

MÉTODO DE ADMINISTRACIÓN Y CONSTRUCCIÓN DE LABORATORIOS  
INTERACTIVOS REMOTOS ORIENTADOS AL TELECONTROL

JUAN PABLO GARCÍA  
961670

CESAR AUGUSTO SÁNCHEZ  
950825

Tesis para optar al título de  
Magister en Ciencias Computacionales con énfasis en Redes de computadores

Director:  
CÉSAR GERMÁN CASTELLANOS DOMINGUEZ  
Phd en telecomunicaciones

Instituto Tecnológico de Estudios Superiores de Monterrey  
Universidad Autónoma de Bucaramanga  
Maestría en Ciencias Computacionales  
Manizales, Colombia  
2005

## TABLA DE CONTENIDO

LISTA DE FIGURAS .....	iv
1. INTRODUCCION .....	5
2. OBJETIVOS .....	7
2.1 Objetivo General.....	7
2.2 Objetivos Específicos.....	7
3. PLATAFORMAS ADMINISTRATIVAS DE CURSOS VIRTUALES.....	8
3.1 LA EDUCACIÓN VIRTUAL EN COLOMBIA .....	8
3.2 PLATAFORMAS DE ADMINISTRACIÓN .....	10
3.3 ESTÁNDARES UTILIZADOS EN EDUCACIÓN VIRTUAL.....	12
3.4 CONCEPTOS DEL ESTÁNDAR AICC.....	13
3.5 MODELOS PEDAGÓGICOS .....	15
3.5.1 Teoría de Gestalt .....	15
3.5.2 Teoría Cognitiva .....	16
3.5.3 El Constructivismo .....	16
3.6 CARACTERÍSTICAS DE LAS PLATAFORMAS DE ADMINISTRACIÓN DE CURSOS VIRTUALES.....	20
3.6.1 Características de Moodle.....	20
3.6.2 Características de WBT Manager .....	27
3.6.3 Características de BlackBoard.....	29
4. LABORATORIOS INTERACTIVOS REMOTOS.....	37
4.1 LABORATORIOS REMOTOS. ANTECEDENTES.....	37
4.2 TELEMETRÍA Y TELECONTROL .....	39
4.3 ESTÁNDARES Y PROTOCOLOS DE COMUNICACIÓN .....	39
4.3.1 RS232.....	40
4.3.2 RS422 / RS485 .....	40
4.3.3 GPIB.....	44
4.4 LENGUAJES DE PROGRAMACIÓN DE INSTRUMENTOS.....	46
4.4.1 Lenguaje estándar SCPI (Standard Commands for Programmable Instruments) .....	46
5. MÉTODO DE ADMINISTRACIÓN y construcción DE LABORATORIOS INTERACTIVOS REMOTOS.....	48
5.1 Modelo de Administración.....	50
5.2 Modelo de comunicación .....	62
5.3 Modelo de sesión.....	65
5.4 Modelo de evaluación.....	73
5.5 Modelo de Construcción de LIRs.....	76
6. modelo de construcción de LIRs.....	77
6.1 Diseño de un laboratorio interactivo remoto.....	77
6.2 Construcción del laboratorio .....	78
6.2.1 Ambientes Web (servidores).....	79
6.2.2 Bases de Datos.....	79
6.2.3 Aplicaciones de ejecución sobre un navegador.....	80
6.2.4 Funciones de programación .....	80

6.2.5	Software de enlace entre interfaces de hardware.....	81
7.	MARCO EXPERIMENTAL .....	82
7.1	IMPLEMENTACIÓN DEL MODELO DE ADMINISTRACIÓN DE LABORATORIOS INTERACTIVOS REMOTOS (LIRs) .....	84
7.2	IMPLEMENTACIÓN DEL MODELO DE COMUNICACIÓN DE LABORATORIOS INTERACTIVOS REMOTOS (LIRs) .....	88
7.3	IMPLEMENTACIÓN DEL MODELO DE SESIÓN DE LABORATORIOS INTERACTIVOS REMOTOS (LIRs) .....	89
7.4	IMPLEMENTACIÓN DEL MODELO DE CONSTRUCCIÓN DE LABORATORIOS INTERACTIVOS REMOTOS (LIRs) .....	91
7.4.1	Diseño del Laboratorio .....	91
7.4.2	Construcción del LIR .....	95
7.5	Pruebas de Rendimiento .....	98
7.5.1	Velocidad de transmisión de paquetes .....	99
7.5.2	Pruebas de Tráfico y Congestión de la Red.....	102
7.6	DISCUSIÓN DE RESULTADOS.....	108
8	CONCLUSIONES .....	110
9	BIBLIOGRAFIA Y REFERENCIAS.....	112
	ANEXOS .....	117
	ANEXO A. TABLA CON PRODUCTOS COMERCIALES CERTIFICADOS POR LA AICC .....	118
	ANEXO B. CASOS DE USO, DIAGRAMAS DE INTERACCIÓN Y CONTRATOS DEL MODELO DE ADMINISTRACIÓN.....	120
	ANEXO C. DUMP DE LA BASE DE DATOS SALIR.....	140
	ANEXO D. CODIGO ARCHIVO: SESION_INICIO_EST.PHP .....	144
	ANEXO E. CODIGO ARCHIVO: HACER_PRACTICA.PHP (SEGMENTO).....	146
	ANEXO F. CODIGO ARCHIVO: INICIO.PHP .....	148
	ANEXO G. CODIGO ARCHIVO: REGISTRO_NOTAS.PHP .....	149
	ANEXO H. PRUEBAS DE RENDIMIENTO.....	150
	ANEXO I. INTERFASES DEL SISTEMA DE ADMINISTRACION SALIR.....	164
	ANEXO J. INTERFASES Y DESCRIPCIÓN DEL FUNCIONAMIENTO DEL LABORATORIO INTERACTIVO REMOTO .....	170
	ANEXO K. APLICATIVO JAVA (APPLET) DEL LIR (COMPILADOR E INTERPRETE DE COMANDOS) .....	175
	ANEXO L. ANALISIS DEL lir: "PRACTICA DE PROGRAMACIÓN Y CONTROL DE DISPOSITIVOS ROBOTICOS".....	194
	ANEXO M. DISEÑO DEL DISPOSITIVO MÓVIL PARA EL LIR: "PRACTICA DE PROGRAMACIÓN Y CONTROL DE DISPOSITIVOS ROBOTICOS".....	199

