionalidad.....	64
5.1.2.4. Evaluación del aspecto de usabilidad.....	68
CAPÍTULO 6. CONCLUSIONES	72
6.1. Aportaciones.....	73
6.2. Limitaciones.....	73
6.3. Trabajos futuros.....	74

ANEXO A. SONDEO ACERCA DEL USO DE CONMUTADORES	75
ANEXO B. CUESTIONARIO DE USABILIDAD Y FUNCIONALIDAD..	77
ANEXO C. PRÁCTICA DE PRUEBA.....	79
ANEXO D. LICENCIA GPL.....	82
Referencias bibliográficas	88
Vita.....	92

Lista de Figuras

Figura 2.1. Un conmutador LAN enlaza un segmento Ethernet de 10 Mbps con un Ethernet de 100 Mbps [4]	10
Figura 2.2. Segmentación de una LAN [5, sec. 2.2.2]	12
Figura 2.3. Segmentación de LAN con conmutadores [5, sec. 2.2.5]	13
Figura 2.4. Interfaz Ethernet/802.3 [5, sec. 2.1.2]	13
Figura 2.5. Colisiones en la tecnología CSMA/CD [5, sec. 2.1.2]	14
Figura 2.6. Ancho de banda para multimedia [5, sec. 2.1.4]	15
Figura 2.7. Conmutación simétrica [5, sec. 2.3.7]	16
Figura 2.8. Conmutación asimétrica [5, sec. 2.3.7]	17
Figura 2.9. VLAN y sus límites físicos [5, sec. 3.2.2]	18
Figura 2.10. Eliminación de los límites físicos [5, sec. 3.2.2]	19
Figura 2.11. Filtrado en las VLAN [5, sec. 3.2.5]	20
Figura 2.12. Filtrado de tramas [5, sec. 3.2.5]	20
Figura 2.13. Rotulación de tramas [5, sec. 3.2.5]	21
Figura 2.14. VLAN estáticas [5, sec. 3.3.3]	22
Figura 2.15. VLAN dinámicas [5, sec. 3.3.4]	22
Figura 2.16. Mayor seguridad en la red [5, sec. 3.4.3]	23
Figura 3.1. Tractor Phelps	26
Figura 3.2. Escritorio “Magic Cap” [9]	27
Figura 4.1. Topología del laboratorio del semestre 3 de CCNA	39
Figura 4.2. Inicio de la herramienta propuesta basada en menús	42
Figura 4.3. Menú principal de la herramienta propuesta	42
Figura 4.4. Notación gráfica [35]	44
Figura 4.5. Máquina de estados finitos de la herramienta propuesta	47
Figura 4.6. La arquitectura MVC utilizada en la herramienta propuesta	50
Figura 4.7. Espacio de trabajo de la interfaz gráfica de usuario	54
Figura 4.8. Función de la capa Control	55
Figura 5.1. Gráfica de la pregunta. ¿Sabes la función que tiene un simulador?	57
Figura 5.2. Gráfica de la pregunta. ¿Alguna vez has utilizado en algún laboratorio los conmutadores Cisco?	58
Figura 5.3. Gráfica de la pregunta. ¿Conoces algún simulador de conmutadores que te permita ejecutar diferentes comandos similares a los reales?	58
Figura 5.4. Gráfica de la pregunta. ¿Cuáles de las siguientes características crees que sean importantes en un simulador de conmutadores?	59
Figura 5.5. Gráfica de la pregunta. De los siguientes elementos, ¿Cuál(es) crees que son importantes para que se agreguen a un simulador de conmutadores?	60
Figura 5.6. Gráfica de la pregunta. ¿Crees que pueda ser de ayuda para realizar las prácticas del laboratorio de Cisco el simulador de conmutadores?	60
Figura 5.7. Gráfica de la pregunta. ¿Si existiera una herramienta de simulación de conmutadores la utilizarías para practicar los laboratorios de Cisco relacionados a los conmutadores?	61

Figura 5.8. Gráfica de la pregunta. ¿Crees que el utilizar la herramienta de simulación de conmutadores pueda mejorar tu desempeño al realizar las prácticas en el laboratorio?	62
Figura 5.9. Gráfica de la pregunta. “Valida correctamente las entradas de los usuarios.”	64
Figura 5.10. Gráfica de la pregunta. “Actualiza los cambios realizados en cada una de las opciones.”	65
Figura 5.11. Gráfica de la pregunta. “Funcionaron cada uno de los comandos utilizados.”	66
Figura 5.12. Gráfica de la pregunta. “Calificación final en el aspecto funcional.”	66
Figura 5.13. Gráfica de la pregunta. “Pregunta abierta.”	67
Figura 5.14. Gráfica de la pregunta. “Amigable.”	68
Figura 5.15. Gráfica de la pregunta. “Intuitiva.”	69
Figura 5.16. Gráfica de la pregunta. “Estable.”	69
Figura 5.17. Gráfica de la pregunta. “Similitud con un conmutador real.”	70
Figura 5.18. Gráfica de la pregunta. “Pregunta abierta.”	71

CAPÍTULO 1. INTRODUCCIÓN

1.1. Introducción

El desarrollo tecnológico y el creciente intercambio internacional de bienes y capitales representan para las compañías modernas tanto retos como oportunidades. Estos retos y oportunidades son: mejora continua de los procesos de manufactura y producción, mayor penetración de productos y apertura de nuevos mercados, eficientización de la cadena de abastecimientos, mejora de las relaciones con los clientes y tantos otros.

El salir adelante frente a estos retos y oportunidades sólo se puede lograr utilizando las mejores técnicas de administración, producción y mercadotecnia. Cualquier cosa menor haría que la empresa no pueda obtener ventajas estratégicas, y con esta pérdida de competitividad, vendría la disminución en las ventas, en los ingresos, y por supuesto, en las utilidades.

Se sabe que el cambio tecnológico es el principal factor de éxito o falla dentro de las empresas al realizar actividades a largo plazo. Las nuevas tecnologías crean nuevos mercados o sustituyen mercados existentes haciendo obsoletas las tecnologías actuales, así como también, los productos, servicios y procesos de producción relacionados con dichas tecnologías [44].

Una gran cantidad de empresas depende fuertemente de las tecnologías de información. Por esto las empresas resienten la falta de profesionales preparados en áreas de redes computacionales y telecomunicaciones.

Esto significa que las personas que laboran en estas áreas requieren de entrenamiento rápido, efectivo, realista y eficiente. Este entrenamiento debe ser adquirido por los profesionistas mientras laboran en otras actividades.

Por otro lado, las universidades modernas ya no pueden conformarse con graduar muchachos que sepan de memoria los libros de texto pero que tienen nulas habilidades prácticas y peor aún, nula exposición al entorno laboral. El entorno laboral actual exige que los egresados posean habilidades técnicas prácticas y experiencia con el manejo de las tecnologías relacionadas con la producción. Esto implica que las instituciones educativas deben poseer grandes instalaciones de equipo de interconexión muy cercano a aquél que se utiliza en misión crítica de las empresas. Estos equipos son caros y la gran mayoría de las escuelas y universidades no pueden adquirir (y constantemente renovar) equipos de interconexión de redes como enrutadores, conmutadores y otros, que puedan darle servicio a cientos y hasta miles de estudiantes al mismo tiempo.

1.2. Metodología científica

1.2.1. Hipótesis

Con la realización de la herramienta de simulación se llega a la siguiente hipótesis, la cual está sujeta a verificación:

Esta herramienta de simulación ayudará a comprender de una manera fácil por medio de un sistema de entrenamiento el manejo de los conmutadores, así como también adquirir conocimientos en el área de interconexión de redes.

1.2.2. Prueba de hipótesis

La comprobación de la hipótesis se llevará a cabo por el método experimental, este método como se refiere Busquerra [49], se tiene que distinguir entre varios experimentos. El que nos interesa es el experimento confirmatorio básicamente son réplicas de experiencias previas o pilotos. Por medio de este método vamos a poner a prueba el sistema ante una cierta cantidad de alumnos que van a evaluar y comprobar el funcionamiento del sistema.

1.3. Objetivo de la tesis

El principal objetivo de esta tesis es crear una herramienta abierta que funcione como un sistema de entrenamiento útil para ayudar a los estudiantes y los profesionistas a ser mejores en el campo laboral por medio de la constante práctica de cada una de las actividades que son realizadas en la academia de Cisco. El programa de la academia de redes de Cisco es un programa de aprendizaje en línea, provee a los estudiantes las habilidades esenciales de la tecnología del Internet en una economía global. El contenido del programa de la academia de Cisco puede ser accedido a través de la red, se pueden realizar exámenes en línea, realizar diferentes actividades de laboratorio y lo más importante, la preparación para las certificaciones sobre los estándares de la industria [45].

La herramienta de simulación debe ser realizada utilizando diferentes metodologías como es el ciclo de vida del software y el modelo MVC, que aseguran calidad y un ciclo de desarrollo eficiente.

1.4. Definición del problema

Los profesionistas de informática y telecomunicaciones que laboran en las diversas entidades productivas del país deben someterse a un proceso constante de entrenamiento, donde el aprendizaje de nuevas habilidades sea rápido y efectivo.

Los estudiantes de las carreras de informática, electrónica y telecomunicaciones deben adquirir habilidades técnicas que sean lo suficientemente realistas para que los preparen para ser productivos dentro de las empresas en el menor periodo de tiempo posible.

La problemática que encontramos para que puedan ser preparados para ser productivos dentro de las empresas, es el alto costo del equipo de interconexión de redes. Además, este equipo es difícil de conseguir y renovar constantemente por las empresas y universidades.

En ocasiones es difícil que los profesionistas o estudiantes que requieren entrenamiento realista sobre equipo de interconexión de redes puedan tener acceso a alguno, ya sea por que el equipo es insuficiente o no está disponible cerca del lugar de entrenamiento.

Para resolver esta problemática, se propone crear una herramienta de simulación capaz de efectuar entrenamiento realista relacionado con la interconexión de redes. Con la herramienta de simulación los profesionistas y estudiantes podrán aprender a configurar los conmutadores, así, cada uno de los profesionistas y estudiantes de universidades podrán contar con un aprendizaje al alcance de sus manos.

1.5. Alcance de la tesis

El alcance en esta tesis se muestra con el desarrollo de una herramienta de simulación de conmutadores con los comandos básicos de configuración. Esta herramienta es útil tanto para el estudiante como para el profesionista porque les facilita el aprendizaje y les brinda el conocimiento necesario para poder operar un conmutador real. Para entender un poco más cual es el alcance de esta tesis se muestra el programa de la academia de Cisco y donde sería utilizada exactamente esta herramienta de simulación.

- **Semestre 1: Fundamentos de redes.** Introduce los conceptos básicos de computación, electrónica y la descripción del modelo OSI con cada una de sus capas.
- **Semestre 2: Enrutadores e IP.** Introduce la relación que existe entre la funcionalidad y la operación de los enrutadores, además de lo relacionado con IP.
- **Semestre 3: Diseño de LAN's.** Introduce el diseño y administración de LAN's, redes virtuales, protocolo IGRP, IPX de Novell.
- **Semestre 4: Diseño de WAN's.** Introduce el diseño y administración de WAN's, protocolo PPP, ISDN y Frame Relay.

Esta herramienta abierta de simulación está relacionada con las prácticas del **tercer semestre** de CCNA en la configuración de redes virtuales, así como todo lo relacionado con el control y configuración de ciertas funciones de los conmutadores.

1.6. Motivación

La motivación para realizar esta tesis está basada en ayudar a los estudiantes y los profesionistas a desenvolverse en la rama laboral, por medio del desarrollo de una herramienta de simulación que les permita efectuar entrenamiento realista acerca de la interconexión de redes. Esta herramienta es útil para aprender a configurar los conmutadores.

Con este trabajo, los estudiantes y profesionistas se beneficiarán en el área de interconexión de redes, porque con el uso de esta herramienta de simulación, serán capaces de aprender a través de un proceso de aprendizaje de una manera más accesible y dinámica, las empresas y las universidades no necesitarán contar con el equipo físico para poder realizar prácticas o entrenamiento sobre la configuración de conmutadores.

Otro motivo es el hecho que la herramienta de simulación es desarrollada para poder ser modificada a beneficio de cada una de las personas por medio del software “open source”.

1.7. Organización de la tesis

En el Capítulo 2 se describen los temas que son importantes para la realización, así como la revisión bibliográfica relacionada con el desarrollo de la tesis. En este capítulo, se explica lo relacionado con el ciclo de vida y la estructura de datos, útil para la realización de la herramienta, así como cada una de sus características. También se explica el funcionamiento de los conmutadores y las diferentes maneras en que pueden ser utilizados.

En el Capítulo 3 se describe la interacción humano-computadora. También se habla acerca del diseño educativo y su importancia en el desarrollo del software relacionado con la educación, así como la explicación de las teorías del aprendizaje que justifican la utilización de la herramienta de simulación.

En el Capítulo 4 se describe la herramienta de simulación con la principal utilización de software libre u “open source”, que implica que las personas puedan modificar el software propuesto para satisfacer sus necesidades específicas. Además, se realiza una comparación entre la herramienta de simulación y los conmutadores reales, así también, se realiza una descripción de los comandos utilizados y la manera de leer cada uno de ellos utilizando la máquina de estados finitos. También se muestran las metodologías utilizadas para poder

obtener una herramienta útil siguiendo cada uno de los pasos requeridos para alcanzar un producto final.

En el Capítulo 5 se describen las pruebas realizadas a la herramienta de simulación con el fin de conocer si la aplicación es óptima y útil para que los estudiantes y los profesionistas la utilicen al realizar sus prácticas antes de llegar al laboratorio.

Por último, en el Capítulo 6 se concluye el trabajo realizado en esta tesis, mostrando las aportaciones generadas, las limitaciones de la herramienta de simulación y los trabajos futuros que pueden ser realizados sobre el área de investigación.

CAPÍTULO 2. MARCO TEÓRICO

2.1. Ciclo de vida

2.1.1. Introducción

Debido al creciente desarrollo del software educativo durante los últimos años, gran parte del mismo ha sido elaborado de forma desorganizada y poco documentada, y considerando el gran crecimiento que tendrá, surge la necesidad de crear una metodología detallada y disciplinada para su desarrollo, mediante los métodos y herramientas, procedimientos y herramientas, que provee la ingeniería de software para construir programas educativos de calidad, siguiendo todo lo relacionado a las teorías educativas y la comunicación adyacente [36].

El software educativo, tiene características particulares en cuanto a comunicación [37] con el usuario, las cuales no se pueden cuantificar mediante métricas porque están relacionadas con conductas de aprendizajes. Pero, las reglas de la construcción de un programa son las mismas ya sea educativo, comercial, de investigación, etc.

Para ese creciente desarrollo del software educativo se requiere presentar una solución informática para el diseño, desarrollo y evolución tanto interna como externa.

2.1.2. ¿Qué es el ciclo de vida?

El ciclo de vida es un elemento importante para la realización de una aplicación, por que ayuda a evitar la inexistencia de marcos teóricos dentro de la realización de software educativo.

Algunos autores como Galvis [38] reconocen la necesidad de un marco de referencia, teniendo en cuenta que se debe lograr la satisfacción de los requisitos en las diversas etapas del desarrollo.

Marqués [39], es uno de los autores que plantea un ciclo de desarrollo para software educativo de programas en 10 etapas, en las cuales se describe de una manera detallada las actividades y recursos necesarios para cada una de ellas.

Se denomina ciclo de vida a toda la vida del software, comenzando con su concepción y finalizando en el momento de la desinstalación del mismo. Un ciclo de vida establece el orden de las etapas del proceso de software y los criterios que se deben tomar en cuenta para poder pasar de una etapa a la siguiente.

El tema del ciclo de vida es importante por que existe un seguimiento de cada una de las etapas que se llevan acabo, algunas organizaciones profesionales y organismos internacionales como la IEEE (Institute of of Electrical and Electronics Engineers) y la ISO/IEC (International Standards Organization/International Electrochemical Commission), han publicado normas tituladas “Standard for Developing Software Life Cycle Processes” (Estándar IEEE para el desarrollo de procesos del ciclo de vida del software) [40] y “Software life-cycle process” (Proceso de ciclo de vida del software) [41].

2.1.3. Estructura de Datos

La aparición de las estructuras de datos fue un respiro para la programación, alrededor de los años 50's, la programación era más difícil que en estos momentos, consistía en programas demasiados extensos y en código binario, mejor conocido como “lenguaje máquina”.

Con el paso del tiempo se fueron introduciendo diferentes conceptos como la estructura de datos, los tipos de datos y la abstracción de datos, probablemente los más utilizados por los programadores en estos momentos son los dos primeros, aunque la elección de cada concepto depende mucho de que se programe.

Las estructuras de datos han sido utilizadas por muchos años desde que se comenzó a almacenar programas de computadoras. Las primeras estructuras de datos en ser utilizadas fueron los “arreglos” de información y su característica es mantener los datos almacenados de una manera secuencial. La clave de la estructura de datos es la introducción de enlaces a otros elementos de datos. De hecho, esto es una manera importante de construir estructuras de datos complejas [26].

2.1.3.1 ¿Qué es una estructura de datos?

Estructura de datos es una representación de la relación lógica entre los datos de elementos individuales. En la arquitectura del software la estructura de datos es tan importante como la estructura del programa [27]. Ahora bien, *un tipo de datos es un tipo compuesto definido por el usuario* [28].

Las propiedades con las que cuentan las estructuras de datos son las siguientes [26]:

- Estática.
- Dinámica.
- Representación.
- Asignación y des-asignación de los campos de almacenamiento.
- Densidad.

2.2. Conmutadores

2.2.1. Introducción

Los conmutadores son dispositivos de comunicación de datos que principalmente se encuentran operando en la capa 2 del modelo OSI, por lo tanto uno se puede referir a ellos como un dispositivo de la capa de enlace de datos [4]. La conmutación existe en la capa 2, que es utilizada por los conmutadores para enviar tramas y está basado en direcciones MAC. La conmutación en la capa 3 que es utilizada por los enrutadores para encaminar un paquete y está basada en información de la capa de red. La diferencia entre la conmutación de la capa 2 y la de la capa 3 es la información contenida dentro de la trama, la cual es utilizada para determinar la interfaz de salida correcta [5, sec. 2.3.3].

Al hablar de la capa 2 del modelo OSI existen principalmente dos dispositivos: los conmutadores y los puentes. En este capítulo se hablará sobre los conmutadores. En ocasiones, los conmutadores son definidos como puentes multi-puerto.

Con el paso del tiempo la tecnología de conmutación ha surgido como una evolución de los puentes tomando en cuenta que son superiores en cuanto al funcionamiento del desempeño, alta densidad de puertos, bajo costo por puerto y la mayor flexibilidad. Todo esto ha contribuido para que la tecnología de conmutación emerja como reemplazo de la tecnología de reenvío entre redes y como un complemento de las tecnologías de enrutamiento [4].

Los conmutadores cuentan con una ventaja principal que es la transparencia que existe en los protocolos de capas superiores, por lo que no requieren examinar información de capas superiores. Además cuentan con filtros que pueden ser utilizados para difundir y multi-difundir los paquetes que no son necesarios, así como también para disminuir el tráfico que es presentado por los dispositivos que se encuentran conectados en todos los segmentos. *Los conmutadores actúan como una "firewall" para algunos errores potencialmente perjudiciales en la red y establecen la comunicación entre un gran número de dispositivos que serán soportados en una LAN sencilla conectados a un puente [4].*

Los conmutadores son generalmente utilizados para segmentar una LAN. También pueden ser utilizados para conectar diferentes medios de una LAN, por ejemplo, una LAN de 10 Mbps Ethernet y una LAN de 100 Mbps Ethernet.

Por otra parte, los conmutadores reducen el número de colisiones en un segmento de red al proveer ancho de banda dedicado en cada segmento de la red.[4]

2.2.2. Tipos de conmutadores

Algunos dispositivos dentro de la red pueden llegar a causar latencia, o mejor conocido como retraso, es decir, el tiempo que le toma a una trama o paquete viajar desde la fuente hasta el destino. Dentro de las LAN es importante cuantificar la latencia, las cuales deben de ser capaces de detectar las colisiones y negociar las transmisiones de forma adecuada.

La latencia puede ser producida por tres componentes principales [5, sec. 2.1.5]:

1. **Tiempo de pulsos:** Es el tiempo que le lleva a la NIC emisora colocar pulsos de voltaje y el tiempo que le lleva a la NIC receptora traducir esos pulsos.
2. **Retardo de propagación:** Es generado cuando la señal tarda cierto tiempo en viajar a lo largo del cable. Por supuesto, entre más largo sea el cable, mayor va a ser el retardo de propagación. Otro factor que influye para que se alcance el retardo de propagación es la velocidad nominal de propagación.
3. **Latencia adicional:** Es producida por todos aquellos dispositivos entre las capas 1 y 3 que se encuentran entre las dos computadoras que están intercambiando información.

Aunque se ha mencionado anteriormente que la latencia se debe a la cantidad de dispositivos que se encuentran a lo largo de la red y a la distancia que existe entre los dispositivos que se encuentran en comunicación, otro factor que puede influir en la latencia es el hecho que se utilicen enrutadores. Los enrutadores realizan toma de decisiones más complejas procesando más datos de encapsulamiento que los conmutadores.

Para corregir el problema de la latencia producida por los errores antes mencionados, los conmutadores utilizan diferentes técnicas de reenvío, estas son: “store and forward switching” (conmutación almacenar-reenviar) y “cut-through switching” (conmutación directa).

En “store and forward switching”, la trama debe ser recibida completamente para después ser enviada, por lo tanto se entiende que la latencia varía dependiendo del tamaño con que cuenta la trama¹ [5, sec. 2.3.2]. Esta técnica es útil por que puede evaluar la trama de manera completa por algún posible error antes de que sea enviada mientras se completa la recepción. Esta es una de las ventajas de los conmutadores sobre los hubs.

En “cut-through switching” permite que el conmutador comience a enviar la trama una vez que se ha recibido suficiente de la misma para tomar una decisión de a donde debe ser enviado, así es como puede ser reducida la latencia a través del conmutador. “Fast-forward” y “fragment-free” son dos formas de conmutación del “cut-through” [4].

¹ La latencia también depende de la marca del conmutador y el tipo de conmutación utilizado

“Fast-forward” cuenta con el nivel más bajo de latencia al enviar el paquete inmediatamente después de que la dirección fuente es recibida; es por esto que algunas veces el mensaje contiene errores, para evitar esto es recomendable utilizar la opción de “fragment-free” con el fin de reducir el número de paquetes con errores. En este método, la latencia es medida desde el primer bit recibido, hasta el último bit transmitido, primero en entrar, primero en salir (FIFO, en sus siglas en inglés).

“Fragment-free” filtra los errores de los paquetes, antes que se comiencen a enviar. Para que el funcionamiento sea el más adecuado, los fragmentos de colisión deben de ser más pequeños que 64 bytes. Cualquier paquete mayor de 64 bytes es válido y recibido sin errores. En este método al igual que “fast-forward” la latencia es medida en FIFO [5, sec. 2.3.9].

Para conocer la latencia es necesario considerar la manera en la cual el conmutador reenvía cada uno de las tramas. Para obtener mayor rapidez, el conmutador toma menor tiempo al momento de verificar los errores.

Existen muchos tipos de conmutadores, como son los conmutadores ATM, conmutadores LAN y diferentes tipos de conmutadores WAN.

2.2.2.1. Conmutadores LAN

Los conmutadores LAN son utilizados para interconectar diferentes segmentos de LAN. Este tipo de conmutadores proveen comunicación libre de colisiones entre dispositivos de red con soporte para múltiples conversaciones. Los conmutadores LAN están diseñados para conmutar datos a grandes velocidades.

En la figura 2.1 se muestra una red simple en donde se encuentran interconectadas una LAN Ethernet de 10 Mbps y una LAN Ethernet 100 Mbps.

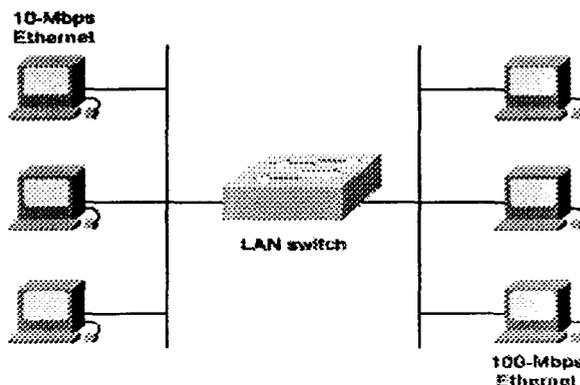


Figura 2.1. Un conmutador LAN enlaza un segmento Ethernet de 10 Mbps con un Ethernet de 100 Mbps [4].

2.2.3. Operaciones básicas de un conmutador

A medida que va pasando el tiempo, los hubs han sido sustituidos por los conmutadores los cuales pueden actuar sin provocar tráfico en las redes de cables ya existentes.

A pesar de que los conmutadores realizan más operaciones que los puentes, no son dispositivos complicados. Los conmutadores realizan diferentes funciones como [4][5, sec. 2.3.1]:

- Controlar el flujo de datos.
- Manejar los errores de transmisiones.
- Proveer direccionamiento físico.
- Administrar el acceso a los medios físicos.
- Analizar las tramas entrantes.
- Tomar decisiones basándose en la información contenida en las tramas.
- Llevar las tramas a su destino. En algunos casos la dirección del destino se encuentra contenida en cada trama.
- Conmutar tramas de datos.
- Mantener las operaciones de conmutación.

2.2.4. Segmentación de LAN

Cuando se segmenta una red se divide en pequeñas unidades, esas unidades son llamadas segmentos las cuales utilizan el método de acceso CSMA/CD. Este método es un mecanismo de acceso a medios mediante el cual los dispositivos que están listos para transmitir datos primero verifican el canal en busca de una portadora. El dispositivo puede transmitir si no se detecta ninguna portadora durante un período de tiempo determinado. Si dos dispositivos transmiten al mismo tiempo, se produce una colisión que es detectada por todos los dispositivos que colindan. Esta colisión subsecuentemente demora las retransmisiones desde esos dispositivos durante un período de tiempo de duración aleatoria. El acceso CSMA/CD es utilizado por Ethernet e IEEE 802.3. [5, sec. 2.2.2].

En la figura 2.2 se muestra una red Ethernet segmentada. La red completa posee 15 computadores (6 servidores de archivos y 9 PC). Al utilizar segmentos en una red, menos usuarios ó dispositivos comparten los mismos 10 Mbps al comunicarse entre sí dentro del segmento [5, sec. 2.2.2].