## LISTA DE FIGURAS

---

Figura 1. Estructura del protocolo AICC .....	14
Figura 2. Telecontrol de un brazo robótico.....	37
Figura 3. Osciloscopio Virtual.....	38
Figura 4. Aplicación multicargas.....	41
Figura 5. Aplicación Multipunto .....	42
Figura 6. Red RS-485.....	44
Figura 7. Modelos que soportan el Método de Administración y Construcción de un LIR orientado al telecontrol .....	48
Figura 8. Actores del Sistema.....	52
Figura 9. Diagrama de casos de uso Adminsitrador del Sistema .....	53
Figura 10. Diagrama de Casos de uso Tutor.....	54
Figura 11. Diagrama de Casos de Uso Estudiante .....	54
Figura 12. Diagrama Entidad Relación .....	56
Figura 13. Diagrama de comunicación para realizar una práctica Cliente-Módulo de administración-Módulo operativo Lir.....	65
Figura 14. Diagrama de interacción para realizar una práctica Cliente-Módulo de administración-Módulo operativo LIR.....	70
Figura 15. Diagrama de un LIR General.....	77
Figura 16. Diagrama Navegacional Administrador del Sistema .....	85
Figura 17. Manejo de Variables de Sesión.....	90
Figura 18. Diagrama del LIR de prueba implementado.....	93
Figura 19. Servidor Web.....	96
Figura 20. Esquema Applet.....	97
Figura 21. Esquema TCP/IP – RS232.....	97
Figura 22. Esquema diagrama Video.....	98
Figura 23. Tráfico: Cliente – LIR. Escenario 1 (una conexión) .....	103
Figura 24. Tráfico: Cliente – LIR. Escenario 1 (dos conexiones) .....	104
Figura 25. Tráfico: Cliente – LIR. Escenario 2 (una conexión) .....	105
Figura 26. Tráfico: Cliente – LIR. Escenario 2 (dos conexiones) .....	106
Figura 27. Tráfico: Cliente – LIR. Escenario 3 (Una conexión).....	107
Figura 28. Tráfico: Cliente – LIR. Escenario 3 (Dos conexiones).....	108

---

# 1. INTRODUCCION

---

El presente trabajo pretende aportar al mundo de la educación un método que sirva de guía para aquellas instituciones en donde el empleo de Laboratorios Interactivos Remotos – LIR - se vea como una realidad a implementar.

Así entonces, se debe entender que si alguien desea poner en marcha un LIR bajo este método, no debe olvidar que el sistema de administración debe estar implementado bajo el mismo método, de lo contrario no será posible poner en marcha todos los protocolos incluidos dentro de los diferentes modelos.

Lo aquí expuesto se centra en el diseño de una Metodología de Diseño de LIR, de manera que la administración sea asumida por un sistema independiente del módulo operativo de LIR y que permita la administración de otros LIR. Así entonces se expone todo lo que la metodología propone para el desarrollo del sistema de administración y de un módulo operativo, específicamente en la parte de la comunicación entre los dos sistemas mencionados; teniendo en cuenta que el sistema de administración además debe tener como básico el manejo de la agenda de los laboratorios, las sesiones de práctica y los productos dados por las mismas como notas, archivos entre otros. El resto de accesorios que se deseen incorporar al sistema de administración ya sean de comunicación (chats, foros, correos), calendarios, cronogramas, pizarras de notas, y demás se asumen desde este proyecto como complementos que no son tenidos en cuenta dentro de la metodología.

Igualmente en el capítulo dedicado al marco Experimental, se describe la manera como los autores del método desarrollan los correspondientes módulos para verificar y poner a prueba el método; aunque no se desconoce que deben realizarse más y variadas pruebas para robustecer el método y demostrar que es confiable.

El presente proyecto ha sido aceptado para ser presentado en el Simposio Iberoamericano de Educación, Cibernética e Informática (SIECI 2005), el cual se celebra

en el contexto de la 4ta conferencia Iberoamericana en Sistemas, Cibernética e Informática (CISCI 2005) y conjuntamente con la 3rd International Conference on Education and Information Systems, Technologies and Applications (EISTA '05), que tendrán lugar en Orlando Florida.

## 2. OBJETIVOS

---

### 2.1 OBJETIVO GENERAL

Investigar, analizar y proponer una metodología para la administración y construcción de LIR que incluya las funciones de acceso, control y seguridad de los laboratorios registrados que ofrezca a los desarrolladores de LIR y estudiantes un ambiente integrado y de fácil uso y alta velocidad, y hacer una implementación de la misma para mostrar su aplicación.

### 2.2 OBJETIVOS ESPECÍFICOS

Diseñar la metodología para la implementación del servidor de administración de LIR.

Diseñar la metodología para la construcción y montaje de un servidor operativo de LIR.

Implementar la metodología desarrollada para la puesta en funcionamiento de un servidor de administración de LIR sobre la LAN de la UAM (Universidad Autónoma de Manizales) y para la construcción de un LIR enfocado a la práctica de la programación de un móvil telecontrolado con el fin de verificar la metodología diseñada.

### 3. PLATAFORMAS ADMINISTRATIVAS DE CURSOS VIRTUALES

---

#### 3.1 LA EDUCACIÓN VIRTUAL EN COLOMBIA

La educación superior en Colombia ha venido desde hace unos años, explorando la enseñanza virtual o educación a distancia/virtual [1] como una posibilidad para las Universidades de expandir su área de cobertura hacia la globalización y salir del antiguo esquema regional que limita a estas instituciones, vistas como empresas, a un mercado muy pequeño, siguiendo ejemplos de países que han incursionado en el tema como México, USA, Canadá, España y Australia entre otros.

En 1989, el Instituto Tecnológico de Educación Superior de Monterrey (México), comienza su expansión en América Latina, ofreciendo maestrías a distancia utilizando las nuevas tecnologías de la época (Transmisión Satelital, Correo Electrónico y Chat Propio), como medios tecnológicos que soportaran el aprendizaje de sus contenidos.[2] Esta expansión, llegó a Colombia en 1992, en convenio con la Universidad Autónoma de Bucaramanga (UNAB), y unos años más tarde (1995), en convenio con las universidades pertenecientes a la Red José Celestino Mutis [3], ofrecieron programas de maestrías a distancia a través de clases satelitales ofrecidas por el ITESM.

En la actualidad, alrededor de 25 Instituciones de Educación superior, ofrecen programas de educación a distancia/virtual en Colombia [4], con planteamientos metodológicos y pedagógicos propios. Pero, la concepción de una propuesta metodológica definitiva que tenga en cuenta los avances y limitantes tecnológicos que imperan en la actualidad en el país, sigue siendo uno de los mayores desafíos que inquieta actualmente a los investigadores y estudiosos en educación. [5]

No sólo la metodología plantea retos en cuanto a la educación a distancia/virtual en Colombia, también están las restricciones tecnológicas que se presentan dada la baja penetración de Internet en los hogares colombianos, el bajo porcentaje de computadores en los hogares, y las restricciones de servicios de telecomunicaciones de

alta velocidad, en costos y cobertura [6]. Estas limitaciones ralentizan el proceso que de una u otra forma, pretende en términos generales llevar la educación superior a todos los rincones de Colombia.<sup>1</sup>

#### Penetración de Internet en América

<b>TOTALES REGIONALES</b>	Población (Est. 2004)	Usuarios Internet	Crecimiento 2000 - 2004	Penetración
Centro America	140,700,000	14,408,600	347.80%	10.20%
Suramerica	362,221,900	39,102,074	173.60%	10.80%
Caribe	38,853,900	2,420,300	332.60%	6.20%
SubTotal	541,775,800	50,930,974	209.50%	10.30%
Norteamerica	325,246,100	222,165,659	105.50%	68.30%
Total America	867,021,900	278,096,633	120.40%	31.70%

<b>POR PAÍS</b>	Población (Est. 2004)	Usuarios Internet	Crecimiento 2000 - 2004	Penetración
Estados Unidos	293,271,500	201,661,159	111.50%	68.80%
Canadá	31,846,900	20,450,000	61,0 %	64.20%
Bermudas	63,600	34,500	38.00%	54.20%
Barbados	264,800	100,000	1566.70%	37.80%
Groenlandia	56,900	20,000	12.40%	35.10%
Uruguay	3,428,900	1,190,120	221.70%	34.70%
Aruba	69,400	24,000	500.00%	34.60%
Bahamas	325,500	84,000	541.20%	25.80%
Chile	15,482,300	4,000,000	127.60%	25.80%
Islas Vírgenes (USA)	122,600	30,000	150.00%	24.50%
Jamaica	2,658,600	600,000	900.00%	22.60%
Granada	94,200	19,000	363.40%	20.20%
Costa Rica	4,205,700	800,000	220.00%	19.00%
Dominica	70,400	12,500	525.00%	17.80%
Puerto Rico	3,942,100	600,000	200.00%	15.20%
Argentina	37,740,400	5,600,000	124,0 %	14.80%
Guyana	869,100	125,000	4066.70%	14.40%
Antigua & Barbuda	70,700	10,000	100.00%	14.10%
México	102,797,200	12,250,000	351.60%	11.90%
Brasil	179,383,500	19,311,854	286.20%	10.80%
Belice	282,500	30,000	100.00%	10.60%
Trinidad & Tobago	1,301,700	138,000	38.00%	10.60%
Perú	27,639,000	2,850,000	14.00%	10.30%
Martinica	393,400	40,000	700.00%	10.20%
Venezuela	24,705,900	2,310,000	143.20%	9.30%
El Salvador	6,281,600	550,000	1275.00%	8.80%
Santa Lucía	165,500	13,000	333.30%	7.90%
Colombia	45,299,400	2,732,200	211.20%	6.00%