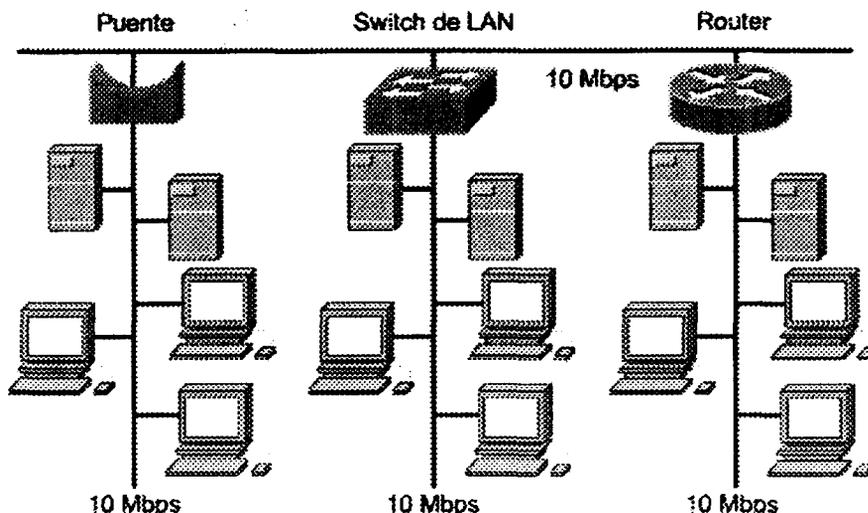


Figura 2.2. Segmentación de una LAN [5, sec. 2.2.2].

2.2.4.1. Ventajas y desventajas de la segmentación de LAN con conmutadores

La conmutación LAN cuenta con algunas ventajas y desventajas cuando son utilizadas por los conmutadores. Las principales ventajas son las siguientes [5, sec. 2.2.5]:

- Compensar la escasez del ancho de banda y los cuellos de botella de la red, como los que pueden suceder entre varias computadoras y un servidor de archivos.
- Tener la capacidad de segmentar una red en micro-segmentos, como se muestra en la figura 2.3. Estos segmentos son de una sola computadora.
- Tener la capacidad de eliminar los dominios de colisión, pero todas las computadoras permanecen conectados al conmutador bajo el mismo dominio de difusión.
- Ser visto como un puente muy rápido multi-puerto. Cada puerto del conmutador se convierte en un dominio de colisión por separado y no serán propagados a otros puertos.

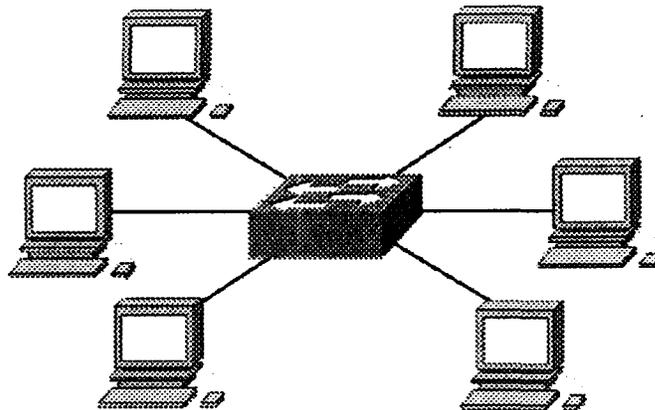


Figura 2.3. Segmentación de LAN con conmutadores [5, sec. 2.2.5].

Como principal desventaja, los conmutadores son más caros que los hubs. Muchas empresas han ido implementando poco a poco el cambio de una hub a un conmutador, hasta que llegue el momento que sean completamente reemplazados.

2.2.4.2. Problemas de comunicación en las LAN

La arquitectura más común dentro de las redes LAN es Ethernet, esta tecnología se utiliza para transportar información a través de los diferentes dispositivos que se encuentran conectados, como computadoras, servidores, impresoras. Ver figura 2.4.

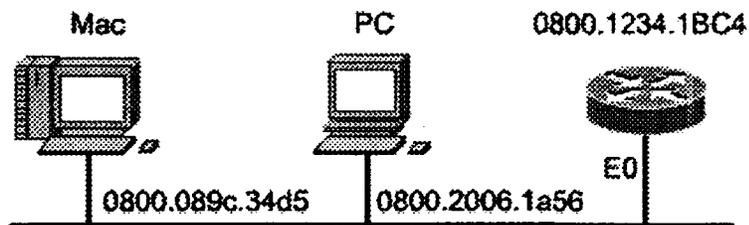


Figura 2.4. Interfaz Ethernet/802.3 [5, sec. 2.1.2].

En esta arquitectura, se utiliza el método de difusión en donde se transmiten y se reciben datos de todos los nodos por ser un medio compartido. Por ser un medio compartido puede contar con problemas severos como lo son [5, sec. 2.1.2]:

- La forma de entrega de las tramas de las LAN Ethernet/802.3.
- (CSMA/CD) permite que solo un equipo transmita a la vez.

- Las aplicaciones multimedia que requieren gran ancho de banda, sumados a la naturaleza de la arquitectura Ethernet, pueden llegar a causar una congestión en la red.
- La latencia que es producida a medida de que los datos viajan a través de las capas de networking 1, 2, 3, así como la que es producida por los repetidores que existen a lo largo de la red.

Uno de los problemas inherentes a la tecnología CSMA/CD son las colisiones, esto se puede ver en la figura 2.5.

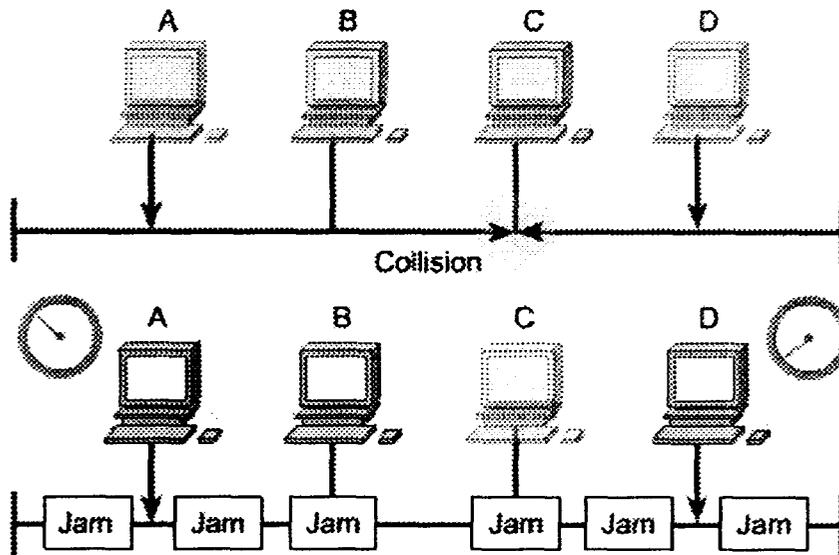


Figura 2.5. Colisiones en la tecnología CSMA/CD [5, sec. 2.1.2].

Como se mencionó anteriormente, las aplicaciones multimedia requieren de un gran ancho de banda, superiores a los 10 Mbps disponibles en las LAN Ethernet/802.3 compartidas. Los tiempos han cambiado y eso se puede observar en la figura 2.6. A lo largo de la era de las computadoras ha crecido la necesidad de transmitir archivos gráficos de gran capacidad, videos, imágenes, etc. Además el crecimiento de usuarios en la red ha llegado a producir grandes congestiones en la red, dando como resultado tiempos de respuesta muy lentos y transferencia de archivos lentos. Para poder reducir la congestión, es necesario contar con un ancho de banda mayor o hacer un buen uso del ancho de banda con el que se cuenta [5, sec. 2.1.4].

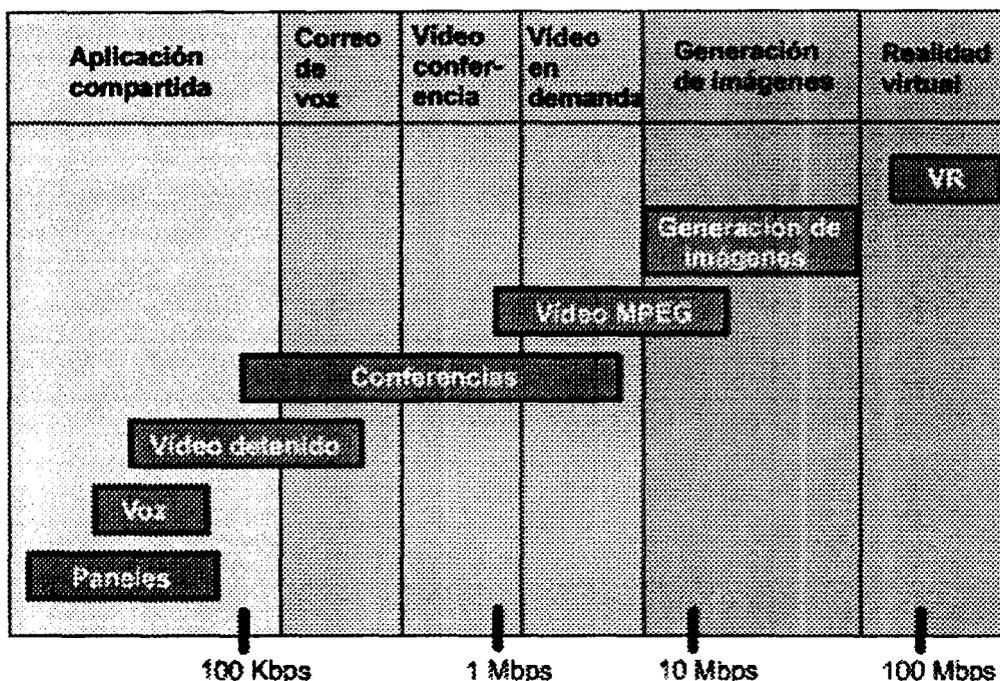


Figura 2.6. Ancho de banda para multimedia [5, sec. 2.1.4].

2.2.5. Conmutación en la capa 2 y VLAN

Como se mencionó anteriormente los conmutadores utilizan la conmutación de la capa 2 para enviar tramas y lo hacen por medio de las direcciones MAC, así mismo, construyen y mantienen las tablas de conmutación que contienen las direcciones MAC que pertenecen a cada puerto o interfaz.

La conmutación de la capa 2 no observa dentro del paquete para obtener la información de la capa de red, sino ve dentro de una trama la dirección MAC del destino. Si de alguna manera no conociera el destino de la trama, envía un mensaje de difusión con la trama a todos sus puertos para aprender el destino correcto, cuando la trama es regresada, el conmutador aprende la localización de la nueva dirección y la agrega a su tabla [5, sec. 2.3.3].

Una Ethernet conmutada, como su nombre lo dice, se basa en una interfaz tipo Ethernet. *Cada nodo está conectado a uno de sus puertos o a un segmento conectado a uno de los puertos del conmutador.* Cuando una computadora se encuentra conectada directamente a un conmutador tiene su propio dominio de colisión y como consecuencia cuenta con todo el ancho de banda de 10 Mbps. A esto se le conoce con el nombre de LAN de Ethernet conmutada comportándose como si solamente existiera un emisor y un receptor, comparte el mismo ancho de banda entre sí, efectuándose una transmisión de datos sin ningún

problema. Es por eso que una LAN de Ethernet conmutada proporciona un mejor rendimiento en el ancho de banda que una LAN Ethernet conectada a un puente o a un hub.

Un conmutador Ethernet puede aprender las direcciones dinámicamente de los dispositivos en la red leyendo las direcciones fuentes de cada paquete transmitido agregándolas a sus tablas. Estas direcciones son aprendidas y almacenadas en el contenido de memoria direccionable (CAM, en sus siglas en inglés) [5, sec. 2.3.5].

Dentro de la conmutación de la capa 2 existe la conmutación simétrica y la conmutación asimétrica. La conmutación simétrica se caracteriza por proporcionar el mismo ancho de banda a cada uno de los puertos del conmutador, tales como puertos de 10 Mbps o puertos de 100 Mbps [5, sec. 2.3.7]. Ver figura 2.7.

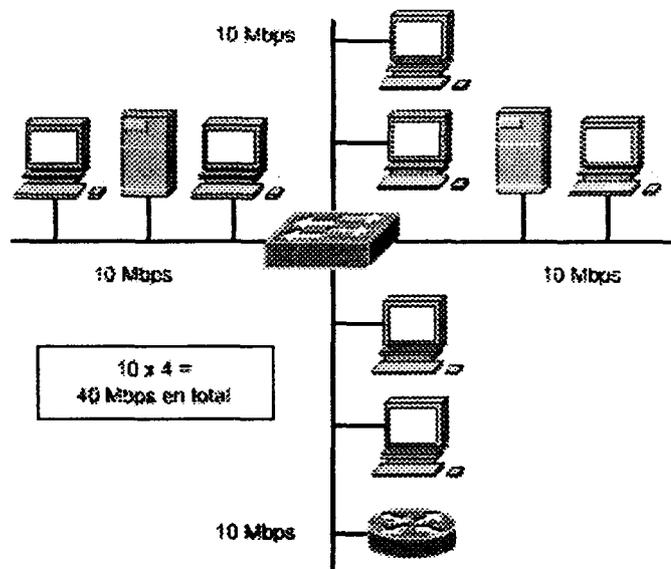


Figura 2.7. Conmutación simétrica [5, sec. 2.3.7].

La conmutación asimétrica cuenta con una proporción de puertos con distinto ancho de banda entre los 10 Mbps y los 100 Mbps. Este tipo de conmutación tiene la características de colocar el mayor ancho de banda donde se encuentran los servicios importantes dentro de la red, para evitar cuellos de botella cuando los clientes quieran acceder a ese servicio [5, sec. 2.3.7]. Esto se puede ser visto en la figura 2.8.

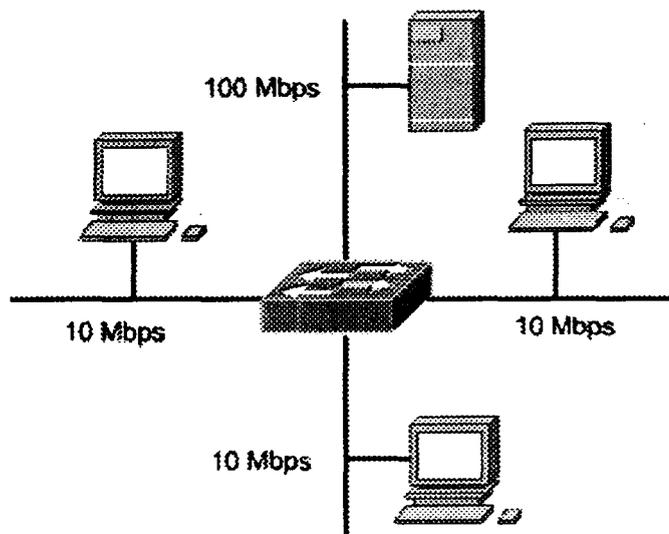


Figura 2.8. Conmutación asimétrica [5, sec. 2.3.7].

2.2.5.1. Configuración de una VLAN

Una VLAN se define como una agrupación lógica de dispositivos que se encuentran conectados a lo largo de la red. Cuando se agrupan los dispositivos dentro de una VLAN, se realiza la configuración sin importar la ubicación física de los mismos. Las VLAN son consideradas como subredes dentro de una organización. La configuración de una VLAN se realiza en los conmutadores por medio del software, además, debe ser configurada de acuerdo a la arquitectura física en la que se encuentra conectada. Por lo general una VLAN debe de seguir el estándar IEEE 802.1q, pero puede llegar a variar [5, sec. 2.3.10][5, sec. 3.1.1].

2.2.5.2. Diferencias entre las LAN tradicionales y las VLAN

Cuando se comparan las LAN tradicionales contra las VLAN resulta que las últimas son una manera mucho más económica de poder agrupar los usuarios en grupos de trabajo. En la figura 2.9 se puede ver las diferencias que existen entre ellas, en donde el enrutador que conecta cada hub llega a proporcionar segmentación y algunas veces puede actuar como una “firewall” de la difusión, en cambio, los segmentos creados por los conmutadores no. Las LAN tradicionales no agrupan a los usuarios de acuerdo a su grupo de trabajo, es por eso que comparten el mismo segmento.

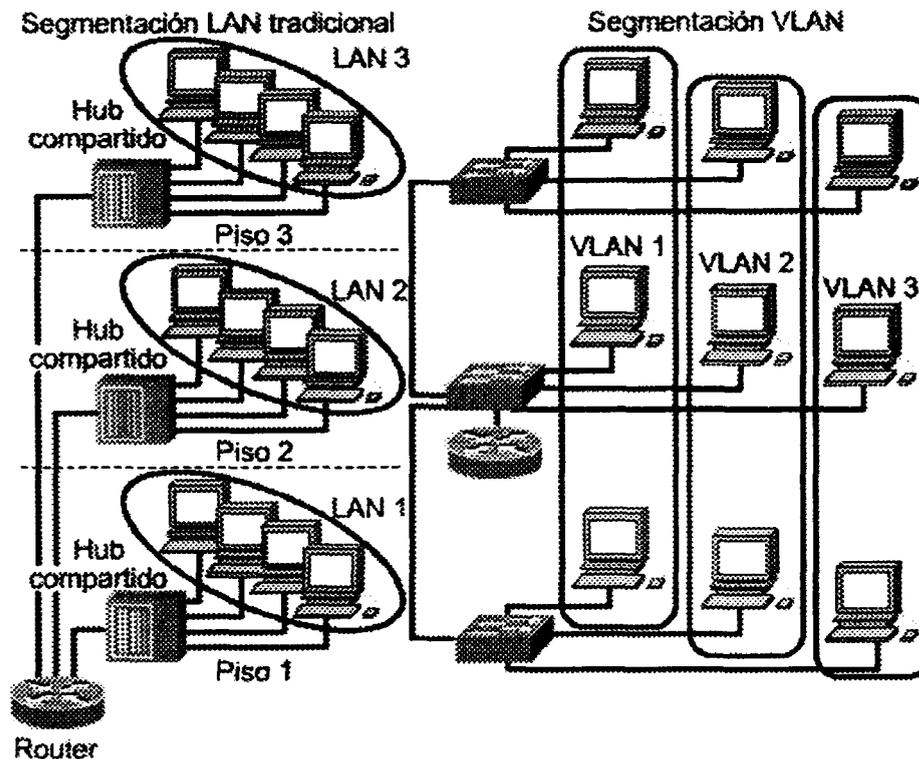


Figura 2.9. VLAN y sus límites físicos [5, sec. 3.2.2].

Algunas de las diferencias principales se mencionan a continuación [5, sec. 3.2.2]:

- Las VLAN funcionan a nivel de Capa 2 y Capa 3 del modelo OSI.
- La comunicación entre VLAN es implementada por la Capa 3.
- Las VLAN utilizan métodos para el control de difusión en la red.
- La asignación de usuarios a una VLAN la realiza el administrador de la red.
- Con las VLAN se puede agregar seguridad, al definir qué nodos son los que se pueden comunicar entre sí.

Por medio de las VLAN, los puertos de los conmutadores así como sus usuarios pueden ser agrupados de diferentes maneras, como se muestra en la figura 2.10.

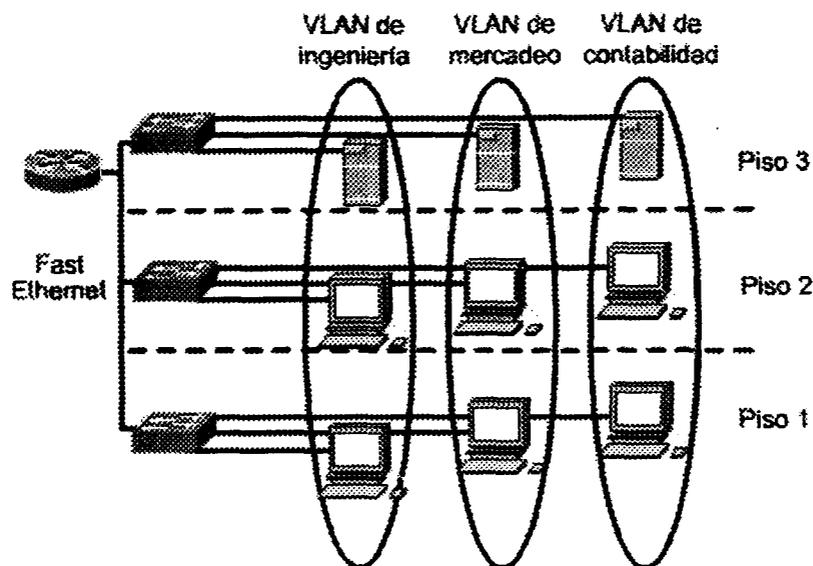


Figura 2.10. Eliminación de los límites físicos [5, sec. 3.2.2].

A continuación se muestran algunos grupos de trabajo definidos que se pueden utilizar:

- Compañeros de trabajo del mismo departamento.
- Equipos de producción
- Grupos de usuarios que comparten aplicaciones de red o software.

Al agrupar tanto los puertos como usuarios con uno o más conmutadores, las VLAN pueden abarcar diferentes infraestructuras conectadas dentro de un edificio o entre edificios, inclusive redes de área amplia (WAN, en sus siglas en inglés).

2.2.5.3. Uso de tramas en las VLAN

Como se sabe, los conmutadores son muy importantes en el desarrollo de las VLAN. Los conmutadores toman decisiones por medio del filtrado y el envío de tramas, basándose en las métricas de VLAN. El conmutador tiene la capacidad de poderse comunicar con otros conmutadores y enrutadores dentro de la misma red. Esto puede ser observado en la figura 2.11.

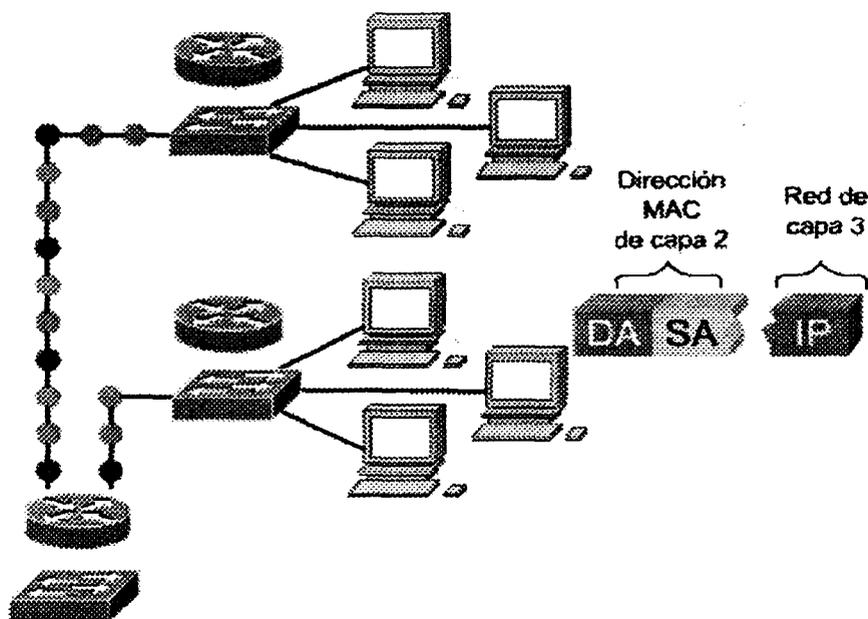


Figura 2.11. Filtrado en las VLAN [5, sec. 3.2.5].

Se utilizan dos técnicas para examinar las tramas cuando es recibido o enviado a través del conmutador, una es acerca del filtrado de tramas y la otra es sobre la identificación de las tramas. Con estas técnicas se determina a donde deben ser enviados, filtrados o difundidos. Estos mecanismos son fáciles de implementar a través de toda la red.

Específicamente el filtrado de tramas examina información de cada una de las tramas. Cada conmutador mantiene una tabla de filtrado en donde son almacenados muchos atributos de cada trama, proporcionando un alto nivel de control administrativo, tal cual se puede observar en la figura 2.12. Los usuarios se encuentran agrupados según las direcciones (MAC) de una computadora, posteriormente el conmutador compara las tramas con cada una de las direcciones de la tabla y sus decisiones se basan respecto a las entradas.

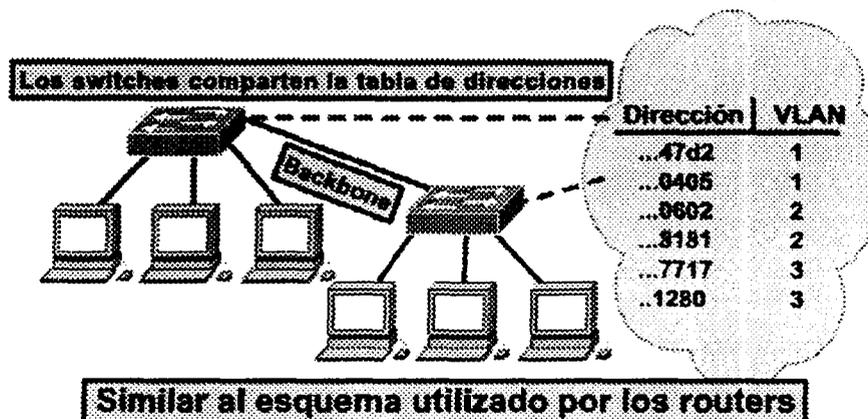


Figura 2.12. Filtrado de tramas [5, sec. 3.2.5].

Anteriormente las VLAN utilizaban un modelo nada escalable basado en filtros y los usuarios eran agrupados por medio de una tabla de filtrado. Cada trama cuenta con un identificador exclusivo, esos identificadores son asignados por el administrador del conmutador en la configuración del mismo. Cuando se asignan etiquetas a cada trama se está ofreciendo una solución más escalable para la implantación de VLAN por medio del estándar IEEE 802.1q.

Cada vez que la trama pasa por un conmutador, es examinada antes de que se realice alguna operación de difusión o transmisión a los demás conmutadores, enrutadores o cualquier dispositivo que se encuentra en la misma red. La identificación de la trama solo ocurre a nivel de la Capa 2 y requiere poco procesamiento o gasto administrativo mínimo. Esto podemos observarlo en la figura 2.13 [5, sec. 3.2.5].

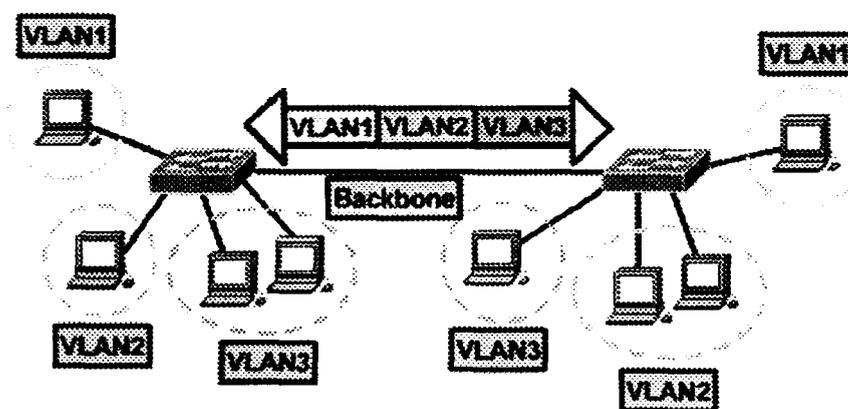


Figura 2.13. Rotulación de tramas [5, sec. 3.2.5].