<sup>1</sup> InternetWorldStats presenta con regularidad las últimas estadísticas de penetración de internet en el mundo. A continuación relacionamos las cifras globales por cada región de América, al igual que las cifras por país clasificadas en orden de mayor a menor penetración.

<b>República Dominicana</b>	8,545,300	500,000	809.10%	5.90%
<b>Ecuador</b>	12,664,700	569,700	216.50%	4.50%
<b>Guadalupe</b>	444,900	20,000	150.00%	4.50%
<b>Surinam</b>	460,300	20,000	70.90%	4.30%
<b>Panamá</b>	3,042,800	120,000	166.70%	3.90%
<b>Guatemala</b>	11,917,800	400,000	515.40%	3.40%
<b>Bolivia</b>	8,879,600	270,000	125.00%	3.00%
<b>Honduras</b>	6,530,300	168,600	321.50%	2.60%
<b>Paraguay</b>	5,469,600	120,000	500.00%	2.20%
<b>Nicaragua</b>	5,642,100	90,000	80.00%	1.60%
<b>Cuba</b>	11,265,100	120,000	100.00%	1.10%
<b>Haití</b>	8,666,200	80,000	1233.30%	0.90%
<b>Subtotal</b>	<b>537,429,400</b>	<b>278,063,633</b>		<b>51.74%</b>
<b>Resto de Países</b>	<b>329,592,500</b>	<b>33,000</b>		<b>0.01%</b>

Fuente. Estadísticas DELTA. <sup>2</sup>

### Penetración de Internet en los Hogares Colombianos

<b>Ciudad</b>	<b>Tiene Computador</b>	<b>Se conecta a Internet</b>
Bogotá	13%	5%
Medellín	11%	6%
Cali	10%	7%
Barranquilla	7%	4%
Total Ciudades	11%	5%
Clase Alta	62%	42%
Clase Media Alta	21%	8%
Clase Media	4%	0.5%
Clase Baja	0.4%	0.0%

Fuente: Estadísticas DELTA.

## 3.2 PLATAFORMAS DE ADMINISTRACIÓN

Las plataformas de administración de cursos virtuales online son tan variadas, como las instituciones que las utilizan. En Latinoamérica y en particular en Colombia, las plataformas comerciales son las más empleadas para la administración de los cursos virtuales, siendo WebCT (creada por Murray Goldberg en la universidad de British Columbia – Canadá) la más utilizada (75%) [7], mientras que BlackBoard, interLearn, microCampus, Moodle y Lotus Notes Learning Space lo son pero en menor porcentaje. Los desarrollos propios, también cobran importancia en la mayoría de los países, principalmente en Argentina (67%). [8]

<sup>2</sup> <http://www.deltaasesores.com/esta/>

Cabe resaltar, que de acuerdo con el Maastricht McLuhan Institute [9] de Holanda, de 50 ambientes o plataformas electrónicas usadas en el mundo, revisaron nueve de las más difundidas y encontraron que, en cuanto a flexibilidad y soporte de ambientes de enseñanza y aprendizaje colaborativos, solamente cuatro estaban bien equipadas para el servicio de aprendizaje y enseñanza, particularmente en los países en desarrollo y eran además eficientes en los diferentes idiomas, estados o departamentos y en redes multimedia. Ellas fueron: WebCT, Blackboard, IntraLearn y TopClass [10].

Actualmente, se observa en Latinoamérica el fenómeno muy particular de la migración de plataformas comerciales hacia la versión GNU o libre de las mismas, desarrollos incentivados por curiosos en el tema como Martin Dougiamas, creador de la herramienta Moodle, que crece en popularidad en el mundo entero [11]. Estos desarrollos asombran por su flexibilidad e incentivan aún más su uso, por el atractivo económico que ofrece a las instituciones, ahorrando de su presupuesto varios miles de dólares que cuesta el licenciamiento de plataformas comerciales.

Los desarrollos propios de plataformas de administración de cursos virtuales, también se presentan en las instituciones en alguna etapa de su proceso de educación virtual, como ocurrió en la Universidad Autónoma de Manizales con su sistema ACME [12], que fue reemplazado posteriormente por la herramienta comercial y profesional WebCT y que finalmente cedió ante la tentación económica de Moodle.

Finalmente, la tendencia en plataformas de administración de cursos virtuales, se orienta hacia las que cumplan con los estándares y recomendaciones definidas por la AICC, IEEE y la IMS, como ejemplo está la aplicación WBT Manager [13] o la plataforma de IBM Lotus Learning Management System (LMS) [14], que cumple con dichas recomendaciones. De hecho, la estandarización de las herramientas tecnológicas es la ruta más apropiada que debe tomar la educación virtual no sólo en nuestro país, sino en el mundo entero.

WebCT, también está esforzándose para entrar en la estandarización con su última versión WebCT campus edición 4x y WebCT Vista 3x tratando de integrar herramientas de autoría en su proceso de publicación como módulos y BlackBoard con su nueva versión del BlackBoard Learning System, que cumple con la especificación SCORM 1.2 y maneja el concepto de Bloques Constructivos y Unidades de Aprendizaje.

### 3.3 ESTÁNDARES UTILIZADOS EN EDUCACIÓN VIRTUAL

Las entidades encargadas de la producción de estándares internacionales, se encuentran trabajando en el desarrollo de una especificación para la implementación de modelos de aprendizaje basados en e-learning o educación virtual, lo que garantiza el uso de normas establecidas por consenso general que contengan las especificaciones técnicas y de calidad mínimas requeridas, que permitan una competencia en condiciones de igualdad a los desarrolladores de plataformas de educación virtual.

Con la creación de un estándar, se logra fomentar el desarrollo de contenidos de una manera más organizada, se asegura el nivel de calidad técnica de los mismos y se facilita el intercambio entre instituciones y sus sistemas de información. Creando estándares abiertos, se evita también el monopolio de empresas de software que desarrollan plataformas administrativas de cursos virtuales.

En términos concretos, los objetivos de un estándar de educación virtual son: durabilidad, lo que significa contenidos con un tiempo estimado de uso más prolongado, interoperabilidad, que le da capacidad al contenido de interactuar con la plataforma de administración sobre la cual se encuentra, así sea diferente de la plataforma en la cual fue creado; y reusabilidad, que brinda la posibilidad de desintegrar el contenido para reutilizar partes que lo integran en la creación de nuevos contenidos.

Las ventajas y beneficios de una estandarización para las instituciones, permiten que lleguen incluso a intercambiar bloques de contenido o quizás a comercializar los mismos,

además, las búsquedas de información podrán extenderse a través de las plataformas, creando un repositorio gigantesco de información.

Actualmente, las instituciones que intentan desarrollar un estándar de educación virtual, hacen referencia a los siguientes puntos, como básicos en su concepción:

- Contenido: Estructura y Empaquetamiento
- Estudiantes: Almacenamiento e intercambio de información, habilidades, privacidad y seguridad.
- Interoperabilidad: Integración de componentes en una plataforma LMS (Learning Management System) e interoperabilidad entre diferentes plataformas.

Aún no se han definido estándares universales en educación virtual, pero ya existen varias especificaciones creadas por organizaciones especializadas como son: AICC (Aviation Industry CBT Committee), IEEE (Institute of Electrical and Electronics Engineers), el IMS Global Learning Consortium y el ADL (Advanced Distributed Learning).

La AICC, ha desarrollado pautas para la creación, entrega y validación de tecnologías de la información y son los creadores de la especificación AICC. La IEEE, creó el LTSC (Learning Technology Standards Committee) y además desarrolló el LOM (Learning Objects Metadata) basado en XML.

El IMS, es el consorcio que agrupa a vendedores, productores, desarrolladores y consumidores de e-learning y su objetivo es la creación de especificaciones que cumplan las exigencias y recomendaciones de la IEEE y la AICC. [15]

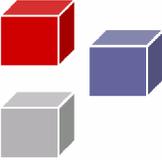
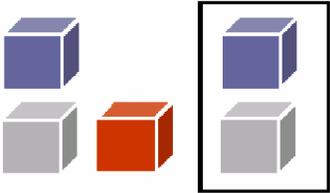
Finalmente, el ADL [16], es una iniciativa del gobierno de los Estados Unidos, encargados de integrar y armonizar las especificaciones de la AICC, IMS e IEEE y creadores del modelo SCORM (Sharable Content Object Reference Model) [17].

### 3.4 CONCEPTOS DEL ESTÁNDAR AICC [18]

Dado que la especificación y recomendación de la AICC, es el primer esfuerzo en cuanto a estandarización en educación virtual, es conveniente mencionar algunos elementos que la componen.

La recomendación AICC cuenta con los elementos definidos en la siguiente tabla [19].

Elementos básicos de la recomendación de la AICC para la creación de Contenidos virtuales

	AU (Assignable Unit)
	IB (Instructional Block)
	Course (Unidad Mínima Asignable)

La interoperabilidad de AICC se puede visualizar en la Figura 1.