2.2.5.4. VLAN estáticas y VLAN dinámicas

En las VLAN estáticas, los puertos son asignados desde un principio, estos puertos permanecen configurados hasta que se produzca algún cambio en ellos, generado por el administrador de la red. Las VLAN estáticas cuentan con la característica de que son redes seguras, fáciles de configurar y monitorear. Su mejor desempeño lo realizan en redes controladas o bien administradas [5, sec. 3.3.3]. Las redes estáticas se pueden observar en la figura 2.14.

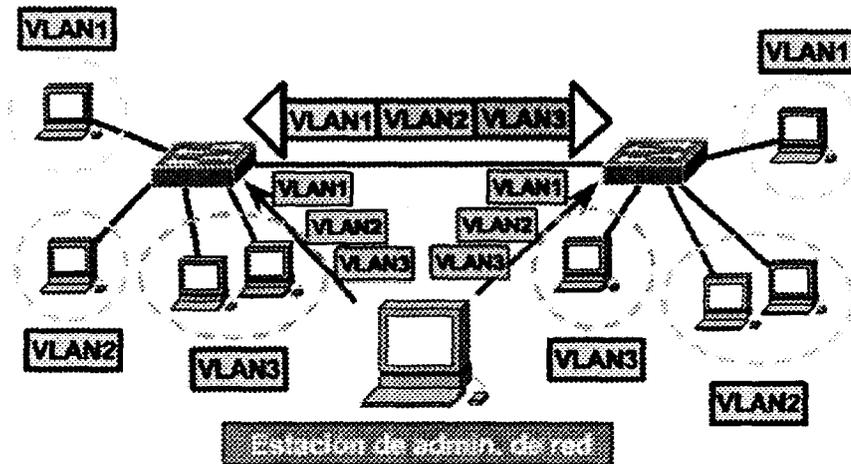


Figura 2.14. VLAN estáticas [5, sec. 3.3.3].

En las VLAN dinámicas, los puertos son asignados de manera automática, por medio de las direcciones MAC, el direccionamiento lógico o el protocolo de los paquetes de datos. Cuando una computadora se encuentra asignada a un puerto sin configuración, inmediatamente el conmutador verifica la dirección MAC de esa computadora y configura el puerto de acuerdo a la VLAN que debe ser asignado. Esta manera de configurar los puertos tiene algunas ventajas o beneficios, entre ellos se puede tomar en cuenta el hecho que se requiere menor administración en el centro de cableado a la hora de agregar o desplazar a algún usuario.

El único inconveniente es que cuando se configura y se mantiene la base de datos por primera vez, requiere de una gran cantidad de administración [5, sec. 3.3.4]. Las redes dinámicas se pueden observar en la figura 2.15.



Figura 2.15. VLAN dinámicas [5, sec. 3.3.4].

2.2.5.5. VLAN mejoran la seguridad en la red

Con el crecimiento de las LAN ha crecido también el interés en mantener seguros los datos confidenciales por medio de la implementación de cuentas de acceso. Las LAN compartidas cuentan con vulnerabilidades, como puertos abiertos, que facilitan el acceso a personas no autorizadas para poder observar todo el tráfico que circula en cierto segmento. Para solucionar este problema se puede implementar una red segmentada en múltiples grupos de difusión permitiendo al administrador de la red las siguientes funciones [5, sec. 3.4.3]:

- Limitar la cantidad de usuarios en una VLAN.
- Evitar que usuarios que no tienen aprobación de conectarse lo hagan, sin que el administrador de la red lo permita.

La realización de las VLAN es simple en todos los aspectos, los puertos pueden ser agrupados de acuerdo al tipo de aplicación que es utilizada o los privilegios de acceso que cuenta cada uno de los usuarios del grupo, las aplicaciones con más riesgo se ubican en un grupo seguro de VLAN. Estos grupos cuentan con restricciones que pueden ser implementadas según las direcciones de las computadoras, tipos de aplicaciones o protocolos. Para mantener más seguro cada uno de los datos que son manejados, se pueden utilizar listas de control de acceso. En la figura 2.16 se puede visualizar a una VLAN segura.

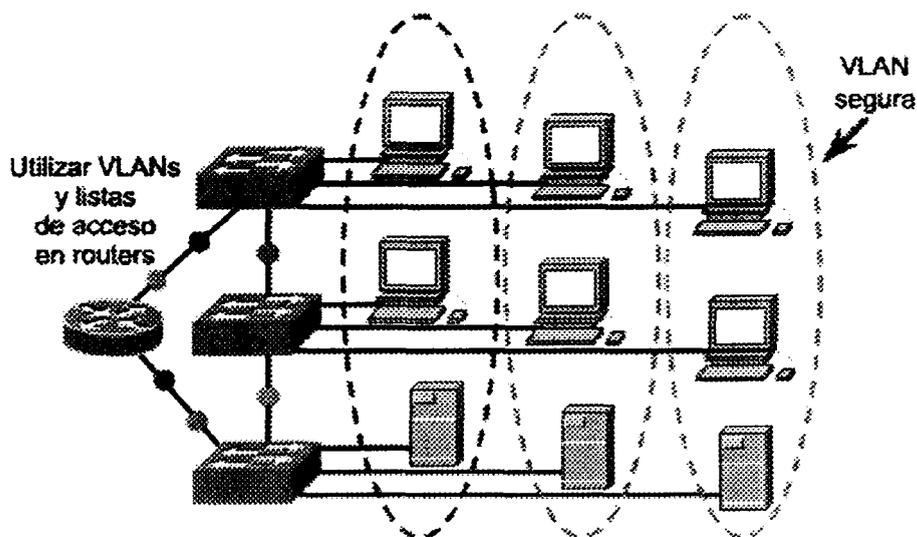


Figura 2.16. Mayor seguridad en la red [5, sec. 3.4.3].

CAPÍTULO 3. SISTEMAS DE ENTRENAMIENTO

3.1. Introducción

Las computadoras, con el paso del tiempo, han ido mejorando en cuanto a capacidad y desempeño. Pero existe un problema; para la mayoría de las personas es muy difícil poder manejar esos aparatos. A pesar de que se vive en un mundo rodeado de ellas no todos están acostumbrados a manejarlas; es por esto que se hace una petición, “Hacer las computadoras más fáciles de utilizar” [16]. Con esta petición se pretende hacer más funcional la computadora para las personas; hacer que hable con ellas, hacer que obtenga la información que requieren, en otras palabras, adaptarlas a las necesidades individuales de cada persona.

El diseño centrado en el humano es más que el cambio de software o hardware en los sistemas de cómputo, lo que se busca es mejorar el uso de la tecnología. Para alcanzar el diseño centrado en el usuario se debe poner especial interés en la relación que existe entre el humano y la computadora.

Actualmente existen tres grandes áreas de cambio de la industria de la computación las cuales se mencionan a continuación. [16]

1. **Velocidad:** Proporcionada por los microprocesadores, para el 2015 su velocidad se incrementará alrededor de 50 veces, proporcionando mayor procesamiento y así como mayor comunicación.
2. **Comunicación inalámbrica:** Por medio de las computadoras del mañana o por los teléfonos celulares pueden las personas controlar su centro de entretenimiento, la seguridad de su casa, entre otras actividades importantes para el humano. En la actualidad la comunicación inalámbrica es casi un hecho, ya se estas utilizando teléfonos celulares con acceso a Internet.
3. **Software:** Algunas veces se requiere contar con la información necesaria para poder continuar con la vida diaria, cuando esto sucede, es necesario contar con la información donde y cuando se requiera, independientemente del equipo que se esté utilizando.

En cuanto al terreno humano, es importante poder satisfacer cada una de sus necesidades con el fin de hacer más fácil el uso de las computadoras. Las personas utilizan dos formas de comunicación para interactuar con otras, esas formas son: ver y hablar. Como usuarios, es más fácil entender lo que pasa alrededor a través de una manera visual, “la visión es como el diálogo, solo que más poderosa” [16].

El diálogo debe ser el principal acercamiento para el intercambio de información entre la computadora y el humano, y la visión debe ser el principal acercamiento para la percepción humana de la información.

Es necesario indicar que las computadoras deben ser funcionales para los humanos, siendo más fáciles de manejar y administrar en cada una de las funciones que realizan. Para los usuarios es muy difícil adaptarse a un nuevo sistema utilizado, para hacer el aprendizaje más fácil debe ser necesaria la implementación de diferentes tecnologías. Debe existir un especial interés en la relación que existe entre el humano y la computadora.

La interacción humano-computadora debe ser considerada desde el inicio del proceso de diseño. La interacción humano-computadora está compuesta por diferentes elementos, los cuales son: *El usuario, la computadora y la interacción.*

Además, en este capítulo se verán temas. Esos temas son acerca de la interacción que existe entre el humano y la computadora, así como quien está involucrado en el diseño de la misma. Se menciona acerca de las interfaces de usuarios en donde existen dos diferentes modelos, el modelo orientado al mecanismo y el modelo orientado a la actividad. Además, dentro del diseño educativo existen una serie de procedimientos para crear entrenamiento, algunos procedimientos se centran en las metas que el usuario conoce. Dentro de la educación existen cuatro aspectos importantes: *aprender a conocer, aprender a hacer y aprender a ser.* Un tema muy importante dentro de este capítulo es acerca de las teorías del aprendizaje. Estas teorías del aprendizaje hablan acerca del conductismo, cognitivismo y constructivismo que son útiles para la construcción del diseño de instrucción.

3.2. Historia de los sistemas de entrenamiento

Pocos años después de que alguna nueva tecnología es lanzada al mercado de la computación, los fabricantes intentan ampliar el mercado construyendo interfaces que simulan una tecnología previa. La meta que se ponen es reducir o eliminar el aprendizaje, haciendo la tecnología ampliamente disponible para la gente que está dispuesta a adaptarse al nuevo sistema. Los usuarios algunas veces se encuentran renuentes a invertir en el aprendizaje de un nuevo sistema si ellos ya conocen diferentes formas para realizar una tarea utilizando viejas tecnologías [7] [8].

Uno de los principales ejemplos de una interfaz basada en la tecnología anterior es el tractor "Phelps", introducido en las granjas en 1901 como un reemplazo de los caballos [19]. El tractor "Phelps" podría ser enganchado a un carro o vagón, los granjeros utilizaban un par de riendas para controlar el tractor tal como cuando controlaban a los caballos, como se puede ver en la figura 3.1.

THE PHELPS TRACTOR.

— 1901 —

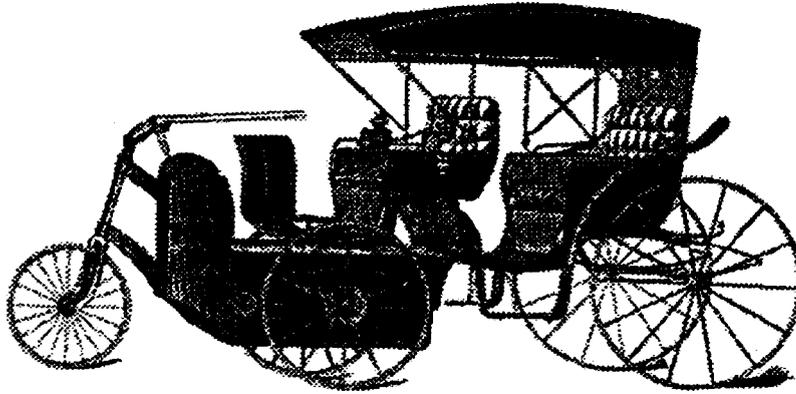


Figura 3.1. Tractor Phelps.

La interfaz de usuario está basada en una tecnología previa – el caballo [19].

El tractor era dirigido jalando la cuerda apropiada. Cuando ambas riendas se soltaban, el tractor iba hacia delante; cuando eran jaladas hacia atrás, disminuía su velocidad y al final se detenía; finalmente, cuando se jalaban hacia atrás de manera fuerte, el tractor retrocedía. Los automóviles controlados por el manejo de ruedas y palancas han estado comercialmente disponibles en los Estados Unidos por casi diez años antes de que el “Phelps” tratara de emular la interfaz de los caballos. Probablemente los diseñadores del tractor “Phelps” creyeron que los granjeros lo encontrarían más fácil de controlar que un vehículo con motor [19].

Los tractores de granja no son los únicos ejemplos que existen. Aunque algunas veces las computadoras se anuncian como que pueden llevar una oficina sin papel, muchas de las interfaces de computadoras están basadas en una metáfora de la oficina tradicional, con documentos, carpetas, archiveros y basurero. En años recientes, intentos para imitar la vieja oficina han aparecido, como en la interfaz “Wang’s Freestyle” [9] y el escritorio “Magic Cap”, como se muestra en la figura 3.2.

Un ejemplo similar en el software de computadora es el “Fractal Design’s Painter”, un programa que intenta simular un amplio rango de brochas, lápices, plumas, pinturas y otras herramientas y materiales del arte de pintar.

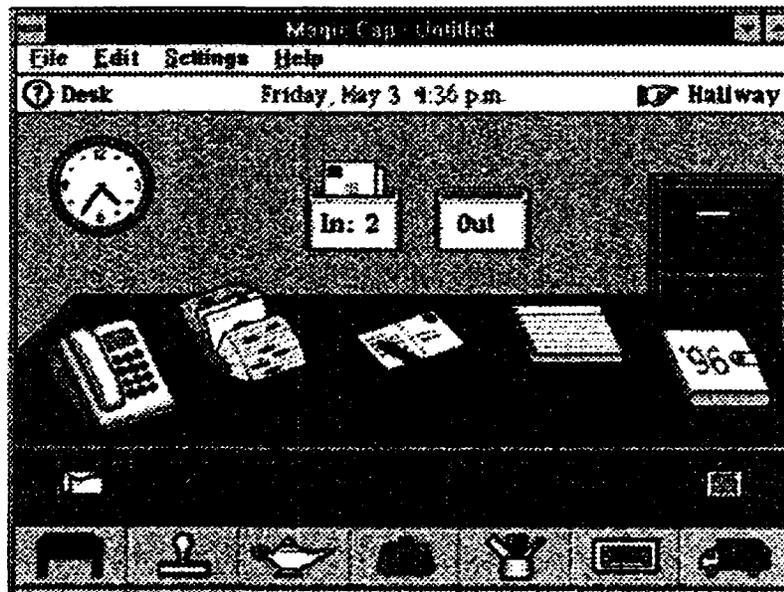


Figura 3.2. Escritorio "Magic Cap" [9].

3.3. Interacción humano-computadora

El objetivo del diseño de las computadoras y los dispositivos relacionados es proporcionar a las personas una manera transparente de las actividades que realizan en su trabajo diario. Para poder lograr lo anterior, es necesario que los diseñadores entiendan la manera de pensar de los usuarios así como las tareas que desempeñan para poder traducir sus conocimientos en un sistema intuitivo. Para poder entender la manera de pensar de los usuarios, muchas veces los diseñadores se convierten en usuarios también [18].

Dentro de la interacción humano-computadora la interfaz no es cualquier cosa que pueda ser conectada en el último momento; su diseño debe ser desarrollado integralmente con el resto del sistema. La interfaz no debe ser solamente presentada como una "cara bonita" sino como un elemento funcional que soporte las tareas que las personas quieren, olvidando los descuidos que puedan ocurrir. Es desde el inicio del proceso del diseño cuando se debe considerar la interacción humano-computadora (HCI, en sus siglas en inglés).

El diseño de sistemas usables y funcionales no es cuestión simplemente de comercialización, sino cada vez más se está convirtiendo en una materia obligatoria. Es por eso que se requieren patrones cuando se está diseñando, seleccionando o inclusive modificando aplicaciones computacionales. Esos patrones deben de ser [18]:

- Convenientes para la tarea.
- Fáciles de utilizar y adaptables para el conocimiento y experiencia de los usuarios.
- Proporcionan la retroalimentación del funcionamiento.
- Despliegan la información en un formato que es personalizable al usuario.
- Son conformados con los “principios de ergonomía del software”.

3.3.1. ¿Qué es la interacción humano-computadora?

Cuando se habla de la interacción humano-computadora, no necesariamente se tiene que contar con un solo usuario enfrente de una computadora. Por *usuario* se entiende que puede llegar a ser desde un usuario hablando individualmente, un grupo de usuarios trabajando juntos o una secuencia de usuarios trabajando en una organización con una tarea específica. Por *computadora* se entiende, dentro del rango de la tecnología, como una computadora de escritorio o un sistema de computación a gran escala, un sistema de procesos o un sistema embebido. Por *interacción* se entiende como cualquier comunicación entre el usuario y la computadora, puede ser directa o indirecta. La interacción directa se da por medio de un diálogo, para obtener retroalimentación acerca del funcionamiento de cada una de las tareas. La interacción indirecta puede implicar algo más, como son los antecedentes. Lo importante es que el usuario interactúe con la computadora para lograr algo [18].

Por tradición, los ergonomistas han hecho énfasis en las características físicas de las máquinas y de los sistemas, además de saber qué tanto afecta el desempeño del usuario. Entonces son incorporados los factores humanos, aunque el término es intercambiable dependiendo de la región, para Reino Unido el término sería ergonomía, pero para Estados Unidos de América y Canadá el término sería factores humanos [18]. Aunque son términos diferentes, los dos tratan con el desempeño del usuario dentro del contexto de un sistema. El uso de las computadoras se ha extendido, incrementando el número de investigaciones especializadas sobre la interacción entre las personas y las computadoras, en relación con los aspectos físicos, psicológicos y teóricos. Cuando se inició con estas investigaciones acerca de la relación que existía entre el humano y la máquina, el nombre con el cual se conocía esa área de investigación era el de *interacción humano-máquina* pero fue cambiando cuando creció el interés por las computadoras.

Con el paso del tiempo la tecnología ha producido un profundo efecto en la manera en que se almacena, accede y se utiliza la información dentro de las empresas [18]. La interacción humano-computadora puede ser utilizada en diferentes disciplinas como una especialización, pero en las que es aceptada como punto central es en la ciencia de las

computadoras y el diseño de sistemas. Es en este último donde la interacción humano-computadora se vuelve una parte esencial dentro del proceso del diseño.

La función que realiza la interacción humano-computadora es envolver el diseño, implementación y evaluación de los sistemas interactivos en un contexto que pueda ser entendido por el usuario en aspectos de su trabajo y tareas principales.

3.3.2. ¿Quién está involucrado en el diseño de la interacción humano-computadora?

La interacción humano-computadora es vista como una área multidisciplinaria. El diseñador ideal puede tener diferentes tipos de experiencia en varios tópicos como son los siguientes [18]:

- Psicología y ciencia cognitiva para darle al diseñador un conocimiento preceptivo del usuario.
- Habilidades para resolver problemas cognitivos.
- Ergonomía para observar las capacidades físicas de los usuarios.
- Sociología para ayudar al diseñador a entender el amplio concepto de la interacción.
- Ciencias de la computación e ingeniería para construir la tecnología necesaria.
- Diseño gráfico para producir una efectiva presentación de la interfaz.
- Técnicas de escritura para producir el manual de usuario.

Para un solo diseñador son demasiados tópicos, inclusive para cuatro, o hasta para un equipo de diseño promedio. Es por eso que los diseñadores tienden a tomar una decisión de irse por un tópico u otro, pero esto es un error, no es posible diseñar un sistema interactivo efectivo de un solo tópico que se encuentra aislado de los demás. Los datos de entrada son requeridos por todos los demás tópicos. Es por eso que se requiere de la interacción de todos los tópicos para poder alcanzar un diseño efectivo.

3.4. Interfaces de usuario

Los modelos del diseño de interfaces utilizados en sistemas mecánicos juegan roles similares en los sistemas de computadoras, se basan en diferentes modelos de interacción. Existen diferentes maneras de diseñar un dispositivo dado. Consciente e inconsciente [7], cada interfaz de uso se basa en un modelo que forma la base para saber cómo es manejado el mecanismo.

Los dos principales modelos para el diseño de dispositivos son [7]:

1. **El modelo orientado al mecanismo:** Se basa en una interfaz, genera un beneficio para los ingenieros y los desarrolladores. Después de todo, si una tarea es ingeniería, el modelo orientado a la actividad es esencialmente el modelo orientado al mecanismo. Las soluciones basadas en el modelo orientado al mecanismo pueden ser simples y elegantes.

Los sistemas con interfaces pueden ser más fáciles de mantener, adaptar y realzar para el ingeniero o para el usuario experto. Estos sistemas se convierten en algo importante si son utilizados en diferentes ambientes para ayudar al usuario en sus tareas. Si la mayoría de los usuarios eventualmente necesitan acceder a la funcionalidad completa del sistema, es mejor introducirlos al modelo orientado al mecanismo desde un principio. Si el sistema falla y las interfaces son funcionales para los usuarios, los problemas serán más fáciles de diagnosticar, reparar, etc.

2. **El modelo orientado a la actividad:** Se basa en una interfaz, en las tareas de los usuarios que proveen un conjunto complementario para los beneficios de los usuarios. Una exitosa interfaz basada en tareas permitiría trabajar con un modelo sencillo.

El principal beneficio del modelo orientado a la actividad es lo rápido y efectivo de aprender. Las operaciones que el usuario debe realizar estarían directamente accesibles en la interfaz, como el apretar un sencillo botón. Una interfaz construida de modo que esté orientada a la actividad debería habilitar el uso eficiente de un sistema por los usuarios expertos, a través de una forma apropiada de navegación, arreglo óptimo de características, ayuda efectiva en características avanzadas, entre otras.

3.5. Diseño educativo

Para entender un poco más sobre lo que se ha mencionado anteriormente, se debe tomar en cuenta el diseño educativo como una abertura para conocer más acerca de los sistemas de entrenamiento.

Uno de los propósitos del diseño educativo es proveer tanto un destino apropiado como el camino correcto para llegar a ese destino, siempre y cuando uno sea responsable de crear un sistema de entrenamiento. El destino es normalmente alguna forma de aprendizaje que los aprendices deben cumplir, mientras que el camino es una de las muchas rutas que el instructor puede seguir para facilitar el aprendizaje.

El diseño educativo exhibe sus bases como un simple proceso para ayudar a crear entrenamiento efectivo de una manera efectiva [10].

Algunas personas se refieren al diseño educativo como la “ciencia” de instrucciones porque es seguida por un conjunto de teorías y métodos y está relacionado con entradas y salidas. El diseño educativo se basa en puras suposiciones del mundo real [15]. Otras personas ven el diseño educativo como un “arte” porque el mejor diseño usualmente tiene una relación directa con la creatividad y el talento del diseñador.

El crear un sistema de entrenamiento sin la utilización del diseño educacional es una principal causa de que el sistema falle [10].

3.5.1. ¿Qué es el diseño educativo?

Anteriormente se discutió que el diseño educativo era una ciencia, un arte o una manera de crear entrenamiento. De alguna manera todos estos conceptos están bien definidos, pero para entender un poco más lo que es el diseño educativo se puede definir como un conjunto de reglas o procedimientos para crear entrenamiento que realiza lo que debe de realizar. Algunos de estos procedimientos se dedican a averiguar qué realiza el entrenamiento, mientras que otros permiten al usuario conocer cuáles son las metas. Otros procedimientos se centran en esas metas, además de que se alcancen.

El diseño educativo es una forma de planear el sistema de entrenamiento desde que se tiene la idea, se plasma en el código, hasta que se realiza la primera revisión y se está listo ejecutarse. *Esto es un modelo trabajado que puede ser utilizado para administrar los conceptos y tareas que son parte de un proceso de entrenamiento exitoso* [10].

Para el diseño educativo todo se basa en la invención, desde el desarrollo de los procedimientos, herramientas y la tecnología del diseño instruccional. Esta tecnología no es algo natural, es hecha por el hombre para satisfacer sus necesidades de aprendizaje [15].

Existen muchas maneras de realizar un diseño educativo, pero eso no evita que cada una siga los principios básicos. De alguna forma u otra utilizando estos principios se va a obtener el éxito en el sistema de entrenamiento.

El diseño educativo es una tecnología que puede ser utilizada para diferentes funciones como son [15]:

- Desarrollo de diferentes experiencias de aprendizaje las cuales promocionan la adquisición de conocimiento o habilidades alcanzadas por el estudiante.
- Incorporación de diferentes estrategias de aprendizaje dentro de las experiencias educacionales para hacer que el conocimiento o las habilidades lleguen a ser más eficientes y efectivas.

La educación envuelve a los estudiantes en actividades de aprendizaje, guía a los usuarios por el conocimiento apropiado, monitorea el desempeño de los estudiantes, así como proporciona retroalimentación de cada una de las actividades de aprendizaje de los estudiantes.

La educación se encuentra fundamentada en los siguientes aspectos importantes: aprender a conocer, aprender a hacer y aprender a ser [14].

- **Aprender a conocer:** Tener una percepción amplia de los conocimientos considerando la posibilidad de profundizar en ellos. Es muy importante aprender a aprender para poder obtener lo mejor que ofrece la educación aprovechándolo al máximo.
- **Aprender a hacer:** Saber hacer las cosas, para obtener un mayor beneficio a la hora que se presenten situaciones en las cuales se requiere trabajar en equipo, aprender a colaborar con los demás. *Aprender a hacer en el marco de las distintas experiencias sociales o de trabajo que se ofrece a los jóvenes y adolescentes.*
- **Aprender a ser:** Tener identidad propia, *actuar con creciente capacidad de autonomía, de juicio, y de responsabilidad personal.* Aprovechar todas las posibilidades dentro de la educación como lo son: memoria, razonamiento, sentido estético, capacidades físicas, aptitud para comunicar.

La educación debe ser vista como un todo en donde los sistemas educativos deben dar prioridad a la adquisición del conocimiento independientemente de la forma de aprendizaje.

3.5.2. Aprendizaje individual

Se debe entender la diferencia entre los estudiantes y los aprendices. Los estudiantes son personas que adquieren sus conocimientos y habilidades por medio de la educación, los aprendices son personas que adquieren sus conocimientos o cambian su comportamiento de acuerdo a sus experiencias. Comparando estos dos significados se puede especificar que todas las personas son aprendices, pero sólo aquellos que se someten a situaciones educativas se convierten en estudiantes. Una vez que una persona comienza a leer un libro o utiliza una herramienta de simulación para poder entender y practicar en ciertos aspectos de su educación se convierte en un estudiante, además es importante mencionar que en la medida que realice cada una de sus prácticas de una manera individual sus conocimientos se van a ir incrementando.

Es muy difícil, pero no imposible, que un aprendiz adquiera sus conocimientos en un grupo, solo se puede lograr si el aprendiz intercambia conocimientos con otro aprendiz, es decir, que exista apoyo entre los miembros de dicho equipo [15].