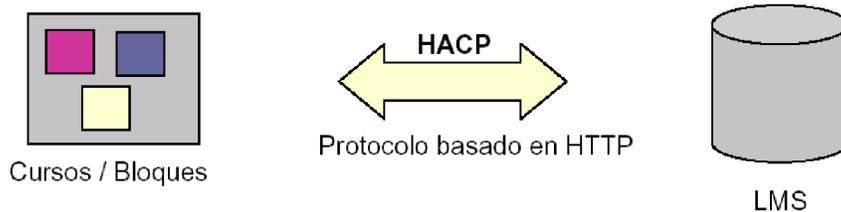


Figura 1. Estructura del protocolo AICC

De acuerdo con la recomendación, la estructura de un curso debe tener las siguientes características particulares:

- Se debe describir mediante archivos CIF (Course Interchange Files)
- Se deben manejar cuatro archivos fundamentales
  - .crs : contiene la información del curso
  - .des: maneja la descripción de cada unidad asignable al curso

- .au: son las unidades asignables
- .cst: es la jerarquía de las unidades asignables en el curso
- Se pueden manejar opcionalmente tres archivos que describen características avanzadas, que son:
  - .pre: prerequisites de cada unidad asignable
  - .cmp: requisitos para completar el curso
  - .ort: relaciones de los objetivos con los contenidos.

Las anteriores características, son las bases que deben seguir los creadores de software para publicación web, aplicados a plataformas de aprendizaje en línea, como son AuthorWare y Dreamweaver, entre otros, para integrarse sin inconvenientes con plataformas en línea, como son WebCT, BlackBoard y Moodle entre otras.

### 3.5 MODELOS PEDAGÓGICOS

Una de las cosas más importantes a la hora de hablar de educación virtual, es la definición e implantación de un modelo pedagógico a seguir. La construcción de programas virtuales o cursos online, no está dada únicamente por los aspectos tecnológicos. Debe existir una profunda reflexión pedagógica que soporte las actividades que se presenten en un programa. Octavio Henao Álvarez [20], haciendo referencia a Leflore [21], propone como modelos pedagógicos a seguir los que se describen a continuación, de los cuales algunos han sido implantados en varias universidades e instituciones del país como la Universidad Autónoma de Manizales<sup>3</sup> y el CES<sup>4</sup> entre otros.

#### 3.5.1 Teoría de Gestalt

Esta teoría está basada en la influencia que tiene la percepción sensorial en el aprendizaje. Utiliza la ventaja que ofrecen algunas características visuales que mejoran la comprensión del tema, tales como: el contraste, la simetría, la intensidad del estímulo, la proximidad y la sencillez.

---

<sup>3</sup> <http://www.uamvirtual.edu.co> Universidad Autónoma de Manizales. Virtual

<sup>4</sup> <http://www.cesvirtual.edu.co/> El Instituto de Ciencias de la Salud CES

Estos elementos permiten configurar los contenidos de una manera agradable a la visual del estudiante, dándose un efecto directo sobre el aprendizaje. Desde esta teoría podrían darse las siguientes recomendaciones para la construcción de un curso virtual:

- Utilizar fondos claros que no interfieran con la nitidez del texto ni de las imágenes
- Agrupar la información que tenga relación entre sí.
- Hacer una adecuada combinación de colores y de su intensidad.
- Incluir pocas animaciones y/o efectos visuales de los textos.
- La información siempre debe estar completa.
- Utilizar vocabulario sencillo en los temas nuevos. De no ser posible, habilitar un glosario donde el estudiante pueda consultar los términos no comprendidos.

### 3.5.2 Teoría Cognitiva

Afirma que gran parte del aprendizaje está dado gracias al desarrollo de mapas conceptuales y a la activación de mapas mentales previamente elaborados. Lo anterior obliga al docente virtual a utilizar medios que aumenten la capacidad de integrar nuevo conocimiento a esquemas previamente definidos por el estudiante. Es así como la utilización de ejemplos que ilustran conceptos y los ejercicios de simulación de la realidad, no solo cumplen con esa premisa sino que poseen un efecto motivador sobre la capacidad de aprendizaje del estudiante.

### 3.5.3 El Constructivismo

El aprendizaje se da en la medida que el estudiante participe activamente en su proceso educativo. Esta participación debe ser fomentada en la educación virtual, ya que la necesidad del estudiante de interactuar con sus docentes y compañeros es una de las características más importantes que definirán el logro de un aprendizaje significativo. La

formulación de problemas para su discusión en grupo exige del estudiante desarrollar capacidad de análisis y de crítica.

Para la definición de un modelo pedagógico, según Manuel Unigarro, [22] deben responderse los siguientes interrogantes, que constituyen las variables que interactuarán en el proceso educativo virtual:

- ¿Qué características específicas tienen los estudiantes a quienes van dirigidos los programas?
- ¿Qué preparación pedagógica y técnica poseen los docentes encargados de impartir este tipo de educación?
- ¿Cuáles son los objetivos de enseñanza que se proponen?
- ¿De que medios tecnológicos se dispone para la definición de las estrategias didácticas y las metodologías a emplear?
- ¿Cómo se entienden los procesos de evaluación y seguimiento?
- ¿Cuáles son los aspectos y/o procesos críticos?

La mayoría de las plataformas de administración de cursos virtuales, tratan de adaptarse a un modelo pedagógico como los anteriormente vistos, es así que herramientas como Moodle, utiliza un modelo pedagógico construccionista social [23]. Según Martin Dougiamas, creador de moodle, este modelo se puede explicar definiendo sus cuatro componentes subyacentes (constructivismo, construccionismo, construccionismo social y conectados y separados) de la siguiente manera:

#### a. Constructivismo.

Este punto de vista mantiene que la gente construye activamente nuevos conocimientos a medida que interactúa con su entorno.