Un estudiante no aprende sin prácticas individuales lo cual es la demostración de sus conocimientos o habilidades.

3.6. Ventajas e desventajas del software de aprendizaje

3.6.1. Ventajas

Las ventajas del software de aprendizaje son las siguientes [17]:

- La motivación es considerado uno de los más importantes motores del aprendizaje, incita al pensamiento y la actividad. Los alumnos se motivan al trabajar con programas educativos, lo que da como resultado mayor dedicación y por lo tanto, un aprendizaje mejor.
- La interacción de los programas mantiene a los alumnos atentos y les predispone a una actividad y a una constante participación, fomentando el desarrollo de su iniciativa, pues se ven obligados continuamente a tomar decisiones ante la respuesta de la computadora a sus acciones.
- Los programas didácticos contienen ejercicios de refuerzo, liberando al profesor de trabajos repetitivos, monótonos y rutinarios, de manera que puede prestar más atención a cada uno de los estudiantes. Algunos programas educativos realizan informes de seguimiento y control que ayudan al profesor a controlar el trabajo realizado por el alumno.
- En el campo de la educación especial, este tipo de materiales proporciona mayores ventajas. La computadora puede abrir caminos alternativos a muchas de las limitaciones físicas y psíquicas que condicionan la comunicación y el acceso a la información.

3.6.2. Desventajas

Las desventajas del software de aprendizaje son los siguientes [17]:

- Una de sus principales limitaciones de los sistemas educativos es la rigidez de los diálogos dentro del diseño orientado al mecanismo. El diálogo entre profesor y alumno es flexible, se puede ajustar a las necesidades que se vayan presentado. Sin embargo, el diálogo con la computadora no podrá evolucionar en una dirección que no haya sido prevista.
- Los alumnos suelen buscar estrategias para cumplir con el mínimo esfuerzo. Muchas veces consiguen las respuestas correctas mediante fallos, es decir, si saben que la computadora, tarde o temprano les da respuestas correctas, los alumnos fallan hasta obtenerlas.
- El software educativo debe motivar, ejercitar, simular, entrenar, debe ser un tutorial con elementos gráficos y explicaciones de calidad. Sin embargo, hay que tener en cuenta que una motivación excesiva puede llegar a provocar ansiedad en el alumno, por lo que el uso de éste debe controlarse.

3.7. Características de un sistema de entrenamiento de administración de conmutadores CISCO

Un sistema de entrenamiento debe de contar con una serie de características que lo hacen diferente de todos los demás, en el aspecto de mejora y confiabilidad, además de realista. Las características que debe contener un sistema de entrenamiento de administración de conmutadores CISCO son las siguientes [11][12][13]:

- **Usabilidad.** Cuánto puede ser utilizado el sistema por los usuarios, para lograr metas específicas con efectividad y satisfacción en algún contexto en específico. En la medida que sea utilizado el sistema de entrenamiento por los usuarios, mayor será el conocimiento adquirido por ellos.
- **Interactividad.** Debe existir interacción entre el humano y la computadora en todos los aspectos y conocimientos, para poder alcanzar el término de usabilidad.
- **Funcionalidad.** Debe existir una relación entre las tareas y sub-tareas del sistema de entrenamiento para alcanzar la máxima funcionalidad que se puede obtener. Sin funcionalidad el sistema sería inservible.
- **Realista.** Ser realista en todos los aspectos, para cuando se presente un problema saberlo resolver como si se estuviera frente al conmutador físicamente.
- **Originalidad y uso de tecnología avanzada.** La originalidad debe ser considerada en todos los aspectos, tanto en las herramientas utilizadas, como en la visualización de la información, así como la principal tecnología avanzada que se esté utilizando actualmente: "open source".
- **Fomento de la iniciativa y el autoaprendizaje.** El usuario debe contar con una iniciativa para realizar las prácticas que se le presenten a fin de irse adentrando a conocer profundamente el sistema. El autoaprendizaje del alumno lo hará dedicado y constante en cada una de las actividades que realice.

El sistema debe contar con una interfaz gráfica que sea funcional para el usuario, tan real como si fuera la misma Hyper-terminal con la cual estuviera trabajando si se contara con el equipo físicamente. El diseño instruccional en que se basa el sistema de entrenamiento es, como se mencionó anteriormente, el modelo orientado a la actividad. Este modelo tiene como principal ventaja que es rápido y efectivo de aprender. Aplicados al sistema de entrenamiento es útil para que los usuarios puedan contar con un interfaz más funcional y accesible para ellos.

CAPÍTULO 4. TRABAJO PROPUESTO

4.1. Descripción del proyecto

4.1.1. Introducción

La herramienta de simulación es una aplicación en la cual el código fuente se encuentra disponible de manera libre (“open-source”) para el escrutinio, la modificación o la distribución [29].

El software “open-source” es código el cual usualmente puede ser copiado y modificado libremente, existe una diferencia entre el software “open-source” y el software propietario que es controlado generalmente por la compañía que lo fabrica. Existe toda una comunidad de programadores, investigadores y compañías las cuales comparten sus trabajos por medio del Internet. El software “open-source” ha crecido en los últimos años con la realización de muchos programas de este tipo y se ha convertido en un completo rival para las compañías que se dedican a la venta de software. El software “open source” es utilizado por compañías, universidades, entre otras entidades.

El mejor software “open-source” conocido es Linux. Este sistema operativo ha sido cobijado por diferentes compañías de computadoras como lo son IBM, Dell computer Corp. y Hewlett-Packard Co. International Data Corp. una empresa dedicada a la investigación de tecnología.

Por supuesto, las personas que se encuentran trabajando en proyectos de software libre tienen ventajas que el dinero no puede comprar [31]. Los programadores no requieren de abogados para poder crear las licencias del software ni para negociar o discutir los términos de los contratos. Este tipo de software, por el simple hecho de ser libre los abogados pierden el interés en este tipo de software en donde no hay dinero de por medio. Cualquier persona puede bajar el software y probarlo, inclusive los programadores que tienen el código fuente, pueden hacerle modificaciones [32].

La herramienta de simulación es desarrollada en software propietario (ver sección 4.1.3), para poder liberarlo es necesario contar con la licencia “GNU General Public License”, esta licencia es útil para poder compartir y modificar el software “libre”(ver Anexo D), es decir, que el software es libre y puede ser accesible para todos los usuarios que aspiran adquirir determinado software.

Para liberar y hacer público cualquier nuevo programa desarrollado, o en este caso la herramienta de simulación es necesario agregar las siguientes notas al programa. Al inicio de cada archivo de código fuente debe ser agregada la palabra “Copyright” o derechos de autor y una referencia de donde puede ser encontrada la nota completa.

Ejemplo:

*esta línea que contenga el nombre del programa y una idea de lo que hace.
Copyright (C) AÑO nombre del autor*

*Este programa es Software Libre; usted puede redistribuirlo
y/o modificarlo bajo los términos de la "GNU General Public
License" como lo publica la "FSF Free Software Foundation",
o (a su elección) de cualquier versión posterior.*

*Este programa es distribuido con la esperanza de que le será
útil, pero SIN NINGUNA GARANTIA; incluso sin la garantía
implícita por el MERCADERO o EJERCICIO DE ALGUN PROPOSITO en
particular. Vea la "GNU General Public License" para más
detalles.*

*Usted debe haber recibido una copia de la "GNU General Public
License" junto con este programa, si no, escriba a la "FSF
Free Software Foundation, Inc.", 59 Temple Place - Suite 330,
Boston, MA 02111-1307, USA.*

Adicionalmente se debe agrega información de cómo contactarle por correo electrónico y convencional.

4.1.2. Comparación entre conmutador real y la herramienta de simulación

Es importante mencionar que los conmutadores son útiles para una empresa o universidad que dependen fuertemente de las tecnologías de información. Estos equipos son demasiados caros y la mayoría de las empresas o universidades no pueden adquirirlos.

El costo de los conmutadores varía dependiendo del modelo, características y funciones con que cuenta. La lista de precios en los conmutadores de la serie 1900 que son utilizados en los cursos de CCNA de Cisco por la universidades y por algunas empresas del área de redes computacionales y de telecomunicaciones va desde \$895.00 hasta \$2995.00 dólares, sin contar con el equipo adicional (computadoras, enrutadores, cables, etc.) que debe ser adquirido para poder realizar cada una de las prácticas que se llevan a cabo en el tercer semestre del CCNA de Cisco.

La necesidad de contar con una herramienta de simulación gratuita apegada a la realidad capaz de sustituir los conmutadores es eminente. Es por eso que se opta por la utilización del software "open source".

Para entender porque una herramienta de simulación es mejor que los conmutadores reales, es necesario hacer una comparación entre ellos en los siguientes aspectos:

4.1.2.1. Economía

Los conmutadores reales son costosos, es muy difícil que una universidad o empresa pequeña pueda adquirir algún equipo como el utilizado en las grandes empresas dedicadas a las áreas de redes computacionales y telecomunicaciones. Como se mostró anteriormente, el costo de los conmutadores reales utilizados en los cursos del CCNA de Cisco es muy alto, además, se deben adquirir otros equipos como computadoras, cables, etc.

En cambio, la herramienta de simulación desarrollada con software libre puede llegar a ser accesible para las universidades y aquellas empresas que no pueden contar con el equipo físico, porque no tendrían que pagar por obtener la herramienta. Además, podrían contar con el código fuente de la aplicación para poder hacer las modificaciones a la misma de acuerdo a sus necesidades.

4.1.2.2. Facilidad de obtención e instalación

Los conmutadores reales son difíciles de conseguir por su elevado costo, como se mencionó anteriormente, además su instalación es compleja por el hecho que se tienen que conectar cada uno de los puertos que van a ser utilizados, así como la configuración de los mismos.

En cambio, para obtener la herramienta de simulación no existe ningún problema. Como se mencionó anteriormente, la herramienta de simulación es un software libre realizado de manera gratuita, puede ser obtenido directamente a través de un sitio de Internet donde se encuentra disponible el código ejecutable y fuente. La herramienta de simulación puede ser utilizada por aquellas instituciones que no cuentan con el capital suficiente para hacerse de equipo de interconexión de redes como los conmutadores. Además su instalación es mucho más sencilla que los conmutadores reales, porque solo se debe instalar la aplicación en el equipo de los estudiantes o empleados de las empresas.

4.1.2.3. Movilidad

Es complicado mover de un lugar a otro los conmutadores reales porque se requiere mover todo el equipo de interconexión de redes. Mover los equipos de un lugar a otro no es factible por que pueden llegar a dañarse, como se sabe, cada conmutador es muy costoso y sería muy difícil de reemplazar.

En cambio, la herramienta de simulación puede ser movida de un lugar a otro sin que exista algún problema, una vez instalada en el equipo de usuario no es necesario volver a instalar, simplemente se ejecuta la aplicación y se pueden realizar cada una de las prácticas del tercer semestre del CCNA de Cisco. Si el usuario no cuenta con un equipo portátil

(Laptop), puede guardar la aplicación en un disco y transportarla para ser instalada en otro equipo donde pueden realizar dichas prácticas.

4.1.2.4. Portabilidad

Tanto los conmutadores reales como la herramienta de simulación son portables, porque pueden ser utilizados tanto en un sistema operativo como en otro. Los conmutadores reales solo requieren de una aplicación diferente dependiendo del sistema operativo para acceder a ellos. En cambio la herramienta de simulación simplemente se instala y puede funcionar de la misma manera sin importar en el sistema operativo en que se encuentre.

4.1.3. Herramientas utilizadas

La herramienta utilizada para la realización de la herramienta de simulación es versátil y está basada en la programación orientada a objetos, la herramienta seleccionada es la siguiente:

- **Visual Basic .NET**

4.1.3.1. Visual Basic .NET

Para la programación de la interfaz y de cada una de las funciones se utilizó Visual Basic .NET, esta herramienta es desarrollada por Microsoft. El lanzamiento de la plataforma .NET viene acompañada de una nueva versión de Visual Basic. Visual Basic .NET cuenta con diferentes características como son el soporte de herencia, además se ha simplificado y modificado para que funcione con el .NET Framework.

La elección de esta herramienta de desarrollo fue por las diferentes características con las que cuenta. A continuación se muestra una lista de ellas [33]:

- Programación orientada a objetos (POO), con clases, herencia y constructores parametrizados.
- Tratamiento de errores con excepciones.
- Sobrecarga de funciones.
- Espacios de nombres.
- Gestión de hilos (threads) libres para la creación de código 100% reentrante, algo importante en servidores de aplicaciones de alta disponibilidad y facilidad de ampliación.
- Opción de verificación de sintaxis más rigurosa ("option strict")

Esta herramienta de desarrollo, como se mencionó anteriormente es muy versátil, con presionar un solo botón se puede compilar y ejecutar, generando el archivo de la aplicación. Además, cuenta con una serie de opciones con las cuales se pueden monitorear las variables en las cuales los resultados no son los esperados. También cuenta con una ayuda en línea para poder determinar la sintaxis de las instrucciones.

Existen algunas diferencias en Visual Basic .NET en comparación con Visual Basic. Algunas referencias a propiedades de los controles han cambiado y aunque acostumbrarse a ello no representa mayor dificultad, al principio puede resultar algo incomodo y molesto. La realización de los instaladores es quizás un poco más complicado que en Visual Basic 6.0 [34].

4.1.4. Funciones de la herramienta de simulación

Esta herramienta de simulación tiene como principal funcionalidad apoyar a los alumnos y profesionistas en el campo de las redes y en el proceso de aprendizaje, tratando que su desempeño sea similar para ellos cuando se encuentran frente a conmutadores reales utilizados en el laboratorio.

La herramienta de simulación dentro del curso de CCNA realiza algunas prácticas del tercer semestre. La topología del laboratorio se muestra en la figura 4.1.

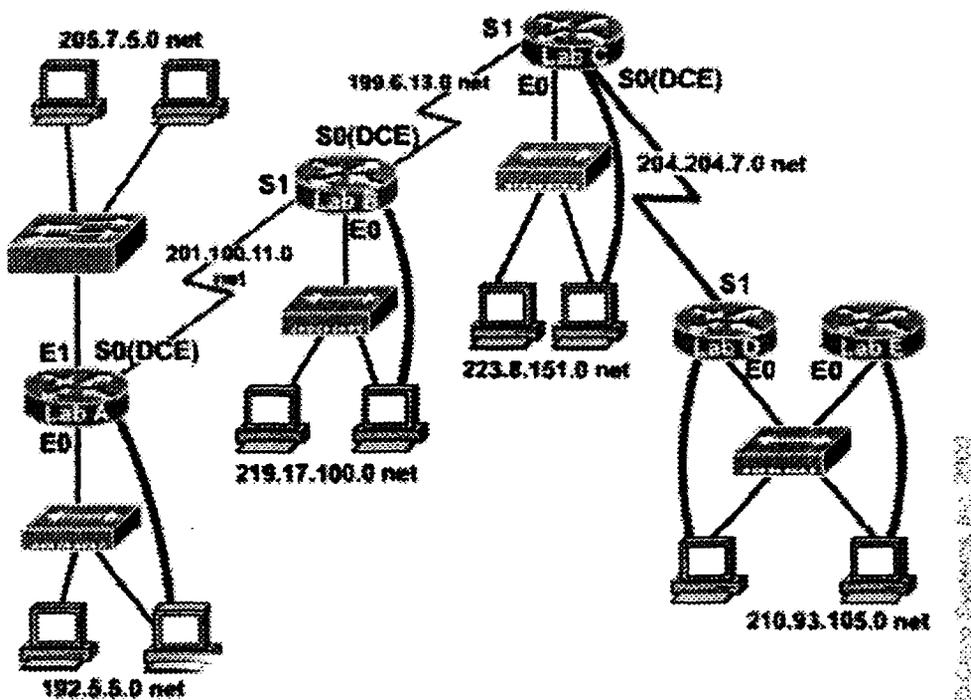


Figura 4.1. Topología del laboratorio del semestre 3 de CCNA.

En este curso se realizan diferentes prácticas que son útiles para aprender a configurar un conmutador en su nivel más bajo.

La herramienta de simulación ayudará a entender de manera básica la administración de los conmutadores Ethernet. La administración del conmutador será por medio de una interfaz a través de un menú como cuando se utiliza la Hyper-terminal y se configura un conmutador real.

En cada una de las prácticas se realizan funciones básicas para la configuración del conmutador, entre esas tareas básicas se pueden observar las siguientes:

1. Configurar el número de intentos para la introducción de la contraseña del conmutador, así como el tiempo en el cual el intruso será detectado una vez alcanzado el número de intentos de la contraseña. También puede ser configurada el tiempo en que “Management Console” permanece inactiva. Cambiar el modo del estado LED a “port status”, “utilization” y “duplex status”.
2. Cambiar la configuración del sistema, en donde pueden ser modificados el nombre del sistema, el modo de conmutación, así como también el uso de multi-difusión para almacenar y enviar los paquetes. Habilitar y deshabilitar el modo “half duplex” y el control de congestión.
3. Configurar la dirección IP del conmutador por medio del menú “Network Management”. En esta opción se encuentra un sub-menú con la opción “IP Configuration”, dentro de este sub-menú puede ser configurada principalmente la dirección IP del conmutador así como su máscara de subred y el “default gateway”. La dirección IP del conmutador puede ser utilizada para realizar un Ping o Telnet al conmutador. La dirección IP también puede ser utilizada para llevar a cabo la administración del conmutador de una manera remota, por medio de una interfaz basada en web. Ahora bien, otra opción dentro del menú “Network Management” se encuentra un sub-menú con la opción “Bridge – Spanning Tree”, dentro de este sub-menú puede ser habilitado o deshabilitado el protocolo de “spanning tree”, así como también pueden ser modificadas las opciones de “bridge priority”, “max age”, hello time” y “forward delay”.
4. Configurar el conmutador para tener acceso por medio de HTTP (Hypertext Transfer Protocol, en sus siglas en inglés.), utilizando Internet Explorer o Netscape para poder tener acceso al conmutador y verificar la configuración del conmutador y sus puertos. Internet Explorer o Netscape proporcionarán una interfaz gráfica mostrando una vista frontal del conmutador pudiendo seleccionar cualquier puerto y verificar sus características. (En esta versión de la herramienta de simulación no podrá ser utilizada esta función a través del navegador de Internet, simplemente se hará la demostración de habilitar y deshabilitar el puerto).
5. Configurar el conmutador para que opere en modo de conmutación “fragment free”. Configurar cualquiera de los puertos en el conmutador de tal manera que opere como “full duplex” y “port fast”. Además de poder agregarle prioridad al puerto.

6. Configurar en el conmutador VLAN's (Virtual Local Area Networks, en sus siglas en inglés), para separar grupos de usuarios basándose en las funciones que realizan sin importar dónde se encuentren físicamente. Normalmente todos los puertos del conmutador se encuentran en la misma VLAN 1, el administrador de red puede crear diferentes números de VLAN's y asignar diferentes números de puertos a esas nuevas VLAN's. Al crear dos VLAN's en un conmutador con 24 puertos, cada VLAN contaría con 12 puertos, los usuarios de una VLAN no tendrán acceso a los recursos de la otra VLAN. Para que una VLAN puedan tener acceso a los recursos de la otra VLAN la comunicación debe ser establecida por medio de un enrutador.

Ahora bien, dentro del menú VLAN's, pueden ser agregados diferentes números de VLAN's con su nombre, said, tamaño del mtu, así como habilitar el estado de la VLAN. Una vez agregadas las VLAN's, pueden ser modificados cada uno de los campos mencionados anteriormente.

7. Verificar la versión actual de "firmware" en la opción "Firmware" del menú principal.

La herramienta de simulación está basada en el funcionamiento de la "Hyper-terminal" de Windows cuando se accede a un conmutador por medio de un cable de consola. La administración de los conmutadores es realizada por medio de menús, como se puede ver en la figura 4.2 y en la figura 4.3, al final de cada pantalla existe un espacio en donde se recibe cada una de las letras, la herramienta de simulación procesa cada una de las letras introducidas, las identifica y ejecuta el menú que será activado, mostrando el resultado en la pantalla principal.

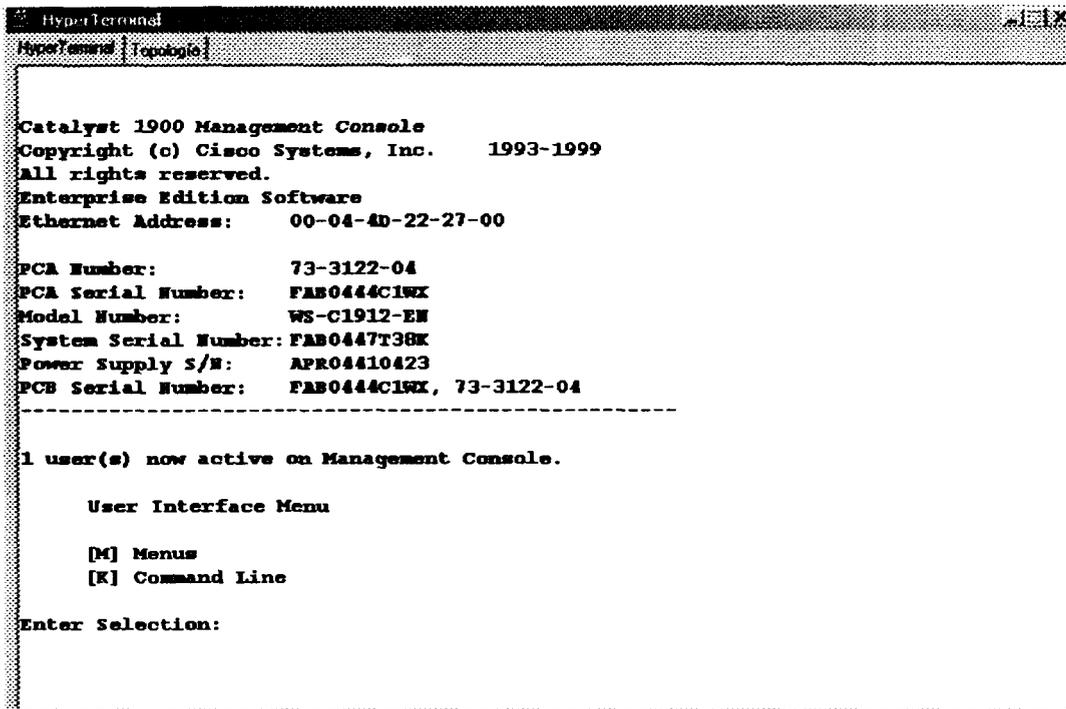


Figura. 4.2. Inicio de la herramienta de simulación basada en menús.

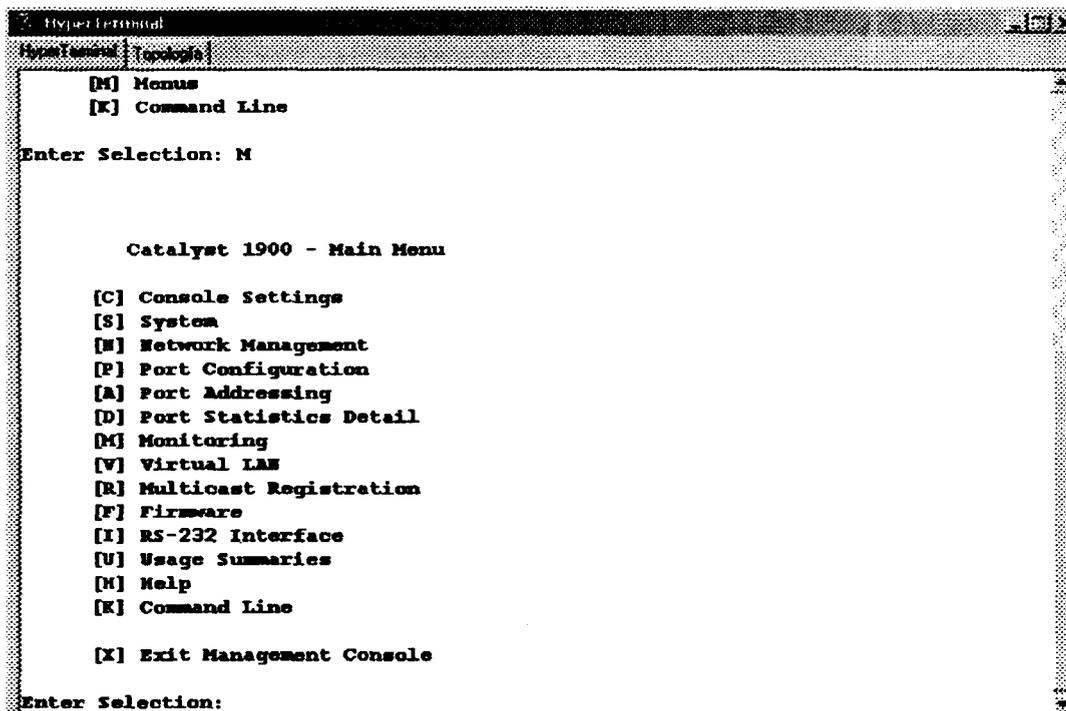


Figura. 4.3. Menú principal de la herramienta de simulación.

4.1.5. Descripción de los comandos de la herramienta de simulación

Un conmutador Cisco de la serie 1900, como se mencionó anteriormente, es administrado por medio de menús. Dentro de cada una de los menús, se puede contar con diferentes características que pueden ser modificadas para que el funcionamiento del conmutador sea el óptimo. La administración del conmutador también puede ser realizada por medio del Internet en donde se puede ver cuáles son los puertos que están activos, cuál es el modo de operación de los puertos, full duplex o half-duplex, entre otras características disponibles.

Para poder mantener la seguridad dentro del conmutador se manejan una serie de contraseñas por medio de las cuales se puede tener acceso a él. Esas contraseñas son las siguientes:

- **Password:** Esta contraseña es utilizada para entrar al conmutador, en donde los comandos son los más básicos y no se pueden realizar muchas operaciones de configuración.
- **Secret Password:** Esta contraseña es utilizada para entrar a un nivel más alto, en donde se tienen otros privilegios y existen más comandos en donde se pueden realizar operaciones sobre la configuración del conmutador.

Dentro de la seguridad del conmutador también existe la opción de establecer el número de intentos que se permiten para poder escribir la contraseña correcta, si ésta no es escrita correctamente después de ese número de intentos, entonces el acceso no será permitido. Estas contraseñas son principalmente utilizadas para cuando el acceso es por medio del Internet y cuando se intenta acceder al conmutador por primera vez en cualquier modo de operación.

El conmutador puede ser accedido por medio de línea de comando, en este aspecto, es parecido a los enrutadores Cisco que son utilizados en el laboratorio de redes, en ese modo de operación pueden existir diferentes comandos como los utilizados en los enrutadores, pero para fines de esta tesis ese modo no será visto.