Todo lo que usted lee, ve, oye, siente y toca se contrasta con su conocimiento anterior y si encaja dentro del mundo que hay en su mente, puede formar nuevo conocimiento que se llevará consigo. Este conocimiento se refuerza si puede usarlo con éxito en el

entorno que le rodea. No sólo es usted un banco de memoria que absorbe información pasivamente, ni se le puede transmitir conocimiento sólo leyendo algo o escuchando a alguien.

Esto no significa que no pueda aprender nada leyendo una página web o asistiendo a una lección. Es obvio que puede hacerlo; sólo indica que se trata más de un proceso de interpretación que de una transferencia de información de un cerebro a otro.

#### b. Construccinismo

El construccionismo explica que el aprendizaje es particularmente efectivo cuando se construye algo que debe llegar a otros. Esto puede ir desde una frase hablada o enviar un mensaje en internet, a elementos más complejos como una pintura, una casa o un paquete de software.

Por ejemplo, usted puede leer esta página varias veces y aun así haberla olvidado mañana; pero si tuviera que intentar explicar estas ideas a alguien usando sus propias palabras, o crear una presentación que explique estos conceptos, entonces se puede garantizar que usted tendría una mayor comprensión de estos conceptos, más integrada en sus propias ideas. Por esto la gente toma apuntes durante las lecciones, aunque nunca vayan a leerlos de nuevo.

#### c. Construccinismo social

Esto extiende las ideas anteriores a la construcción de cosas de un grupo social para otro, creando colaborativamente una pequeña cultura de elementos compartidos con significados compartidos. Cuando alguien está inmerso en una cultura como ésta, está aprendiendo continuamente acerca de cómo formar parte de esa cultura en muchos niveles.

Un ejemplo muy simple es un objeto como una copa. El objeto puede ser usado para muchas cosas distintas, pero su forma sugiere un conocimiento acerca de cómo

almacenar y transportar líquidos. Un ejemplo más complejo es un curso en línea: no sólo las formas de las herramientas de software indican ciertas cosas acerca de cómo deberían funcionar los cursos en línea, sino que las actividades y textos producidos dentro del grupo como un todo ayudarán a definir a cada persona su forma de participar en el grupo.

#### d. Conectados y Separados

Esta idea explora más profundamente las motivaciones de los individuos en una discusión. Un comportamiento separado es cuando alguien intenta permanecer objetivo, se remite a lo hechos y tiende a defender sus propias ideas usando la lógica buscando agujeros en los razonamientos de sus oponentes. El comportamiento conectado es una aproximación más empática, que intenta escuchar y hacer preguntas en un esfuerzo para entender el punto de vista del interlocutor. El comportamiento constructivo es cuando una persona es sensible a ambas aproximaciones y es capaz de escoger una entre ambas como la apropiada para cada situación particular.

En general, una dosis saludable de comportamiento conectado en una comunidad de aprendizaje es un potente estimulante para aprender, no sólo aglutinando a la gente sino también promoviendo una reflexión profunda y un replanteamiento de las propias opiniones y puntos de vista.

Por otro lado, las herramientas como BlackBoard, que cumplen la especificación SCORM 1.2 (Sharable Content Object Reference Model - SCORM)<sup>5</sup>, se refiere a Unidades de Aprendizaje y Bloques Constructivos, como sus bases para el fomento del aprendizaje constructivista, además fomenta la creación de objetos instruccionales, con un ambiente técnico compartido, que provee una serie de pautas, especificaciones y estándares, convirtiéndose en un modelo de creación y publicación de educación virtual.

---

<sup>5</sup> <http://www.adlnet.org/index.cfm?fuseaction=scormabt>

### 3.6 CARACTERÍSTICAS DE LAS PLATAFORMAS DE ADMINISTRACIÓN DE CURSOS VIRTUALES

La mayoría de las plataformas de administración de cursos virtuales, cuenta con una serie de características propias muy importantes y dependiendo de si cumplen o no las especificaciones de la AICC, utilizan bloques comunes entre ellas.

Se describen entonces, las características propias de una herramienta que no cumple con las especificaciones como Moodle y otra que sí lo hace, como WBT Manager y posteriormente otra que contiene de ambas BlackBoard.

#### 3.6.1 Características de Moodle.

Las características de Moodle corresponden a los siguientes elementos, según la página de documentación de la herramienta [24].

##### a. Diseño general

- o Promueve una pedagogía constructorista social (colaboración, actividades, reflexión crítica, etc.).
- o Apropia para el 100% de las clases en línea, así como también para complementar el aprendizaje presencial.
- o Tiene una interfaz de navegador de tecnología sencilla, ligera, eficiente, y compatible.
- o Es fácil de instalar en casi cualquier plataforma que soporte PHP. Sólo requiere que exista una base de datos (y la puede compartir).
- o Con su completa abstracción de bases de datos, soporta las principales bases de datos (excepto en la definición inicial de las tablas).
- o La lista de cursos muestra descripciones de cada uno de los cursos que hay en el servidor, incluyendo la posibilidad de acceder como invitado.