4.1.5.1. Lectura de caracteres

Cada caracter es leído desde la ventana de la aplicación como se mostró en las figuras 4.2 y 4.3 y es identificado para mostrar la selección que se ha hecho y para poder realizar diferentes operaciones con el conmutador.

El proceso para reconocer cada uno de los caracteres que son escritos en el programa se describe a continuación:

1. **Introducción del caracter.** El alumno o profesionalista, introduce en la ventana de la herramienta de simulación el caracter del menú que esta siendo visualizado para que ejecute la función que el usuario desea.
2. **Captura del caracter.** La herramienta de simulación captura el caracter para poder decidir qué opción del menú será ejecutada. Se utiliza una estructura de control "switch-case", dependiendo del caracter que se captura es la acción que realiza.
3. Si el caracter leído no existe en el menú que está siendo desplegado en la herramienta de simulación, entonces la aplicación no realiza ninguna acción, esperando a que se introduzca un caracter válido.

El recorrido (lectura) de cada uno de los caracteres a través de los menús es hecho utilizando una máquina de estado finitos en donde se estable que acción se realiza si se selecciona algún caracter de los diferentes menús que existen en la herramienta de simulación, esto con el fin de tener un seguimiento de lo realizado en ella.

El funcionamiento de las máquinas de estados finitos consiste en ir de un estado a otro, una vez que los caracteres han sido recibidos. Lo que se tiene que realizar es simplemente ir pasando de estado en estado siguiendo cada una de las flechas de las transiciones que existen en autómeta. Por ejemplo, el autómeta de la figura 4.4 y la palabra de entrada "bb". El autómeta inicia su operación en el estado q_0 -que es el estado inicial-, y al recibir la primera b pasa al estado q_2 , pues en el diagrama hay una flecha de q_0 a q_2 con la letra b. Luego, al recibir la segunda b de la palabra de entrada, pasará del estado q_2 a él mismo, pues en la figura se puede ver una flecha que de q_2 regresa al mismo estado, con la letra b [35].

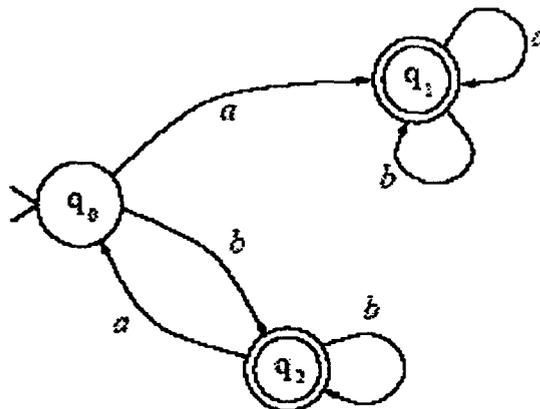


Figura 4.4. Notación gráfica [35].

El camino puede ser visualizado en el diagrama de estados como una *trayectoria* recorrida de estado en estado [35]. Por ejemplo, para el autómata finito de la figura 2.4 la trayectoria seguida para la palabra ab consiste en la secuencia de estados: q_0, q_1, q_1 .

Dentro de los autómatas finitos, donde pertenecen las máquinas de estados finitos, los estados son los únicos medios que pueden recordar los eventos que ocurren (por ejemplo, los caracteres que se han leído hasta ese momento); esto quiere decir que son máquinas de memoria limitada. Por ejemplo, las computadoras digitales son máquinas de memoria limitada, aunque el número de estados dentro de su memoria puede llegar a ser enorme [35].

Las máquinas de estados finitos pueden ser representadas además de una notación gráfica, por un formato matemático para representar una misma información que contiene un diagrama de estados. Al utilizar terminología matemática en vez de dibujos, se refiere a una *notación formal* [35].

Para fines de esta tesis, la notación matemática no será necesaria. Como se mencionó anteriormente, en la herramienta de simulación existen una serie de estados que pueden ir cambiando dependiendo de la selección que se realice en cada uno de los menús que tiene la herramienta.

Pueden existir diferentes errores cuando se está diseñando la máquina de estados finitos, los cuales deben ser controlados para que el funcionamiento sea el óptimo. Dentro de estos se encuentran los siguientes:

1. Que “sobren palabras”, es decir, que el autómata acepte palabras que no debería aceptar. Por lo tanto es una solución incorrecta.
2. Que “faltan palabras”, que existan palabras en el lenguaje considerado que no son aceptadas por el autómata finito, cuando deberían de serlo. Por lo tanto es una solución incompleta.

Volviendo al ejemplo anterior de la figura 2.4, si se propusiera ese autómata para el lenguaje de la palabra en el alfabeto (a, b) que no tiene varias a's seguidas. Esta solución no puede ser, porque [35]:

1. Hay palabras, como “baa”, que tiene secuencias continuas de la letra “a”, sin embargo, son aceptadas por el autómata.
2. Hay palabras, como “ba”, que no tienen secuencias seguidas de “a”, sin embargo, no son aceptadas por el autómata.

Dentro de la herramienta de simulación no se está buscando formar palabras, simplemente se busca una secuencia de ellas para poder llegar a un menú determinado.

En la pantalla de la herramienta de simulación se despliega el menú que da la opción de acceder, ya sea al modo menús o al modo línea de comando. Para fines de esta tesis sólo estará activo el modo en forma de menús. Por lo tanto, para poder alcanzar el menú principal es necesario introducir la letra “M”, una vez dentro de ese menú los estudiantes y profesionistas pueden acceder a cualquier sub-menú del menú principal. Esto puede ser observado en las figuras 4.2 y 4.3.

Ahora bien, si el siguiente menú que es seleccionado es “Network Management”, será necesario presionar la letra “N” y así sucesivamente en cada uno de los menús a los cuales se quiere tener acceso. Como se puede ver se ha formado una secuencia de letras para determinar cual es el menú que se está accediendo. Esta secuencia es la siguiente:

La secuencia para poder acceder al menú “Network Management” sería “MN” y así para cada uno de los menús que se necesiten ser accedidos. La máquina de estados finitos puede contener diferentes estados para acceder a cada uno de los menús que se encuentran en la herramienta de simulación. Esos estados se muestran en la figura 4.5. Se puede observar que una vez alcanzado el menú principal, la secuencia para los menús es la siguiente:

1. Console Settings
2. System
3. Network Management
4. Port Configuration
5. VLAN
6. Firmware

La máquina de estados finitos de la herramienta de simulación se muestra a continuación:

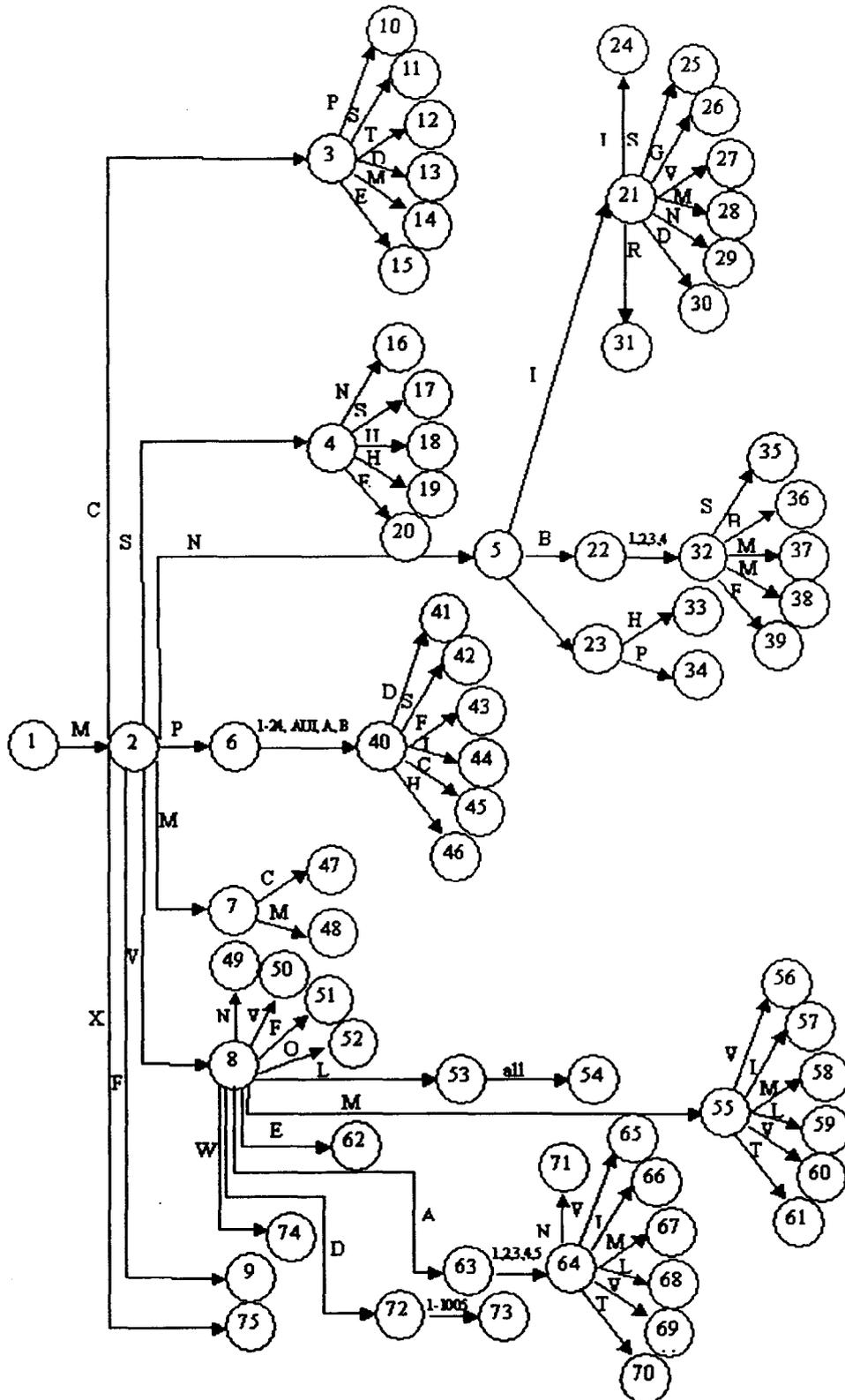


Figura 4.5. Máquina de estados finitos de la herramienta de simulación.

4.2. Estructura del diseño

4.2.1. Análisis del diseño

Para poder demostrar el funcionamiento de la herramienta de simulación es necesario mostrar el análisis de su diseño.

Dentro del análisis para llevar a cabo la herramienta de simulación, se realizó la obtención de la información asistiendo a los cursos del CCNA de Cisco. Tomar los cursos antes mencionados fue importante porque ayudó a ver el funcionamiento de los conmutadores de una manera real, como se desenvuelven los alumnos ante una situación presentada en cada práctica o en cada caso de estudio y como la resuelven cada vez que se les presentan. Además es importante el estudio teórico de cada uno de los temas para poder comprender profundamente cada una de las prácticas que se llevan a cabo.

Existen cuatro cursos o semestres dentro del CCNA de Cisco, el más importante y donde son vistos los conmutadores es el tercer curso o semestre, en este semestre se estudian cada una de las partes que componen a un conmutador, las funciones que realizan y las características generales que lo componen.

4.2.2. Descripción del diseño

En la herramienta de simulación se busca que el comportamiento sea el más cercano a los conmutadores reales que se encuentran en los laboratorios de redes, ya sea dentro de una empresa en donde laboran los profesionistas o en una universidad para los estudiantes.

Es por eso que en esta tesis se toma en cuenta las metodologías, métodos, herramientas y procedimientos de la ingeniería de software, que deben ser utilizados para lograr un producto óptimo. Aplicando los conocimientos conjuntamente de las teorías educativas y de comunicación, permitirán que el producto sea el más adecuado desde el punto de vista educativo.

Desde la perspectiva de la herramienta de simulación, se parte de una necesidad, se especifican los requisitos, se obtiene el diseño mismo, el código respectivo y por último el sistema de software.

Dentro del diseño de la herramienta de simulación existen diferentes elementos que son importantes para la realización de dicho diseño. La importancia de esos elementos es amplia, porque son útiles durante la implementación de la misma.

Los elementos que se mencionan son los siguientes:

- Ciclo de vida de la herramienta de simulación.
- Estructura de la herramienta de simulación.

La herramienta de simulación, durante su desarrollo pasa por una serie de etapas que juntas se denominan ciclo de vida, en cada una de las etapas se definen los procesos, las actividades y las tareas a desarrollar. Además, se llevó a cabo un control de fechas de entregas de cada una de las fases para tener un completo seguimiento y conocer el progreso del proyecto.

La metodología de la herramienta de simulación persigue dos necesidades importantes:

- Realización de una mejor aplicación, conduciéndolo a una mejor calidad
- Un proceso de desarrollo controlado, por medio de seguimiento de cada una de las fases.

Para la construcción de la herramienta de simulación, se siguió el proceso descrito a continuación:

- La obtención de los requisitos del software,
- El diseño del sistema de software (diseño preliminar y diseño detallado),
- La implementación,
- Las pruebas,
- La instalación,
- El mantenimiento, y
- La ampliación y actualización del sistema.

4.2.3. Estructura de la aplicación

La herramienta de simulación está completamente implementada por las herramientas mencionadas al inicio de este capítulo, DevC++ y WinGlade. Su implementación está basada en la arquitectura de programación *Model-View-Control* (MCV, en sus siglas en inglés, ver figura 4.7). La arquitectura MVC se refiere a tres capas para el desarrollo de una aplicación. Cada capa es distinta de las otras dos y realiza una tarea específica.

La capa Modelo (Model en inglés), contiene todos los accesos de la base de datos y la lógica de negocios de la aplicación. Esta capa algunas veces es diseñada para ser una aplicación independiente, es por eso que esta capa de la arquitectura MCV puede ser utilizada en otras aplicaciones si es necesario.

La capa Vista (View, en inglés), es la interfaz del usuario de la aplicación. Esta capa incluirá todas las entradas y salidas hacia y desde el usuario, así como también el diseño

gráfico de la aplicación. El usuario verá esta parte la mayoría de las veces, idealmente, los detalles de las dos capas restantes permanecerán ocultas para el usuario.

La capa Control, mantiene unidas cada unas de las demás capas de la arquitectura MVC. Maneja el intercambio de datos que existe entre la capa Modelo y la capa Vista, así como también controla el flujo de la aplicación.

La separación del código dentro de la arquitectura MVC es ideal para el desarrollo de aplicaciones [43]. Desde que las tres capas se encuentran separadas, la posibilidad del reuso del código ha crecido en gran cantidad.

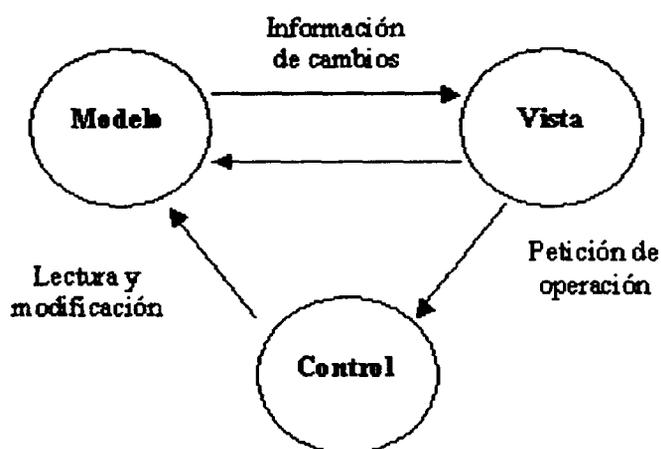


Figura 4.7. La arquitectura MVC utilizada en la herramienta de simulación.

4.2.3.1. Capa Modelo

La capa Modelo contiene la estructura de datos en donde serán almacenados cada uno de los valores que requieren ser cambiados en la configuración del simulador de conmutadores cuando se esté practicando con la herramienta de simulación.

Con todos estos conceptos mencionados anteriormente podremos definir un tipo de estructura para la herramienta de simulación, con el fin de poder almacenar elementos importantes para el funcionamiento del mismo.

La estructura de la aplicación se encuentra conformada de la siguiente manera:

Structure Console

```
Dim pass_intrusion As String
Dim silent_time As String
Dim inactive_timeout As String
Dim default_status_led As String
End Structure
```

La estructura Console es utilizada para almacenar los datos referentes al menú Console. Esos datos son los siguientes:

- El número de intentos para introducir el password (pass_intrusion),
- El tiempo en segundos de la detección de algún intruso (silent time),
- El tiempo en segundo para poner de manera inactiva la consola (inactive_timeout),
- El modo de estado de LED (default_status_led).

Structure Sistem

```
Dim name_system As String
Dim switching_mode As String
Dim store_and_forward As String
Dim half_duplex As String
```

End Structure

La estructura System es utilizada para almacenar los datos referentes al menú System. Esos datos son los siguientes:

- El nombre del sistema o conmutador (name_system),
- El modo de conmutación (switching_mode),
- La utilización del modo “store and forward” para multicast (store_and_forward),
- Habilitar o deshabilitar el modo “half duplex” (half_duplex).

Structure Network_ip

```
Dim ip_address As String
Dim subnet_mask As String
Dim default_gateway As String
Dim vlan As String
Dim dns_1 As String
Dim dns_2 As String
Dim domain_name As String
Dim rip As String
```

End Structure

La estructura Network_ip es utilizada para almacenar los datos referentes al menú Network_ip. Esos datos son los siguientes:

- La dirección IP del conmutador (ip_address),
- El número de mascara de subred (subnet_mask),
- La dirección IP del “default gateway”(default_gateway),
- El número de VLAN a la que pertenece el conmutador (vlan),
- La dirección IP del DNS 1 (dns_1),
- La dirección IP del DNS 2 (dns_2),
- El nombre del dominio (domain_name),
- Habilitar o deshabilitar la utilización de RIP (rip).

```
Structure Network_bridge
  Dim spanning_tree As String
  Dim bridge_priority As String
  Dim max_age As String
  Dim hello_time As String
  Dim forward_delay As String
End Structure
```

La estructura Network_bridge es utilizada para almacenar los datos referentes al menú Network_bridge. Esos datos son los siguientes:

- Habilitar o deshabilitar el algoritmo y protocolo de “spanning tree” (spanning_tree),
- Introducir el valor de la prioridad del puente (bridge_priority),
- El tiempo en segundos de la edad máxima (max_age),
- El tiempo en segundo del saludo al prender el conmutador (hello_time),
- El tiempo en segundo del retraso hacia delante (forward_delay).

```
Structure Network_http
  Dim http As String
  Dim http_port As String
End Structure
```

La estructura Network_http es utilizada para almacenar los datos referentes al menú Network_http. Esos datos son los siguientes:

- Habilitar o deshabilitar el protocolo HTTP (http),
- Introducir el número de puerto a utilizar (http_port).

```
Structure Port
  Dim name_port As String
  Dim status_port As String
  Dim full_duplex As String
  Dim port_priority As String
  Dim path_cost As String
  Dim port_fast_mode As String
End Structure
```

La estructura Port es utilizada para almacenar los datos referentes al menú Port. Esos datos son los siguientes:

- Nombre o descripción del puerto (name_port),
- Habilitar o deshabilitar el estado del puerto (status_port),
- Habilitar o deshabilitar el modo “full duplex” (full_duplex),
- Introducir el valor de la prioridad del puerto (port_priority),
- Introducir el costo del camino (path_cost),
- Habilitar o deshabilitar el modo “port fast” (port_fast_mode).

Structure Monitoring

```
Dim capturing_frame As String
End Structure
```

La estructura Monitoring es utilizada para almacenar los datos referentes al menú Monitoring. Esos datos son los siguientes:

- Habilitar o deshabilitar el modo de captura de paquetes (capturing_frames),

Structure Vlan

```
Dim domain_name As String
Dim vtp_mode As String
Dim vtp_pruning_mode As String
Dim vtp_traps As String
End Structure
```

La estructura Vlan es utilizada para almacenar los datos referentes al menú Vlan. Esos datos son los siguientes:

- El nombre del dominio (domain_name),
- Habilitar o deshabilitar el modo “VTP” (vtp_mode),
- Habilitar o deshabilitar el modo “VTP pruning” (vtp_pruning_mode),
- Habilitar o deshabilitar el modo “VTP traps” (vtp_traps).

Structure Vlan_modify

```
Dim name_vlan As String
Dim said As String
Dim mtu As String
Dim bridge_1 As String
Dim bridge_2 As String
Dim vlan_state As String
End Structure
```

La estructura vlan_modify es utilizada para almacenar los datos referentes al menú vlan_modify. Esos datos son los siguientes:

- El número de la VLAN a insertar o modificar (una sola VLAN) (numero_vlan),
- Nombre o descripción de la VLAN (name_vlan),
- El número de SAID (said),
- El número de MTU (mtu),
- El número de Bridge 1 (bride_1),
- El número de Bridge 2 (bride_2),
- Habilitar o deshabilitar el estado de la VLAN (vlan_state).

4.2.3.2. Capa Vista

La capa Vista despliega los datos desde la capa Modelo, donde se encuentran almacenados en la interfaz que es mostrada al usuario, como se muestra en la figura 4.8. Cada vez que alguna letra es introducida en la herramienta de simulación se despliega cierta información. Esa información puede llegar a ser información que el usuario requiere para almacenarla dentro de la estructura de datos, o bien, información necesaria para seguir observando la configuración en la cual se encuentra el simulador de conmutadores.

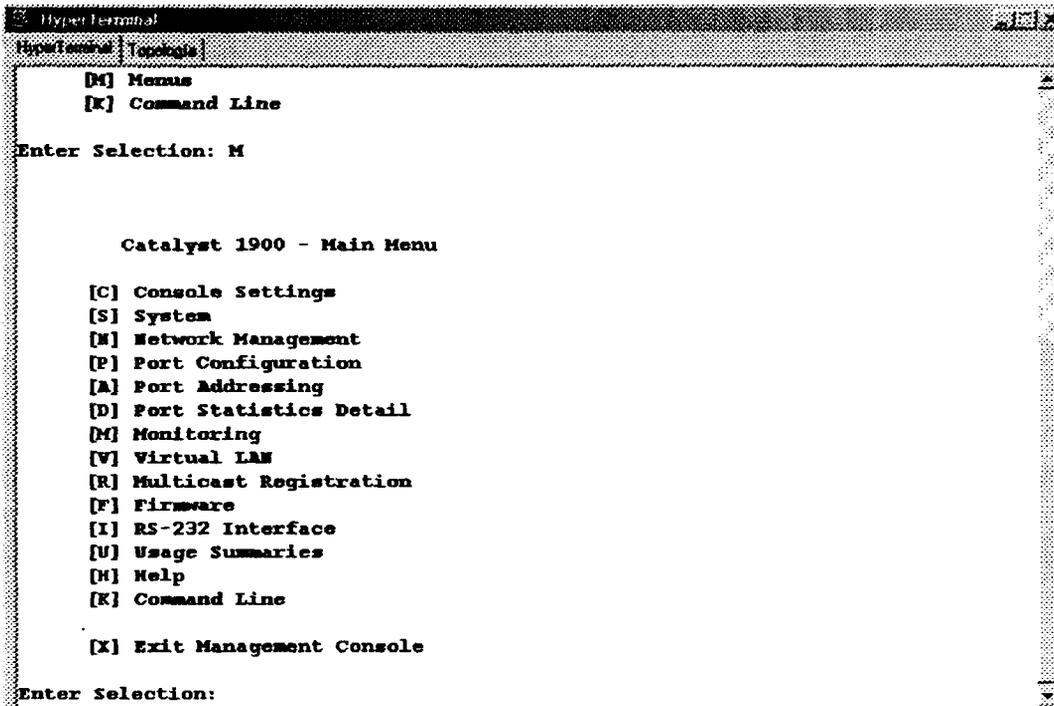


Figura 4.8. Espacio de trabajo de la interfaz gráfica de usuario.

4.2.3.3. Capa Control

La capa Control, contiene una serie de funciones que mantiene el control principal de cada una de los comandos que son manejados dentro de la herramienta de simulación. Cuando se ejecuta un comando en la interfaz gráfica del usuario, la capa Control verifica en la capa Modelo que los elementos que se están buscando existan dentro de la estructura de datos y entonces, sean mostrados al usuario.

En la capa Control, se puede verificar que se lleve a cabo un “ping” para mostrar que las direcciones IP dentro de la configuración del conmutador pertenecen a la misma subred, utilizando la estructura para obtener los datos necesarios para realizar el “ping”, así como también, utilizando la interfaz gráfica para mostrar los resultados de la operación. Si el “ping” no es alcanzado, las direcciones IP no pertenecen a la misma subred.

En la figura 4.9 se muestra cómo es utilizada la capa Control entre las capas Vista y Modelo.

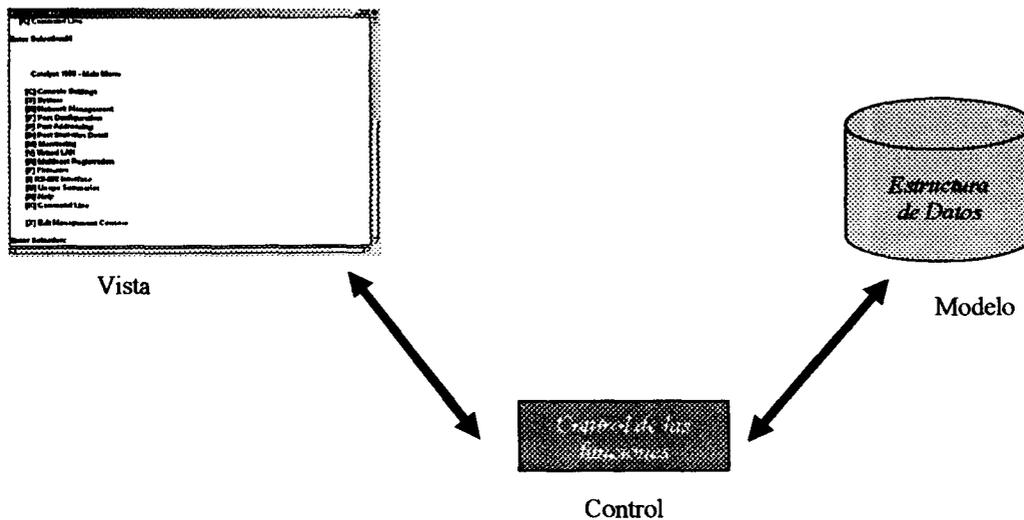


Figura 4.9. Función de la capa Control.

4.3. Conclusiones

En este capítulo es importante mencionar que el código fuente de la herramienta de simulación está totalmente disponible de manera libre para poder hacer todo tipo de modificaciones y además es completamente gratuita para aquellas empresas chicas y medianas, así como universidades que no pueden adquirir el equipo de interconexión de redes para la realización de las prácticas del tercer semestre del CCNA de Cisco. Cabe mencionar que la herramienta de simulación cuenta con características importantes en comparación con los conmutadores reales. Esas características son sobre la economía, facilidad de obtención e instalación, movilidad y portabilidad de la herramienta de simulación.

Además, es importante señalar la utilización de la máquina de estados finitos para determinar el recorrido (lectura) de cada uno de los caracteres que son utilizados en los menús. El uso de esta metodología es importante porque existe un seguimiento de cada uno de los estados con son alcanzados para llegar a cierto menú dentro de la herramienta de simulación.

Es importante mencionar que la obtención de la información es primordial en cada proceso de diseño y desarrollo de una aplicación. La obtención de la información fue importante para la realización de la herramienta de simulación porque ayuda a conocer más de cerca el funcionamiento, como son capturados los datos para realizar modificaciones en la configuración de los conmutadores y como son presentados en la pantalla. Además, la arquitectura MVC utilizada en la herramienta de simulación es importante. Con la utilización de tres capas totalmente independientes (*Model-View-Control*) para el desarrollo de la herramienta de simulación, se amplía la capacidad de reuso del código en gran medida.

CAPÍTULO 5. PRUEBAS REALIZADAS

5.1. Introducción

Para corroborar que la herramienta de simulación es capaz de ayudar a los estudiantes y profesionistas a practicar en sus cursos de la academia de Cisco, fue necesario realizar diferentes encuestas a los alumnos que cursan o han llevado alguna materia relacionada con los cursos del CCNA de Cisco.

5.1.1. Resultados de la encuesta del Anexo A

En el anexo A se muestra una encuesta en donde los resultados exponen la carencia de un simulador de configuración de conmutadores para apoyar tanto a los estudiantes como a los profesionistas en la práctica.

Los resultados de la **pregunta 1** en donde se le cuestiona al estudiante si conoce cual es la función de un simulador son los siguientes.

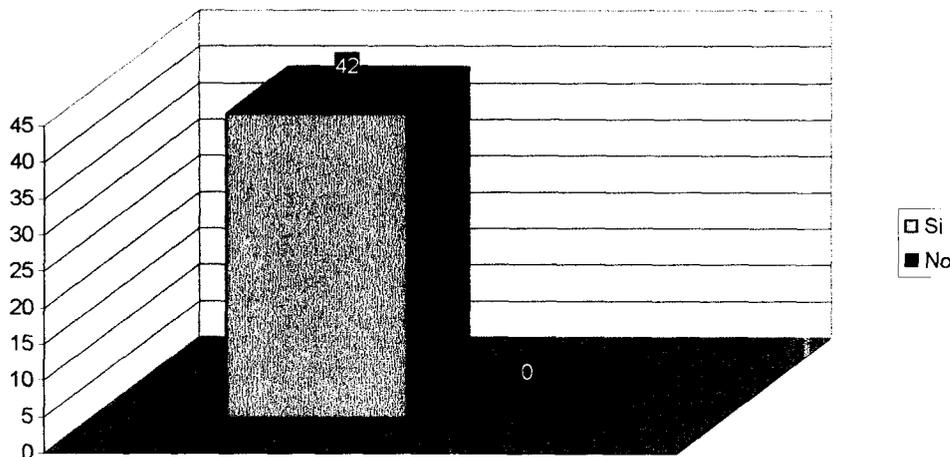


Figura 5.1. Gráfica de la pregunta. ¿Sabes la función que tiene un simulador?

Una vez vista la gráfica de los resultados de la **pregunta 1** se puede determinar que todos los participantes en la encuesta tenían idea de la función que tiene un simulador.

Los resultados de la **pregunta 2** en donde se le cuestiona al estudiante si ha utilizado alguna vez en el laboratorio los conmutadores son los siguientes.

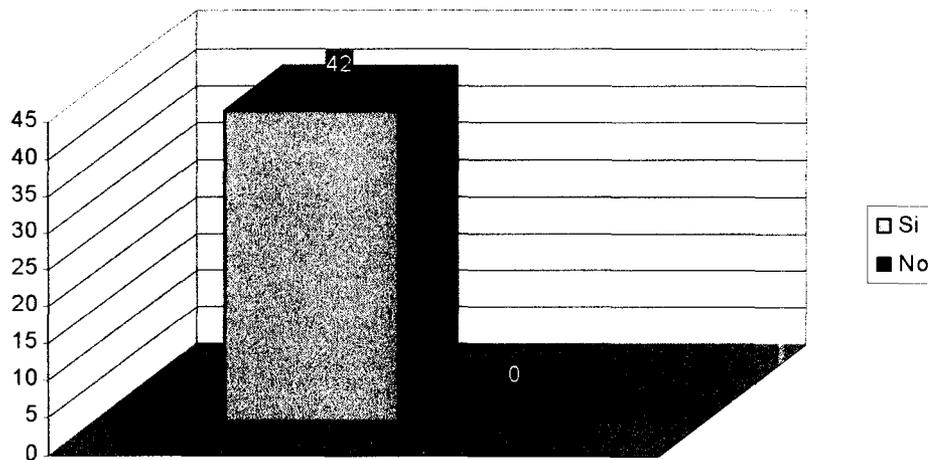


Figura 5.2. Gráfica de la pregunta. ¿Alguna vez has utilizado en algún laboratorio los conmutadores Cisco?

Una vez vista la gráfica de los resultados de la **pregunta 2** se puede determinar que todos los participantes en la encuesta han utilizado alguna vez los conmutadores en los laboratorios de Cisco.

En la **pregunta 3** se le cuestionó al estudiante si conoce algún simulador que le permita ejecutar comandos de los conmutadores, el 31% (13 participantes) contestó que si y además específico que ese simulador es ConfigMaker. Los resultados son los siguientes.

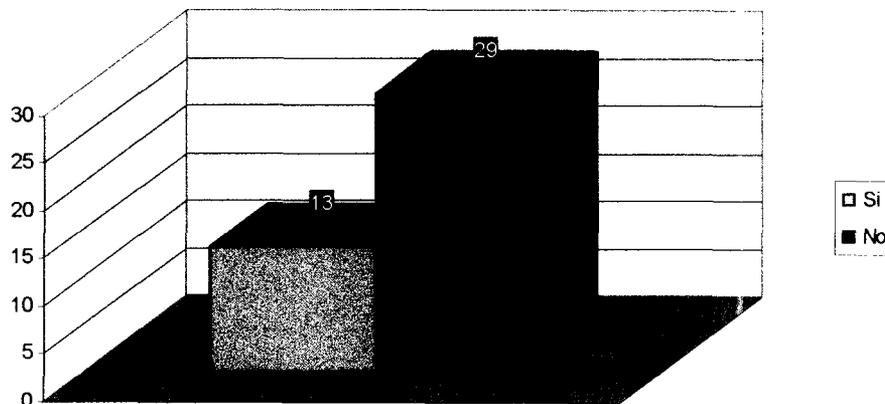


Figura 5.3. Gráfica de la pregunta. ¿Conoces algún simulador de conmutadores que te permita ejecutar diferentes comandos similares a los reales?

Una vez vista la gráfica de los resultados de la **pregunta 3** se puede determinar que la mayoría de los participantes en la encuesta no conocen un simulador de conmutadores que le permitan ejecutar diferentes comandos.

En la **pregunta 4** se le cuestionó al estudiante que característica es importante para considerar en un simulador de conmutadores. Los resultados son los siguientes.

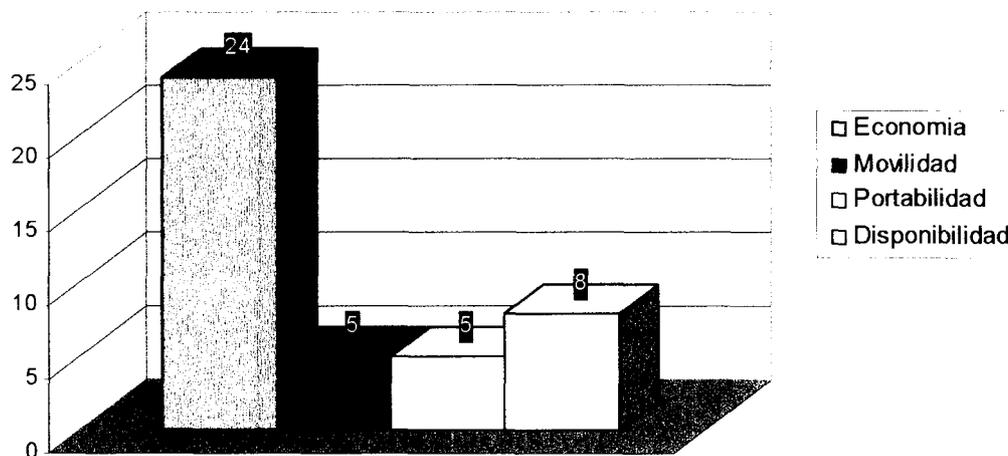


Figura 5.4. Gráfica de la pregunta. ¿Cuáles de las siguientes características crees que sean importantes en un simulador de conmutadores?

Una vez vista la gráfica de los resultados de la **pregunta 4** se puede determinar que la mayoría de los participantes en la encuesta seleccionaron que la característica de economía es la más importante en un simulador de conmutadores, quizás por el fácil acceso que tendrían ellos para el uso del mismo.

En la **pregunta 5** se le cuestionó al estudiante cuales elementos cree que son importantes para que sean agregados a un simulador de conmutadores. Los resultados son los siguientes.

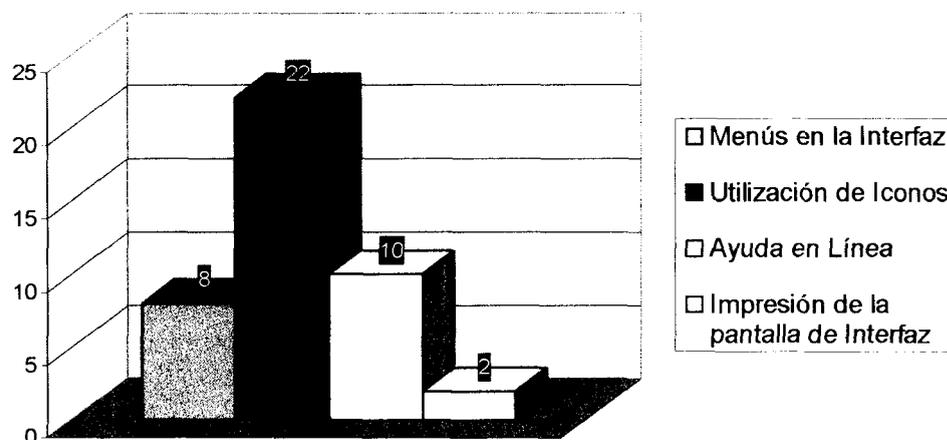


Figura 5.5. Gráfica de la pregunta. De los siguientes elementos, ¿Cuál(es) crees que son importantes para que se agreguen a un simulador de conmutadores?

Una vez vista la gráfica de los resultados de la **pregunta 5** se puede determinar que la mayoría de los participantes en la encuesta seleccionaron que el elemento de utilización de iconos es el más importante dentro de un simulador de conmutadores, quizás porque para ellos es necesario el uso de iconos para tener fácil acceso a cada una de las funciones que presente el simulador.

En la **pregunta 6** se le cuestionó al estudiante si el simulador de conmutadores puede ayudar a realizar las prácticas del laboratorio de Cisco. Los resultados son los siguientes.

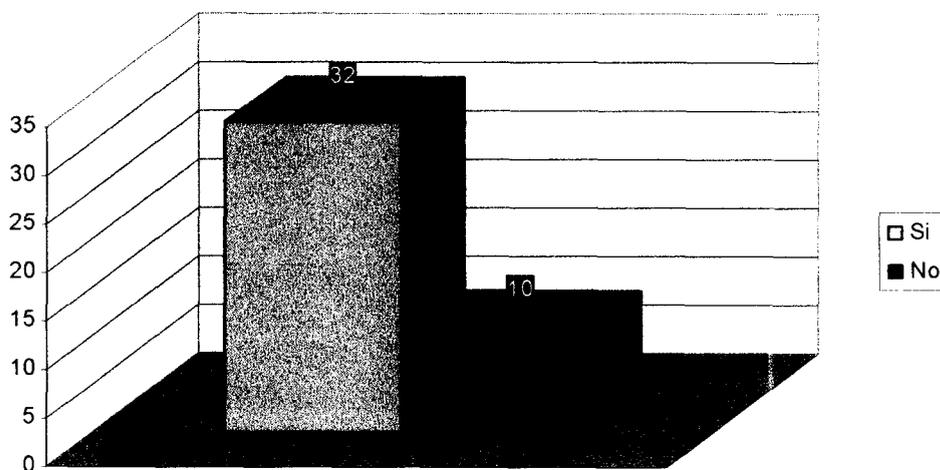


Figura 5.6. Gráfica de la pregunta. ¿Crees que pueda ser de ayuda para realizar las prácticas del laboratorio de Cisco el simulador de conmutadores?

Una vez vista la gráfica de los resultados de la **pregunta 6** la mayoría de los participantes en la encuesta determinaron que el simulador de conmutadores si puede ser de ayuda dentro de los laboratorios para realizar las prácticas antes de llegar al laboratorio. Los participantes que contestaron mencionaron que sería innecesario el realizar la práctica antes de llegar al laboratorio, para ellos de cualquier manera se tenía que volver a realizar la práctica.

En la **pregunta 7** se le cuestionó al estudiante si existiera un simulador de conmutadores lo utilizaría para realizar cada una de las prácticas que se llevan a cabo en el laboratorio de Cisco. Los resultados son los siguientes.

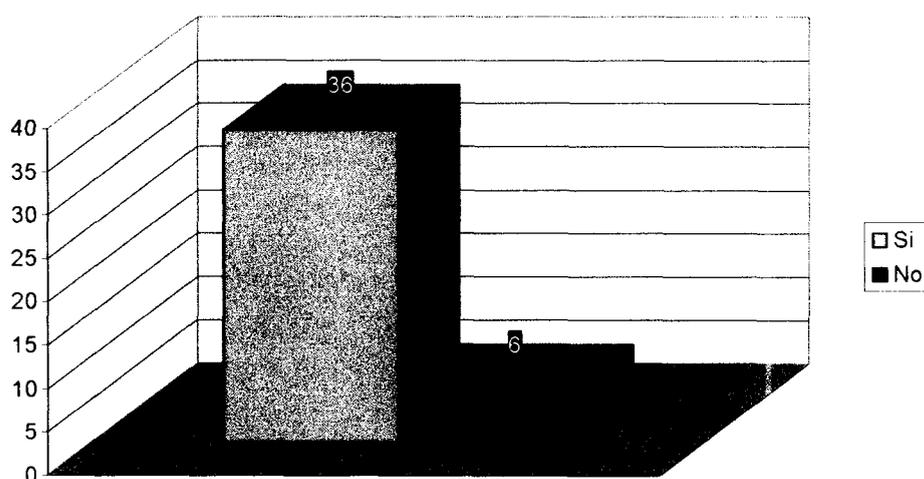


Figura 5.7. Gráfica de la pregunta. ¿Si existiera una herramienta de simulación de conmutadores la utilizarías para practicar los laboratorios de Cisco relacionados a los conmutadores?

Una vez vista la gráfica de los resultados de la **pregunta 7** la mayoría de los participantes utilizarían una herramienta de simulación de conmutadores si existiera, para realizar cada una de las prácticas de los laboratorios de Cisco relacionados con el CCNA del mismo.

En la **pregunta 8** se le cuestionó al estudiante si cree que la utilización de una herramienta de simulación de conmutadores puede mejorar su desempeño al realizar cada una de las prácticas del laboratorio de Cisco. Los resultados son los siguientes.

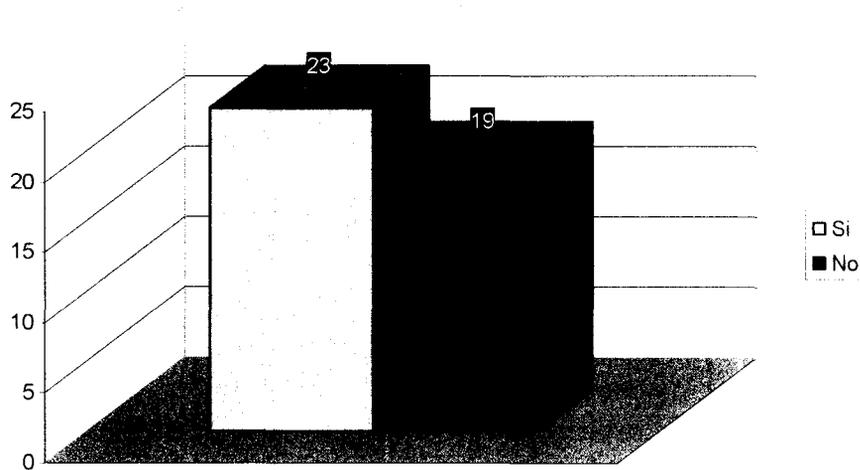


Figura 5.8. Gráfica de la pregunta. ¿Crees que el utilizar la herramienta de simulación de conmutadores pueda mejorar tu desempeño al realizar las prácticas en el laboratorio?

Una vez vista la gráfica de la **pregunta 8** se puede notar que los resultados obtenidos fueron demasiado similares, el 55% para el sí y el 45% para el no. Los participantes que contestaron que si, agregaron que era demasiado útil porque así llegaban preparados para realizar cada una de las prácticas llevadas en el laboratorio. Por el contrario, los participantes que contestaron que no, creen que no mejora su desempeño al realizar cada una de las prácticas del laboratorio.

5.1.2. Resultados de la encuesta del Anexo B

En el anexo B se muestra una encuesta realizada para evaluar el aspecto de usabilidad y el aspecto de funcionalidad, además fue necesario utilizar una práctica del laboratorio (ver Anexo C). Con el anexo C, los participantes en la encuesta, **16 en este caso**, analizarían y probarían el simulador de conmutadores completando los espacios mostrados en el anexo B.

La práctica de prueba mostrada en el anexo C cuenta con suficientes comandos para poder evaluar el simulador de conmutadores. Para realizar dicha prueba, se le daba al participante una hoja con las instrucciones necesarias para realizar cada uno de los pasos, indicándole los comandos a ejecutar en la herramienta de simulación.

5.1.2.1. Funcionalidad

El aspecto de funcionalidad es importante para verificar que el funcionamiento de la herramienta de simulación sea el correcto y que se asemeje a la realidad, para confirmar que así sea, es necesario cumplir con ciertas características. Para evaluar el aspecto de

funcionalidad fue necesario realizar una encuesta (ver Anexo B), en la primera parte de este documento se especifican diferentes características relacionadas con la funcionalidad de la herramienta de simulación. Estas características son las siguientes:

- Funcionaron bien todos los comandos utilizados.
- Actualiza los cambios realizados.
- Valida las entradas a la línea de comandos.
- Grado de satisfacción en cuanto a la funcionalidad.

Las características mencionadas anteriormente cuentan con una ponderación, esta ponderación es la siguiente: 5 – Muy Malo, 6 – Malo, 7 – Regular, 8 – Bueno, 9 – Muy Bueno y 10 – Excelente.

Además de las características mencionadas se incluyó una pregunta abierta en donde se cuestiona a los participantes acerca de que cambios realizaría en el aspecto funcional. Los resultados sobre el aspecto de funcionalidad se detallan en la sección 5.1.2.3.

5.1.2.2. Usabilidad

Así como el aspecto de funcionalidad es importante, el aspecto de usabilidad también lo es. Para poder evaluar el aspecto de usabilidad es necesario poder definirlo. Conforme a ISO 9241, en su parte 11 se refiere a, “enfatisa que la usabilidad de la terminal visual es dependiente en el contexto de uso y que el nivel de usabilidad alcanzado dependerá de las circunstancias específicas en las cuales un producto es utilizado. El contexto de uso consiste de los usuarios, tareas, equipo y el ambiente físico y social que puedan influir en la usabilidad de un producto en un sistema de trabajo”[50].

Para evaluar el aspecto de usabilidad fue necesario realizar, al igual que el aspecto de funcionalidad, una encuesta (ver Anexo B), en la segunda parte de este documento se especifican diferentes características relacionadas con la usabilidad de la herramienta de simulación. Estas características son las siguientes:

- Amigable.
- Robusto.
- Estable.
- Intuitivo.
- Similaridad con un conmutador real.
- Grado de satisfacción en cuanto a la facilidad de uso.

Las características mencionadas anteriormente cuentan con una ponderación, esta ponderación es la siguiente: 5 – Muy Malo, 6 – Malo, 7 – Regular, 8 – Bueno, 9 – Muy Bueno y 10 – Excelente.

Además de las características mencionadas se incluyó una pregunta abierta en donde se cuestiona a los participantes acerca de que cambios realizaría en el aspecto de usabilidad. Los resultados sobre el aspecto de usabilidad se detallan en la sección 5.1.2.4.

5.1.2.3. Evaluación del aspecto de funcionalidad

En esta sección es necesario resaltar que las calificaciones a la herramienta de simulación no fueron menores a 7, por lo que las calificaciones menores a esa calificación no aparecen en cada una de las gráficas.

La **primera pregunta** que se les hizo a los participantes es: **Valida correctamente las entradas de los usuarios.** Esta pregunta se realizó con el fin de verificar que cada uno de los comandos fuera válido para la herramienta de simulación, desde la validación de cada una de las letras utilizadas en cada uno de los menús hasta palabras completas o direcciones IP. En este aspecto de funcionalidad se les pidió a los participantes que evaluaran cada una de los pasos de la práctica para conocer la funcionalidad que presenta la herramienta de simulación en este punto. Los resultados de la encuesta son los siguientes:

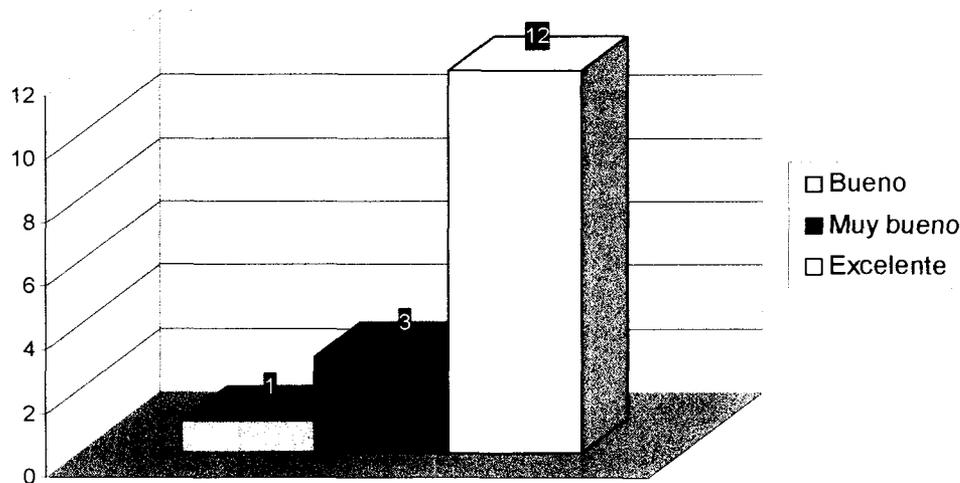


Figura 5.9. Gráfica de la pregunta. “Valida correctamente las entradas de los usuarios.”

Como se puede ver en la gráfica anterior, la mayoría de los participantes estuvieron de acuerdo que la herramienta de simulación valida correctamente cada una de las entradas que el usuario ingresa en la aplicación.

La **segunda pregunta** que se les hizo a los participantes es: **Actualiza los cambios realizados en cada una de las opciones.** Esta pregunta se realizó con el fin de conocer si cualquier cambio efectuado por el usuario en la herramienta de simulación es mostrado al regresar a cada uno de los menús. Los resultados de la encuesta son los siguientes:

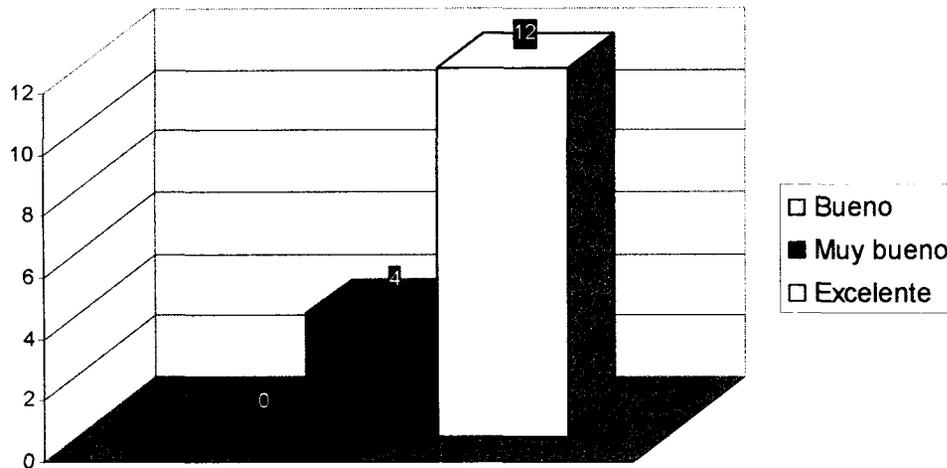


Figura 5.10. Gráfica de la pregunta. “Actualiza los cambios realizados en cada una de las opciones.”

Como se puede ver en la gráfica anterior, la mayoría de los participantes estuvieron de acuerdo que la herramienta de simulación actualiza correctamente cada uno de los cambios realizados en cada una de las opciones de la aplicación.

La **tercera pregunta** que se les hizo a los participantes es: **Funcionaron cada uno de los comandos utilizados.** Esta pregunta se realizó con el fin de conocer si cada uno de los comandos que fueron introducidos funciona correctamente y realizan cada una de las actividades que deben realizar dentro de la herramienta de simulación. Esta pregunta está muy relacionada con la pregunta anterior, si los cambios que se realizan son actualizados, quiere decir que la herramienta de simulación funciona correctamente. Los resultados de la encuesta son los siguientes:

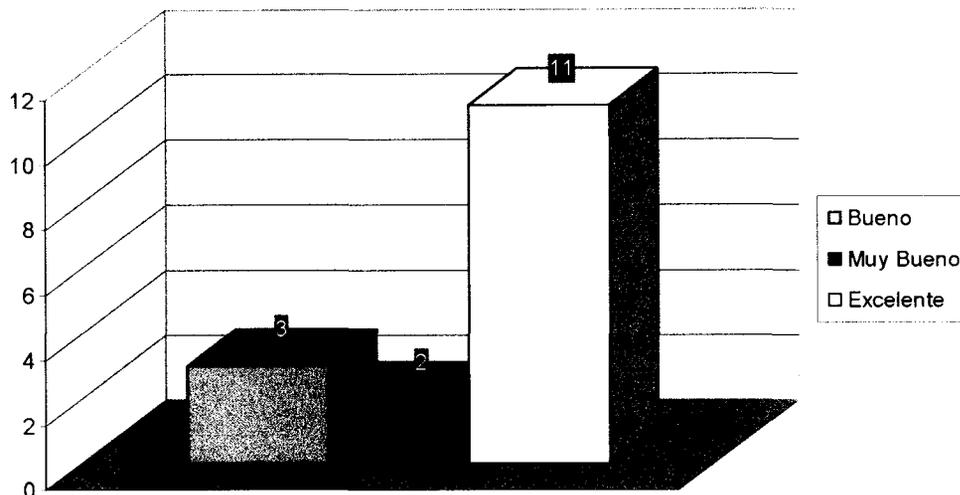


Figura 5.10. Gráfica de la pregunta. “Funcionaron cada uno de los comandos utilizados.”

Como se puede ver en la gráfica anterior, la mayoría de los participantes estuvieron de acuerdo que la herramienta de simulación funciona correctamente en cada uno de los comandos que fueron utilizados, solo pocos participantes mencionaron que existe un error al introducir el número de puertos que son asignados a la VLAN 2, cabe señalar que no es un error de funcionamiento sino de visualización del cursor.

La **cuarta pregunta** que se les hizo a los participantes es: **Calificación final en el aspecto funcional.** Esta pregunta se realizó con el fin de conocer el veredicto final de cada uno de los participantes después de haber analizado las respuestas anteriores

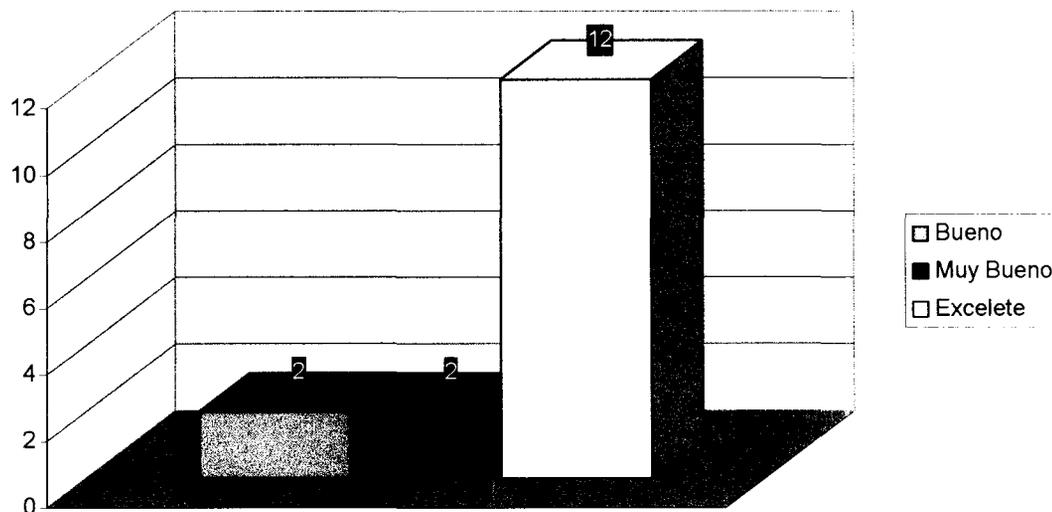


Figura 5.10. Gráfica de la pregunta. “Calificación final en el aspecto funcional.”

Como se puede ver en la gráfica anterior, la mayoría de los participantes estuvieron de acuerdo que la herramienta de simulación funciona correctamente al colocarle en su mayoría una calificación de 10 por lo que la herramienta de simulación es óptima y puede ser funcional para ayudar tanto a los alumnos como a los profesionistas para practicar sobre la configuración de los conmutadores.

Preguntas abiertas

En la evaluación del aspecto de funcionalidad se incluyó una pregunta abierta para que cada uno de los participantes opinara sobre temas no examinados en la encuesta (ver Anexo B). La pregunta realizada es la siguiente:

➤ ¿Qué cambios le harías en el aspecto funcional?

Para esta pregunta las respuestas fueron sobre algunos detalles con que cuenta la herramienta de simulación, el primer detalle se refiere a que es necesario hacer un click en la interfaz de la herramienta de simulación para que pueda funcionar de manera correcta y el segundo detalle se refiere al paso 11 de la segunda práctica de prueba (ver Anexo C) en donde es necesario mover el “scrollbar” de la interfaz hacia abajo para visualizar los datos tal y como lo haría el conmutador real.

Es importante mencionar que los participantes tuvieron razón en los comentarios realizados porque así ayudan a mejorar la herramienta de simulación, que su funcionamiento sea el más óptimo.

En la figura 5.11 se muestra la cantidad de participantes que opinaron acerca del primer detalle (hacer click al iniciar la aplicación) y la cantidad de participantes que opinaron acerca del segundo detalle (el paso 11 de la segunda práctica de prueba). Los resultados son los siguientes.

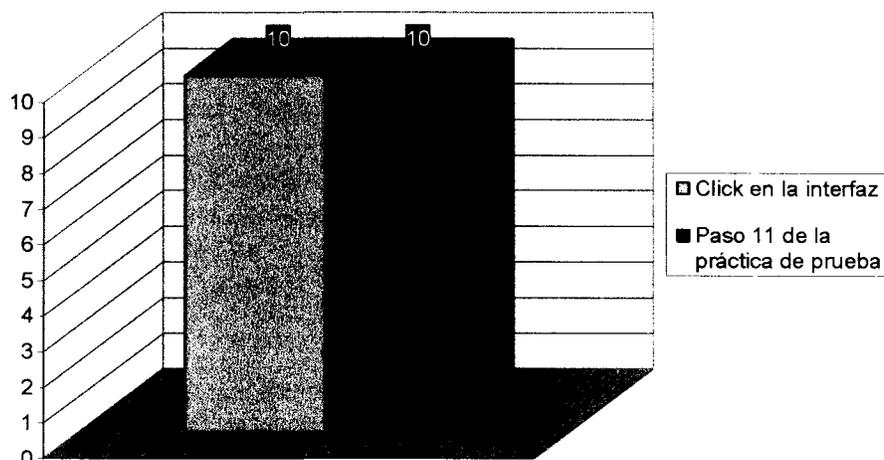


Figura 5.11. Gráfica de la pregunta. “Pregunta abierta.”

Cabe señalar que el número de participantes fue el mismo (16 participantes), en esta pregunta hubo algunos participantes que coincidieron en que existen los dos detalles mencionados anteriormente, pero hubo algunos que solo se percataron del primer detalle o del segundo detalle.

5.1.2.4. Evaluación del aspecto de usabilidad

En esta sección es necesario resaltar que las calificaciones a la herramienta de simulación no fueron menores a 7, por lo que las calificaciones menores a esa calificación no aparecen en cada una de las gráficas.

La **primera pregunta** que se les hizo a los participantes es sobre si la herramienta de simulación es: **Amigable**. Esta pregunta se realizó con el fin de corroborar que tan amigable es le herramienta de simulación. Los resultados de la encuesta son los siguientes:

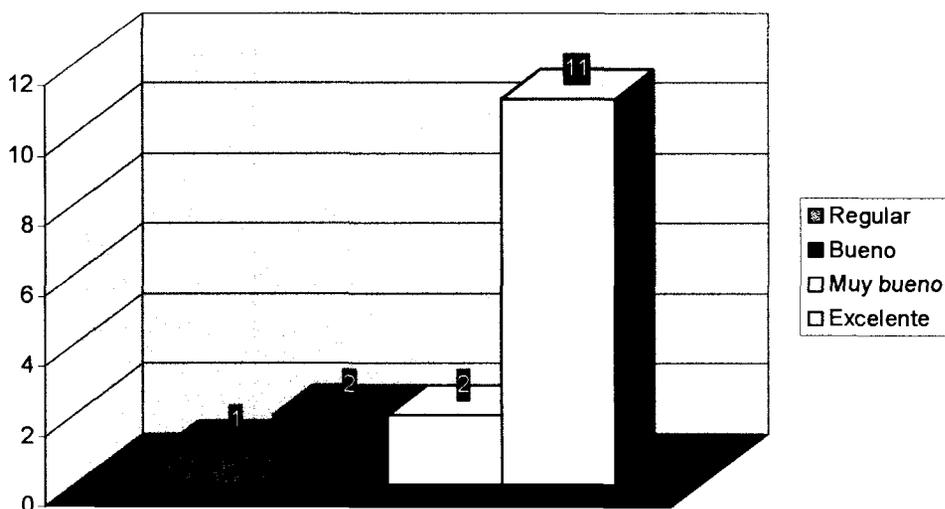


Figura 5.12. Gráfica de la pregunta. “Amigable.”

Como se puede ver en la gráfica anterior, la mayoría de los participantes estuvieron de acuerdo que la herramienta de simulación es amigable y el participante no tuvo dificultad para utilizar la aplicación. La herramienta de simulación puede ser definida como una aplicación muy sencilla para ser utilizada por los resultados mostrados en la gráfica anterior.

La **segunda pregunta** que se les hizo a los participantes es sobre si la herramienta de simulación es: **Intuitiva**. Esta pregunta se realizó con el fin de corroborar si la herramienta de simulación puede ser utilizada por gente que no tiene experiencia en el manejo de la aplicación, si la persona puede intuir que es lo que se puede realizar con la herramienta de simulación. Los resultados de la encuesta son los siguientes:

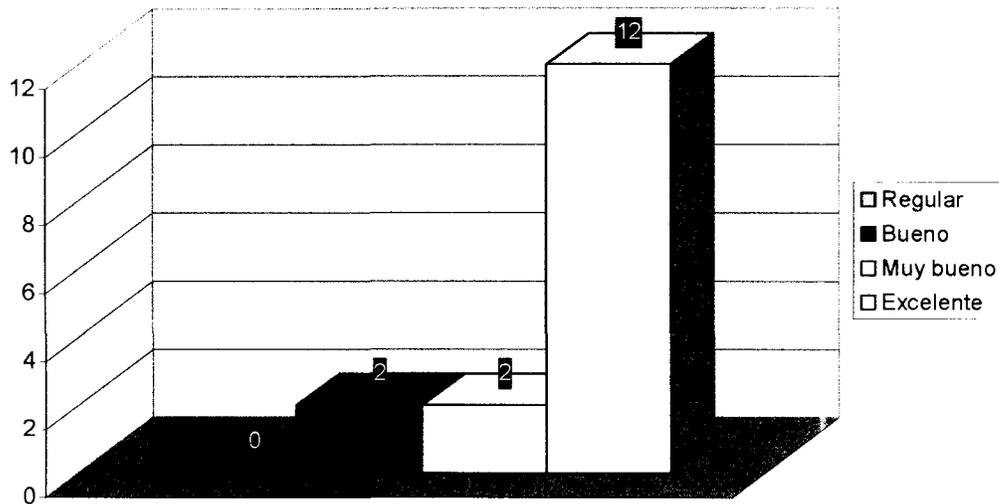


Figura 5.13. Gráfica de la pregunta. “Intuitiva.”

Como se puede ver en la gráfica anterior, la mayoría de los participantes estuvieron de acuerdo que la herramienta de simulación sea intuitiva y el participante no tuvo dificultad para utilizarla. De acuerdo a la pregunta anterior los resultados son similares porque la herramienta de simulación es muy sencilla de manejar y es muy fácil descifrar cuales son los pasos que se tienen que seguir para llegar a un comando en específico.

La **tercera pregunta** que se les hizo a los participantes es sobre si la herramienta de simulación es: **Estable**. Esta pregunta se realizó con el fin de corroborar si la herramienta de simulación es estable, si existe alguna inconsistencia en cada uno de los cambios que se realizan. Los resultados de la encuesta son los siguientes:

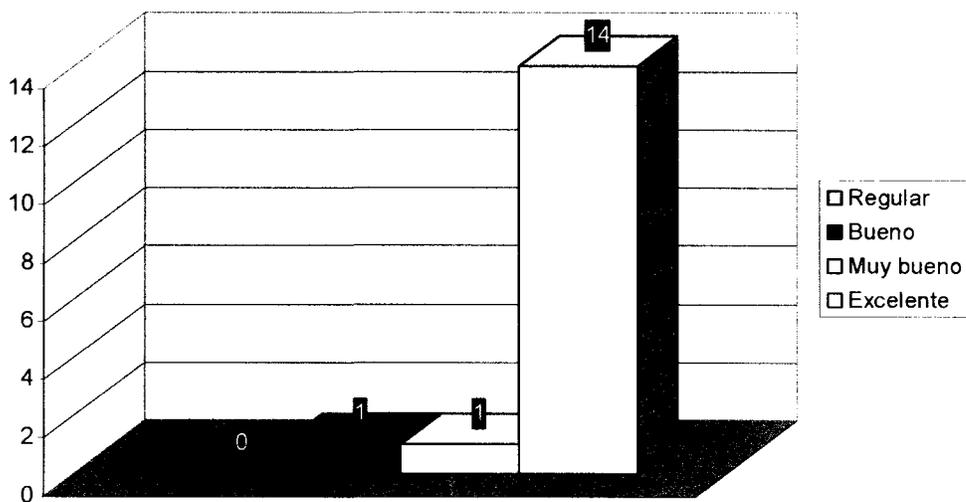


Figura 5.14. Gráfica de la pregunta. “Estable.”

Como se puede ver en la gráfica anterior, la mayoría de los participantes estuvieron de acuerdo que la herramienta de simulación es estable, presenta algunas inconsistencias que los participantes señalaron en la pregunta realizada en la segunda parte de la encuesta (ver Anexo B). En general los resultados fueron favorables.

La **cuarta pregunta** que se les hizo a los participantes es sobre si la herramienta de simulación tiene: **Similitud con un conmutador real**. Esta pregunta se realizó con el fin de corroborar si la herramienta de simulación cuenta con cierta similitud con un conmutador real, si la interfaz y la manera de realizar cada uno de los comandos se asemeja a los conmutadores que se encuentran en el laboratorio.

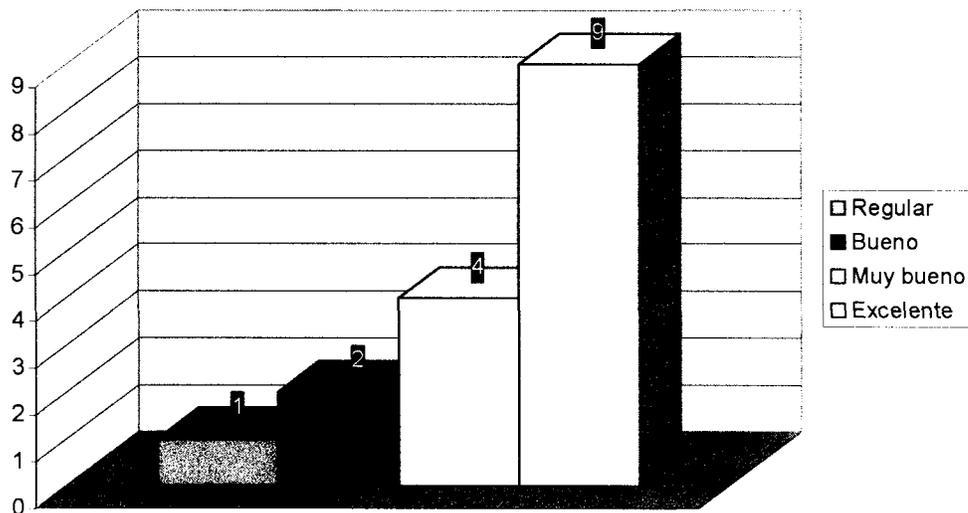


Figura 5.15. Gráfica de la pregunta. “Similitud con un conmutador real.”

Como se puede ver en la gráfica anterior, la mayoría de los participantes estuvieron de acuerdo que la herramienta de simulación tiene una similitud con los conmutadores reales, pero algunos participantes no estuvieron de acuerdo con esta pregunta, quizás porque la herramienta de simulación cuenta con una pestaña en donde puede ser vista la topología sobre la cual se está trabajando.

Preguntas abiertas

En la evaluación del aspecto de usabilidad se incluyó una pregunta abierta para que cada uno de los participantes opinara sobre temas no examinados en la encuesta (ver Anexo B). La pregunta realizada es la siguiente:

- ¿Qué cambios le harías en el aspecto de usabilidad?

Para esta pregunta las respuesta de los participantes fueron muy claras sobre los cambios que podrían hacerse a la herramienta de simulación en el aspecto de usabilidad, estos

cambios son, el poder contar con más topologías para poder practicar en diferentes características de la red, así como también fueron muy pocos los participantes que dijeron que era necesario incrementar el tamaño de la letra utilizada en la interfaz. Estos comentarios hechos por los participantes son útiles, porque en versiones futuras pueden ser resueltas cada una de sus inconformidades.

En la figura 5.16 se muestra la cantidad de participantes que opinaron acerca de los diferentes cambios que pueden ser realizados en la herramienta de simulación. Los resultados son los siguientes.

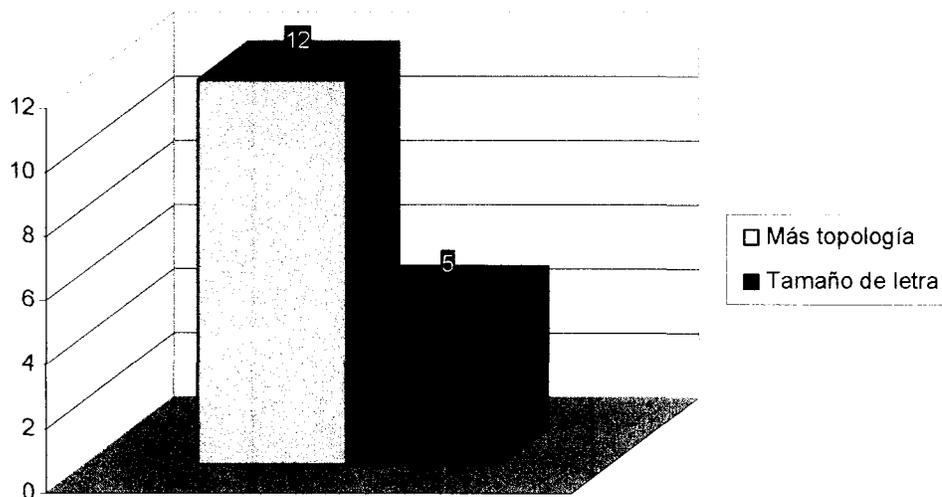


Figura 5.16. Gráfica de la pregunta. “Pregunta abierta.”

Cabe señalar que el número de participantes fue el mismo (16 participantes), en esta pregunta hubo algunos participantes que coincidieron en que existen los dos detalles mencionados anteriormente, pero hubo algunos que solo se percataron del primer detalle o del segundo detalle.

CAPÍTULO 6. CONCLUSIONES

Con el crecimiento del desarrollo tecnológico, las empresas intentan salir adelante frente a los retos que se les puedan presentar utilizando las mejores tecnologías de información relacionadas con las telecomunicaciones y redes computacionales.

Por otro lado, las universidades modernas requieren graduar jóvenes con habilidades técnicas para poder enfrentarse al entorno laboral. Las instituciones educativas deben contar con grandes instalaciones de equipo de interconexión de redes similar al utilizado en las empresas.

Para algunas universidades estos equipos son caros, por ejemplo, los enrutadores y conmutadores, entre otros. Es por eso que, debido a la demanda de los estudiantes y los profesionistas, se observa la necesidad de realizar una herramienta de simulación capaz de poder apoyar a éstos para practicar con los conmutadores y obtener habilidades y conocimientos necesarios para enfrentarse al entorno laboral. La herramienta de simulación fue diseñada para enfocarse en el tercer semestre del CCNA de Cisco.

Es muy importante mencionar, para la realización de la herramienta de simulación fue necesario asistir a los cursos del CCNA de Cisco y así obtener toda la información necesaria sobre el funcionamiento y las características de los conmutadores. Toda la información recabada fue útil para poder realizar una herramienta de simulación apegada a la realidad, que los alumnos sientan que están trabajando como si fuera la consola de la Hyper-terminal.

Además, la herramienta de simulación fue realizada de tal manera que el código fuente este disponible de manera libre (“open-source”).

Es importante mencionar que la herramienta de simulación podrá ser aplicada de igual manera tanto en la industria como en la educación, sin necesidad de una técnica didáctica diferente porque se apega a la realidad simulando que el usuario se encuentra frente a un equipo de interconexión de redes.

La herramienta de simulación funciona como un sistema de entrenamiento, basándose en el diseño educativo, útil como una forma de aprendizaje para los estudiantes y profesionistas. La herramienta de simulación al igual que los sistemas de entrenamiento, se basan en situaciones del mundo real.

En esta herramienta de simulación fue necesario seguir una serie de metodologías o elementos importantes para su realización, como el modelo MVC. El modelo MVC es importante porque se establecen tres capas que son desarrolladas por separado para que cada una de ellas pueda ser modificada por separado sin necesidad de modificar todo el código de la aplicación.

Después de la realización de las encuestas para probar los aspectos de usabilidad y funcionalidad, así como también una sesión de pruebas a la herramienta de simulación, los

participantes pudieron expresar su opinión sobre los aspectos mencionados anteriormente, dando como resultado una herramienta de simulación útil tanto para los estudiantes como para los profesionistas para mejorar su aprendizaje.

6.1 Aportaciones

En la presente tesis se cuenta con diferentes aportaciones importantes. En primer lugar, la realización de la herramienta de simulación de configuración de conmutadores utilizando software “open-source”. Esta herramienta de simulación es implementada para ayudar a los estudiantes y profesionistas a poder prepararse en el campo laboral de una manera sencilla sin la necesidad de tener que adquirir el equipo de interconexión de redes. Además cuenta con una ventaja, con el software “open-source” la herramienta de simulación puede ser modificada o aumentada por los mismos usuarios con algunos conocimientos de programación para poder modificar la topología y realizar prácticas diferentes a las establecidas en los cursos del CCNA de Cisco.

Otra aportación importante es que la interfaz de usuario de la herramienta de simulación es similar a la utilizada por los alumnos en las prácticas, se encuentra apegada a la realidad. Dentro de los modelos del diseño de interfaces existe el modelo orientado a la actividad, utilizado para la realización de la herramienta de simulación. El principal beneficio de utilizar este modelo es lo rápido y efectivo de aprender porque las operaciones que realiza el usuario se encuentran accesibles dentro de la interfaz, como simplemente presionar una tecla.

El costo de los conmutadores reales es muy elevado para universidades y empresas pequeñas. Otra aportación importante es el hecho de que la herramienta de simulación es completamente gratuita. Además es muy fácil poder mover la aplicación de un lugar a otro, simplemente basta con instalarla y el usuario puede empezar a realizar las prácticas del curso sin necesidad del equipo real y de cables.

6.2. Limitaciones

Las limitaciones en esta tesis están relacionadas con la implementación de la herramienta de simulación, una de ellas es la selección de cada una de las acciones o comandos que realiza la herramienta de simulación. En esa selección algunos comandos de configuración de los conmutadores quedan fuera de la implementación. La razón para dicha selección es que los conmutadores reales cuentan con diferentes comandos que inclusive en la vida real no son utilizados. Adicionalmente algunos comandos más no son utilizados en los cuatro semestres del CCNA de Cisco, por lo que, obviamente no sería de utilidad implementarlos y en algunos casos sería demasiado complicado.

Otra limitante es el hecho de que la red en la cual se encuentra el conmutador es fija, es decir, siempre se cuenta con la misma topología. Por lo tanto, cada vez que se quiera practicar, se hará sobre la misma topología y no existirá gran variedad de elementos y topologías para poder experimentar con diferentes escenarios que se puedan presentar.

6.3. Trabajos futuros

De acuerdo a los diferentes números de limitaciones que se mencionaron con anterioridad, es necesario realizar diferentes propuestas para extender el trabajo realizado y poder contar con una mejora en todos los aspectos de la herramienta de simulación.

A continuación se mencionan los diferentes trabajos que pueden ser realizados:

- **Extender el número de comandos.** Se pueden agregar diferentes características o comandos que son útiles para tener una herramienta óptima en cuanto a funcionamiento. Esos comandos o características pueden ser aquellos como el ver las configuraciones del conmutador por medio de un “browser” en Internet, poder analizar cada uno de los puertos, sus contraseñas, entre otras características. Además, puede ser incluido el comando *telnet*, que puede ser utilizado para tener acceso a otro dispositivo de la red desde cualquier punto de la misma.
- **Aumentar el número de topologías.** Esta característica es necesaria para poder realizar diferentes prácticas y no solamente centrarse en una sola. Inclusive puede ser incluido otro conmutador para que pueda ser configurado en una topología diferente y existir comunicación entre los diferentes equipos de interconexión que se encuentren en dicha topología.
- **Mejorar la herramienta de simulación.** La herramienta de simulación puede ser mejorada para trabajar en el curso de CCNP de Cisco, agregando nuevas características y que trabaje en modo línea de comando y no en modo de menú. Puede ser implementada para trabajar de esa forma y ejecutar diferentes comandos que son aplicados en modo línea de comando en el séptimo semestre del curso del CCNP de Cisco. Así, la herramienta de simulación podrá estar más completa porque puede ser funcional para los cursos de CCNA y CCNP de Cisco.

ANEXO A - SONDEO ACERCA DEL USO DE CONMUTADORES

Para conocer la opinión acerca de la herramienta de simulación, se realizó una encuesta a un grupo de alumnos que han cursado o cursan el tercer semestre de los cursos del CCNA de Cisco.

La encuesta es la siguiente:

Sexo _____ Edad _____

1. ¿Sabes la función que tiene un simulador?

Sí _____

No _____

2. ¿Alguna vez has utilizado en algún laboratorio los conmutadores Cisco?

Sí _____

No _____

3. ¿Conoces algún simulador de conmutadores que te permita ejecutar diferentes comandos similares a los reales?

Sí _____ Escribe el nombre _____

No _____

4. ¿Cuáles de las siguientes características crees que sean importantes en un simulador de conmutadores?

Economía (es completamente gratuita) _____

Movilidad (puede ser trasladada de un lugar a otro) _____

Portabilidad (puede ser utilizada en diferentes sistemas operativos) _____

Disponibilidad (puede hacer uso del simulador en cualquier momento) _____

Otra _____

5. De los siguientes elementos, ¿Cuál(es) crees que son importantes para que se agreguen a un simulador de conmutadores?

Menús en la interfaz _____

Utilización de iconos _____

Ayuda en línea _____

Impresión de la pantalla de interfaz _____

Otros _____

6. ¿Crees que pueda ser de ayuda para realizar las prácticas del laboratorio de Cisco el simulador de conmutadores?

Si _____

No _____

¿Por qué?

7. ¿Si existiera una herramienta de simulación de conmutadores la utilizarías para practicar los laboratorios de Cisco relacionados a los conmutadores?

Si _____

No _____

8. ¿Crees que el utilizar la herramienta de simulación de conmutadores pueda mejorar tu desempeño al realizar las prácticas en el laboratorio?

Si _____

No _____

¿Por qué?

¡Muchas gracias!

ANEXO B - CUESTIONARIO DE USABILIDAD Y FUNCIONALIDAD

Para conocer la opinión acerca de la herramienta de simulación, se realizó una encuesta a un grupo de alumnos que han cursado o cursan el tercer semestre de los cursos del CCNA de Cisco.

La encuesta es la siguiente:

Sexo _____ Edad _____

Califica los siguientes aspectos del simulador que acabas de probar:

5 = Muy malo 6 = Malo 7 = Regular 8 = Bueno 9 = Muy Bueno 10 = Excelente}

FUNCIONALIDAD

Aspecto	5	6	7	8	9	10
a) Valida correctamente las entradas de los usuarios						
b) Actualiza los cambios realizados en cada una de las opciones						
c) Funcionaron cada uno de los comandos utilizados						
d) Calificación final en el aspecto funcional						

e) ¿Qué cambios le harías en el aspecto funcional?

USABILIDAD

Aspecto	5	6	7	8	9	10
a) Amigable						
b) Robusta						
c) Estable						
e) Similitud con un conmutador real						
f) Calificación final en la facilidad de uso						

g) ¿Qué cambios le harías en el aspecto de usabilidad?

¡Muchas gracias!

ANEXO C - PRÁCTICA DE PRUEBA

Con el fin de evaluar el funcionamiento del prototipo creado para esta tesis, se diseñó la práctica de prueba mostrada en este Anexo con la que alumnos voluntarios probarían el Simulador y llenarían una forma o cuestionario (mostrado en el Anexo B) en el que expresarían su opinión acerca de él, enfocándose en dos aspectos muy importantes: el aspecto de usabilidad y el aspecto de funcionalidad.

Primera Práctica de Prueba

Paso 1 – Da doble click en el icono de la aplicación y espera a que se este ejecutando y da un clic sobre la caja de texto.

1. ¿Cuál es el número del modelo del conmutador?

2. ¿Cuál es la dirección Ethernet del conmutador?

3. ¿Cuál es la revisión y el tipo del firmware del conmutador?

Paso 2 – Configura el conmutador para que opere como fragment free.

Selecciona la opción [S] System del menú principal y revisa la configuración que existe en el menú.

4. ¿Cuál es el modo actual en el que el conmutador se encuentra configurado?

5. ¿Cuál opción del menú permite cambiar el modo?

6. ¿Qué opciones están disponibles?

Paso 3 – Configura el puerto para full duplex. Entrar a [P] Port Configuration, seleccionar el puerto 1.

7. ¿Está habilitado full duplex? ¿Qué opción es utilizada para habilitarlo?

Paso 4 – Utilizando las opciones del menú de VLAN, configura las VLAN's.

9. Verifica la configuración de la VLAN por default seleccionando la opción [E] **VLAN Membership**. ¿Cuál es el número actual de puertos en la VLAN 1?

10. Asigna los puertos del 7-12 a la VLAN 2. Lista los pasos que se requieren para hacerlo.

11. Verifica la lista nuevamente. ¿Qué cambio observas?

¡Muchas gracias!

ANEXO D – LICENCIA GPL

PREAMBULO

Los contratos de licencia de la mayor parte del software están diseñados para quitarle su libertad de compartir y modificar dicho software. En contraste, la "GNU General Public License" pretende garantizar su libertad de compartir y modificar el software "libre", esto es para asegurar que el software es libre para todos sus usuarios. Esta licencia pública general se aplica a la mayoría del software de la "FSF Free Software Foundation" (Fundación para el Software Libre) y a cualquier otro programa de software cuyos autores así lo establecen. Algunos otros programas de software de la Free Software Foundation están cubiertos por la "LGPL Library General Public License" (Licencia Pública General para Librerías), la cual puede aplicar a sus programas también.

Cuando hablamos de software libre, nos referimos a libertad, no precio. Nuestras licencias "General Public Licenses" están diseñadas para asegurar que:

1. usted tiene la libertad de distribuir copias del software libre (y cobrar por ese sencillo servicio si así lo desea)
2. recibir el código fuente (o tener la posibilidad de obtenerlo si así lo desea)
3. que usted puede modificar el software o utilizar partes de el en nuevos programas de software libre
4. que usted esté enterado de que tiene la posibilidad de hacer todas estas cosas.

Para proteger sus derechos, necesitamos hacer restricciones que prohíban a cualquiera denegarle estos derechos o a pedirle que renuncie a ellos. Estas restricciones se traducen en algunas responsabilidades para usted si distribuye copias del software, o si lo modifica.

Por ejemplo, si usted distribuye copias de un programa, ya sea gratuitamente o por algún importe, usted debe dar al que recibe el software todos los derechos que usted tiene sobre el mismo. Debe asegurarse también que reciban el código fuente o bien que puedan obtenerlo si lo desean. Y por último debe mostrarle a esa persona estos términos para que conozca los derechos de que goza.

Nosotros protegemos sus derechos en 2 pasos: (1) protegiendo los derechos de autor del software y (2) ofreciéndole este contrato de licencia que le otorga permiso legal para copiar, distribuir y modificar el software.

Además, para la protección de los autores de software y la nuestra, queremos asegurarnos de que toda persona entienda que no existe ninguna garantía del software libre. Si el software es modificado por alguien y lo distribuye, queremos que quienes lo reciban sepan que la copia que obtuvieron no es la original, por lo que cualquier problema provocado por quien realizó la modificación no afectará la reputación del autor original.

Finalmente, cualquier programa de software libre es constantemente amenazado por las patentes de software. Deseamos evadir el peligro de que los re-distribuidores de un programa de software libre obtengan individualmente los derechos de patente con el fin de volver dicho programa propietario. Para prevenir esto, hemos dejado en claro que cualquier patente deberá ser licenciada para el uso libre de toda persona o que no esté licenciada del todo.

A continuación se describen con precisión los términos y condiciones para copiar, distribuir y modificar el software.

TERMINOS Y CONDICIONES PARA COPIA, MODIFICACION Y DISTRIBUCION

0. Esta licencia aplica a cualquier programa o trabajo que contenga una nota puesta por el propietario de los derechos del trabajo estableciendo que su trabajo puede ser distribuido bajo los términos de esta "GPL General Public License". El "Programa", utilizado en lo subsecuente, se refiere a cualquier programa o trabajo original, y el "trabajo basado en el Programa" significa ya sea el Programa o cualquier trabajo derivado del mismo bajo la ley de derechos de autor: es decir, un trabajo que contenga el Programa o alguna porción de él, ya sea íntegra o con modificaciones y/o traducciones a otros idiomas. De aquí en adelante "traducción" estará incluida (pero no limitada a) en el término "modificación", y la persona a la que se aplique esta licencia será llamado "usted".

Otras actividades que no sean copia, distribución o modificación no están cubiertas en esta licencia y están fuera de su alcance. El acto de ejecutar el programa no está restringido, y la salida de información del programa está cubierta sólo si su contenido constituye un trabajo basado en el Programa (es independiente de si fue resultado de ejecutar el programa). Si esto es cierto o no depende de la función del programa.

1. Usted puede copiar y distribuir copias fieles del código fuente del programa tal como lo recibió, en cualquier medio, siempre que proporcione de manera conciente y apropiada una nota de derechos de autor y una declaración de no garantía, además de mantener intactas todas las notas que se refieran a esta licencia y a la ausencia de garantía, y que le proporcione a las demás personas que reciban el programa una copia de esta licencia junto con el Programa.

Usted puede aplicar un cargo por el acto físico de transferir una copia, y ofrecer protección de garantía por una cuota, lo cual no compromete a que el autor original del Programa responda por tal efecto.

2. Usted puede modificar su copia del Programa o de cualquier parte de él, formando así un trabajo basado en el Programa, y copiar y distribuir tales modificaciones o bien trabajar bajo los términos de la sección 1 arriba descrita, siempre que cumpla con las siguientes condiciones:

1. Usted debe incluir en los archivos modificados notas declarando que modificó dichos archivos y la fecha de los cambios.
2. Usted debe notificar que ese trabajo que distribuye contiene totalmente o en partes al Programa, y que debe ser licenciado como un conjunto sin cargo alguno a cualquier otra persona que reciba sus modificaciones bajo los términos de esta Licencia.
3. Si el programa modificado lee normalmente comandos interactivamente cuando es ejecutado, usted debe presentar un aviso, cuando el programa inicie su ejecución en ese modo interactivo de la forma más ordinaria, que contenga una noticia de derechos de autor y un aviso de que no existe garantía alguna (o que sí existe si es que usted la proporciona) y que los usuarios pueden redistribuir el programa bajo esas condiciones, e informando al usuario como puede ver una copia de esta Licencia. (Excepción: si el programa en sí es interactivo pero normalmente no muestra notas, su trabajo basado en el Programa no tiene la obligación de mostrar tales notas)

Estos requerimientos aplican al trabajo modificado como un todo. Si existen secciones identificables de tal trabajo que no son derivadas del Programa original, y pueden ser razonablemente consideradas trabajos separados e independientes como tal, entonces esta Licencia y sus términos no aplican a dichas secciones cuando usted las distribuye como trabajos separados. Pero cuando usted distribuye las mismas secciones como parte de un todo que es un trabajo basado en el Programa, la distribución del conjunto debe ser bajo los términos de esta Licencia, cuyos permisos para otras personas que obtengan el software se extienden para todo el software, así como para cada parte de él, independientemente de quién lo escribió.

No es la intención de esta sección de reclamar derechos o pelear sus derechos sobre trabajos hechos enteramente por usted, en lugar de eso, la intención es ejercer el derecho de controlar la distribución de los trabajos derivados o colectivos basados en el Programa. Adicionalmente, el simple agregado de otro trabajo NO basado en el Programa al Programa en cuestión (o a un trabajo basado en el Programa) en algún medio de almacenamiento no pone el otro trabajo bajo el alcance de esta Licencia.

3. Usted puede copiar y distribuir el Programa (o un trabajo basado en él, bajo la Sección 2) en código objeto o en forma de ejecutable bajo los términos de las secciones 1 y 2 arriba descritas siempre que cumpla los siguientes requisitos:

1. Acompañarlo con el correspondiente código fuente legible por la máquina, que debe ser distribuido bajo los términos de las secciones 1 y 2 y en un medio comúnmente utilizado para el intercambio de software, o
2. Acompañarlo con una oferta escrita, válida por al menos 3 años y para cualquier persona, por un cargo no mayor al costo que conlleve la distribución física del código fuente correspondiente en un medio comúnmente utilizado para el intercambio de software, o

3. Acompañarlo con la información que usted recibió sobre la oferta de distribución del código fuente correspondiente. (Esta alternativa está permitida sólo para distribución no-comercial y sólo si usted recibió el Programa en código objeto o en forma de ejecutable con tal oferta de acuerdo a la sub-sección b anterior)

El código fuente de un trabajo significa la forma preferida de hacer modificaciones al mismo. Para un trabajo ejecutable, un código fuente completo significa todo el código fuente de todos los módulos que contiene, mas cualquier archivo de definición de interfaces, mas los programas utilizados para controlar la compilación y la instalación del ejecutable.

Sin embargo, como excepción especial, no se requiere que el código fuente distribuido incluya cualquier cosa que no sea normalmente distribuida con las componentes mayores (compilador, kernel, etc.) del sistema operativo en el cual el ejecutable corre, a menos de que una componente en particular acompañe al ejecutable.

Si la distribución del ejecutable o del código objeto se hace ofreciendo acceso a copiar desde un lugar designado, entonces el ofrecer acceso equivalente para copiar el código fuente desde el mismo lugar se considera distribución del código fuente, aunque las demás personas no copien el código fuente junto con el código objeto.

4. Usted no puede copiar, modificar, sub-licenciar ni distribuir el Programa a menos que sea expresamente bajo esta Licencia, de otra forma cualquier intento de copiar, modificar, sub-licenciar o distribuir el programa es nulo, y automáticamente causará la pérdida de sus derechos bajo esta Licencia. Sin embargo, cualquier persona que haya recibido copias o derechos de usted bajo esta Licencia no verán terminadas sus Licencias ni sus derechos perdidos mientras ellas continúen cumpliendo los términos de esta Licencia.

5. Usted no está obligado a aceptar esta Licencia, dado que no la ha firmado. Sin embargo, nada le otorga el permiso de modificar o distribuir el Programa ni sus trabajos derivados. Estas acciones están prohibidas por la ley si usted no acepta esta Licencia. Sin embargo, modificando o distribuyendo el Programa (o cualquier trabajo basado en el Programa) indica su aceptación de esta Licencia y de todos sus términos y condiciones para copiar, distribuir o modificar el Programa y/o trabajos basados en el.

6. Cada vez que usted redistribuye el Programa (o cualquier trabajo basado en el Programa), la persona que lo recibe automáticamente recibe una licencia del autor original para copiar, distribuir o modificar el Programa sujeto a estos términos y condiciones. Usted no puede imponer ninguna restricción adicional a las personas que reciban el Programa sobre los derechos que en esta Licencia se les otorga. Usted no es responsable de forzar a terceras personas en el cumplimiento de esta Licencia.

7. Si como consecuencia de un veredicto de un juzgado o por el alegato de infringir una patente o por cualquier otra razón (no limitado solo a cuestiones de patentes) se imponen condiciones sobre usted que contradigan los términos y condiciones de esta Licencia, éstas no le excusan de los términos y condiciones aquí descritos. Si usted no puede distribuir el

producto cumpliendo totalmente con las obligaciones concernientes a la resolución oficial y al mismo tiempo con las obligaciones que se describen en este contrato de Licencia, entonces no podrá distribuir más este producto. Por ejemplo, si una licencia de patente no permitirá la distribución del Programa de forma libre de regalías (sin pago de regalías) por parte de quienes lo reciban directa o indirectamente, entonces la única forma de cumplir con ambas obligaciones es renunciar a la distribución del mismo.

Si cualquier parte de esta sección resulta inválida, inaplicable o no obligatoria bajo cualquier circunstancia en particular, la tendencia de esta es a aplicarse, y la sección completa se aplicará bajo otras circunstancias.

La intención de esta sección no es la de inducirlo a infringir ninguna ley de patentes, ni tampoco infringir algún reclamo de derechos, ni discutir la validez de tales reclamos; esta sección tiene el único propósito de proteger la integridad del sistema de distribución del software libre, que está implementado por prácticas de licencia pública. Mucha gente ha hecho generosas contribuciones a la amplia gama de software distribuido bajo este sistema favoreciendo así la constante aplicación de este sistema de distribución; es decisión del autor/donador si su Programa será distribuido utilizando este u otro sistema de distribución, y la persona que recibe el software no puede obligarlo a hacer ninguna elección en particular.

Esta sección pretende dejar muy en claro lo que se cree que será una consecuencia del resto de esta Licencia.

8. Si la distribución y/o el uso del Programa se restringe a algunos países ya sea por patentes, interfases protegidas por derechos de autor, el propietario original de los derechos de autor que ubica su Programa bajo esta Licencia puede agregar una restricción geográfica de distribución explícita excluyendo los países que aplique, dando como resultado que su distribución sólo se permita en los países no excluidos. En tal caso, esta Licencia incorpora la limitación como si hubiera sido escrita en el cuerpo de esta misma Licencia.

9. La "FSF Free Software Foundation" puede publicar versiones nuevas o revisadas de la "GPL General Public License" de uno a otro momento. Estas nuevas versiones mantendrán el espíritu de la presente versión, pero pueden diferir en la inclusión de nuevos problemas o en la manera de tocar los problemas o aspectos ya presentes.

Cada versión tendrá un número de versión que la distinga. Si el Programa especifica un número de versión para esta Licencia que aplique a él y "cualquier versión subsecuente", usted tiene la opción de seguir los términos y condiciones de dicha versión o de cualquiera de las posteriores versiones publicadas por la "FSF". Si el programa no especifica una versión en especial de esta Licencia, usted puede elegir entre cualquiera de las versiones que han sido publicadas por la "FSF".

10. Si usted desea incorporar partes del Programa en otros Programas de software libre cuyas condiciones de distribución sean distintas, deberá escribir al autor solicitando su autorización. Para programas de software protegidas por la "FSF Free Software

Foundation", deberá escribir a la "FSF" solicitando autorización, en ocasiones hacemos excepciones. Nuestra decisión será guiada por dos metas principales:

- mantener el estado de libertad de todos los derivados de nuestro software libre
- promover el uso comunitario y compartido del software en general

NO EXISTE GARANTIA ALGUNA

11. DEBIDO A QUE EL PROGRAMA SE OTORGA LIBRE DE CARGOS Y REGALIAS, NO EXISTE NINGUNA GARANTIA PARA EL MISMO HASTA DONDE LO PERMITA LA LEY APLICABLE. A EXCEPCION DE QUE SE INDIQUE OTRA COSA, LOS PROPIETARIOS DE LOS DERECHOS DE AUTOR PROPORCIONAN EL PROGRAMA "COMO ES" SIN NINGUNA GARANTIA DE NINGUN TIPO, YA SEA EXPLICITA O IMPLICITA, INCLUYENDO, PERO NO LIMITADA A, LAS GARANTIAS QUE IMPLICA EL MERCADEO Y EJERCICIO DE UN PROPOSITO EN PARTICULAR. CUALQUIER RIESGO DEBIDO A LA CALIDAD Y DESEMPEÑO DEL PROGRAMA ES TOMADO COMPLETAMENTE POR USTED. SI EL SOFTWARE MUESTRA ALGUN DEFECTO, USTED CUBRIRA LOS COSTOS DE CUALQUIER SERVICIO, REPARACION O CORRECCION DE SUS EQUIPOS Y/O SOFTWARE QUE REQUIERA.

12. EN NINGUN CASO NI BAJO NINGUNA CIRCUNSTANCIA EXCEPTO BAJO SOLICITUD DE LA LEY O DE COMUN ACUERDO POR ESCRITO, NINGUN PROPIETARIO DE LOS DERECHOS DE AUTOR NI TERCERAS PERSONAS QUE PUDIERAN MODIFICAR Y/O REDISTRIBUIR EL PROGRAMA COMO SE PERMITE ARRIBA, SERAN RESPONSABLES DE LOS DAÑOS CORRESPONDIENTES AL USO O IMPOSIBILIDAD DE USAR EL PROGRAMA, SIN IMPORTAR SI SON DAÑOS GENERALES, ESPECIALES, INCIDENTALS O CONSEQUENTES CORRESPONDIENTES AL USO O IMPOSIBILIDAD DE USAR EL PROGRAMA (INCLUYENDO PERO NO LIMITADO A LA PERDIDA DE INFORMACION O DETERIORO DE LA MISMA AFECTANDOLO A USTED, A TERCERAS PERSONAS QUE SEA POR FALLAS EN LA OPERACION DEL PROGRAMA O SU INTERACCION CON OTROS PROGRAMAS) INCLUSIVE SI TAL PROPIETARIO U OTRAS PERSONAS HAYAN SIDO NOTIFICADAS DE TALES FALLAS Y DE LA POSIBILIDAD DE TALES DAÑOS.

Referencias Bibliográficas

- [1] Dimitris N. Chorofas. *Visual Programming Technology*. McGraw-Hill. 1997.
- [2] Nan C. Shu. *Visual Programming*. Van Nostrand Reinhold Company. 1988.
- [3] Marat Boshernitsan, Michael Downes. *Visual Programming Languages: A Survey*. Computer Science Division, University of California, Berkeley. December 1997.
- [4] Clark, Kennedy, and Kevin Hamilton. *Cisco LAN Switching*. Indianapolis: Cisco Press, August 1999.
- [5] CNAP Curriculum, semestre 3, 2000.
- [6] John B. Black, Will Thalheimer, Hilary Wilder, Donna de Soto, Paul Picard. *Constructivist Design of Graphic Computer Simulations*. Teachers College, Columbia University, 2000. <http://www.ilt.columbia.edu/publications/cdgcs.html>
- [7] Donald R. Gentner, Jonathan Grudin. *Design Models for Computer-Human Interfaces*. Sun Microsystems, University of California, Irvine, Junio, 1996.
- [8] W. Mackay, *Triggers and Barriers to Customizing Software*. Proc. Computer-Human Interface '91, ACM Press, New York, 1991, pp. 153-160.
- [9] S.R Levine and S.F. Ehrlich, *The Freestyle System: A Design Perspective*, in Human – Machine Interactive Systems, A. Klinger, ed., Plenum, New York, 1991, pp. 3-21.
- [10] George M. Piskurich, *Rapid Instructional Design: Learning ID Fast and Right*, Pfeiffer, February 2000.
- [11] Mario Alberto Moreno Rocha, MSc IT, *Diseño de Software Interactivo Centrado en el Usuario*, Universidad Tecnológica de la Mixteca, 2000
- [12] J. Bravo, M. Ortega, M. Prieto, *Entorno de Simulación en la Educación a Distancia*, Departamento de Informática – UCLM, 1998.
- [13] L. Miguel Encarnacao, Robert J. Barton III, Jan Jungclaus, *A Web-based Distributive Simulator for the ALVIN Deep Submersible Vehicle – Innovative Approaches and Early Experiences*, Fraunhofer CRCG, Inc., 2001.
- [14] Jacques Delors, *La educación encierra un tesoro*, UNESCO, publicado en la UNESCO en 1996

- [15] M. David Merrill, Leston Drake, Mark J. Lacy, Jean Pratt & the ID₂ Research Group, *Reclaiming Instructional Design*, Utah State University, 1992.
- [16] Michael L. Dertouzos, *The Unfinished Revolution*, Harper Collins Publishers, Inc. 2001.
- [17] María José Gil Larrea, Jorge González Barturen, Universidad de Deusto-Facultad de Ingeniería Dpto. Ingeniería del Software, *Un simulador didáctico para el aprendizaje infantil: amigo*. 1994.
- [18] Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale, *Human-Computer Interaction*, Second Edition, Prentice Hall Europe, 1998.
- [19] F. Clymer, *Treasury of Early American Automobiles*, McGraw-Hill, New York, 1950.
- [20] Sierra, E. A. *Examining the Suitability of the Pattern Oriented Paradigm in the Educational Software Design Process*. In Press. Institute for Learning Technologies, Columbia University. 1999.
- [21] Alexander, C. *A Timeless Way of Building*. Oxford University Press. Oxford. England. 1999.
- [22] Sierra, E. A. *A Cognitivist Instructional Approach applied to the design of intelligent tutoring systems*, Proceedings of the Argentine Symposium on Artificial Intelligence, 221-232. 1999.
- [23] Duffy, M. & Jonassen, H. *Constructivism and the technology for instruction: a conversation*. Lawrence Erlbaum Associates. Washington. USA. 1982.
- [24] Wilson, B. G. *Constructivist Learning Environments: Case Studies in Instructional Design*. Educational Technology Publications. Englewood Cliffs, New Jersey. 1996.
- [25] Enrique Ariel Sierra, Alejandro Hoissian, Ramón Gracia-Ramírez, *Sistemas expertos que recomiendan estrategias de instrucción. Un modelo para su desarrollo*, Departamento Electrotecnia Facultad de Ingeniería Universidad Nacional del Comahue. 1997.
- [26] Vick, Ramamoorthy. *Handbook of Software Engineering*. Van Nostrand Reinhold Company, 1984.
- [27] Roger S. Pressman. *Software Engineering. A Practitioner's Approach*. Third Edition. McGraw-Hill. 1992.
- [28] <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/vclang/html/vcsmprstruct.asp>

- [29] Islamabad Businessline, *Free software aids BSNL directory printing*, Biblioteca digital, Mar 18, 2003.
- [30] William M. Bulkeley and Rebecca Buckman, *Microsoft Wages Quiet Campaign Against Free Software*, Wall Street Journal; New York, N.Y.; Dec 9, 2002.
- [31] Peter Wayner, *Free Software Revolution*, Upside; Foster City; Jul 2000.
- [32] Esther Schindler, *Obtaining free software and digital tools online at SourceForge*. Inside the Internet; Louisville; Jan 2003.
- [33] <http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/art22.asp>, 2003
- [34] <http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/art40.asp>, 2003
- [35] Dr. Ramón Brena. *Autómatas y Lenguajes, Un enfoque de diseño*. Instituto Tecnológico de Monterrey, Campus Monterrey. Verano 2003.
- [36] Cataldi, Z.1, Lage, F.2, Pessacq, R.3 y García Martínez, R.4, *Ingeniería de software educativo*, 1. Laboratorio de Sistemas Operativos y Base de Datos. Departamento de computación. Facultad de Ingeniería UBA. 2. Facultad de Ingeniería. UNLP. 3. Centro de Ingeniería del Software e Ingeniería del Conocimiento (CAPIS) ITBA. 4. Laboratorio de Sistemas Inteligentes. Departamento de Computación. Facultad de Ingeniería UBA. 1999.
- [37] Gallego D. y Alonso C. *Multimedia*. UNED. España. 1997.
- [38] Galvis A. *Software educativo multimedia aspectos críticos del ciclo de vida*. Revista Brasileira de Informática no Educação. Sociedade Brasileira de Computação, 1996.
www.janus.ufse.br:1085/revista/nr1/galvis_p.htm
- [39] Marquès P. *Metodología para la elaboración de software educativo en Software Educativo. Guía de uso y metodología de diseño*. Barcelona Estel. 1995.
www.xtec.es/~pmarques, www.doe.d5.ub.es
- [40] IEEE. *Standard for Developing Software Life Cycle Process. IEEE std. 1074-1991* Nueva York. IEEE computer Society. 1991.
- [41] ISO. *ISO/IEC 12701-1, Software Life-cycle Process*. 1994.
- [42] Boehm B. *Software Engineering Economics*, Englewood Cliffs, Nueva Jersey. 1981.
- [43] Ronnie Johansson, *Intelligent Motion Planning and Control of a Multi-Robot System*, The Royal Institute of Technology (KTH), Stockholm, Sweden, April 2000 - October 2000
- [44] Frederick Betz, *Strategic Technology Management*, International Editions, McGraw-Hill, 1994.

- [45] Cisco Networking Academy Program, 2003
<http://cisco.netacad.net/public/academy/index.html>
- [46] Edward R. Tufte. *Visual Explanations. Images and Quantities, Evidence and Narrative*. Graphics Press. 1997.
- [47] Ivan Sutherland, I. B. *SKETCHPAD, A man-machine graphical communication system*. In Proceedings of the Spring Joint Computer Conference, pp. 329-346, 1963.
- [48] Smith, D. C. *PYGMALION: A Creative Programming Environment*. PhD dissertation, Stanford University, 1975.
- [49] Rafael Bisquerra. *Métodos de Investigación en la Educación*. CEAC, 1989.
- [50] International Standard ISO 9241-11, *Ergonomic requirements for office work with visual display terminals (VDTs), Part 11: Guidance on usability*, First Edition, March 1998.