- Los cursos pueden clasificarse por categorías y también pueden ser buscados - un sitio Moodle puede albergar miles de cursos.
- Se ha puesto énfasis en una seguridad sólida en toda la plataforma. Todos los formularios son revisados, y las cookies encriptadas, por ejemplo.
- La mayoría de las áreas de introducción de texto (recursos, mensajes de los foros, entradas de los diarios y demás) pueden ser editadas usando el editor HTML, tan sencillo como cualquier editor de texto de Windows.

b. Administración del sitio

- El sitio es administrado por un usuario administrador, definido durante la instalación.
- Los temas permiten al administrador personalizar los colores del sitio, fuentes y presentación para ajustarse a sus necesidades.
- Pueden añadirse nuevos módulos de actividades a los ya instalados en Moodle.
- Los paquetes de idiomas permiten una localización completa de cualquier idioma. Estos paquetes pueden editarse usando un editor integrado. Actualmente hay paquetes para 43 idiomas.
- El código está escrito de forma clara en PHP bajo la licencia GPL, fácil de modificar.

c. Administración de usuarios

- Los objetivos son reducir al mínimo el trabajo del administrador, manteniendo una alta seguridad.
- Soporta un rango de mecanismos de autenticación a través de módulos de autenticación, que permiten una integración sencilla con los sistemas existentes.
- Método LDAP: las cuentas de acceso pueden verificarse en un servidor LDAP. El administrador puede especificar qué campos usar.
- IMAP, POP3, NNTP: las cuentas de acceso se verifican contra un servidor de correo o de noticias (news). Soporta los certificados SSL y TLS.

- Base de datos externa: Cualquier base de datos que contenga al menos dos campos puede usarse como fuente externa de autenticación.
- Cada persona necesita sólo una cuenta para todo el servidor. Por otra parte, cada cuenta puede tener diferentes tipos de acceso.
- Una cuenta de administrador controla la creación de cursos y determina los profesores, asignando usuarios a los cursos.
- Una cuenta como autor de curso permite sólo crear cursos y enseñar en ellos
- Los profesores pueden tener los privilegios de edición quitados para que no puedan modificar el curso (p.e. para tutores a tiempo parcial)
- Seguridad: los profesores pueden añadir una clave de acceso para sus cursos, con el fin de impedir el acceso de quienes no sean sus estudiantes. Pueden transmitir esta clave personalmente o a través del correo electrónico personal, etc.
- Los profesores pueden inscribir a los alumnos manualmente si lo desean.
- Los profesores pueden dar de baja a los estudiantes manualmente si lo desean, aunque también existe una forma automática de dar de baja a los estudiantes que permanezcan inactivos durante un determinado período de tiempo (establecido por el administrador).
- Se anima a los estudiantes a crear un perfil en línea de sí mismos, incluyendo fotos, descripción, etc. De ser necesario, pueden esconderse las direcciones de correo electrónico.
- Cada usuario puede especificar su propia zona horaria, y todas las fechas marcadas en Moodle se traducirán a esa zona horaria (las fechas de escritura de mensajes, de entrega de tareas, etc.).
- Cada usuario puede elegir el idioma que se usará en la interfaz de Moodle (Inglés, Francés, Alemán, Español, Portugués, etc.).

d. Administración de cursos

- Un profesor sin restricciones tiene control total sobre todas las opciones de un curso, incluido el restringir a otros profesores.

- o Se puede elegir entre varios formatos de curso tales como semanal, por temas o el formato social, basado en debates.
- o Ofrece una serie flexible de actividades para los cursos: foros, diarios, cuestionarios, recursos, consultas, encuestas, tareas, chats y talleres.
- o En la página principal del curso se pueden presentar los cambios ocurridos desde la última vez que el usuario entró en el curso, lo que ayuda a crear una sensación de comunidad.
- o La mayoría de las áreas para introducir texto (recursos, envío de mensajes a un foro, entradas en el diario, etc.) pueden editarse usando un editor HTML WYSIWYG integrado.
- o Todas las calificaciones para los foros, diarios, cuestionarios y tareas pueden verse en una única página (y descargarse como un archivo con formato de hoja de cálculo).
- o Registro y seguimiento completo de los accesos del usuario. Se dispone de informes de actividad de cada estudiante, con gráficos y detalles sobre su paso por cada módulo (último acceso, número de veces que lo ha leído) así como también de una detallada historia de la participación de cada estudiante, incluyendo mensajes enviados, entradas en el diario, etc. en una sola página.
- o Integración del correo. Pueden enviarse por correo electrónico copias de los mensajes enviados a un foro, los comentarios de los profesores, etc. en formato HTML o de texto.
- o Escalas de calificación personalizadas - los profesores pueden definir sus propias escalas para calificar foros, tareas y diarios.
- o Los cursos se pueden empaquetar en un único archivo zip utilizando la función de copia de seguridad. Éstos pueden ser restaurados en cualquier servidor Moodle.

e. Módulo de tareas

- o Puede especificarse la fecha final de entrega de una tarea y la calificación máxima que se le podrá asignar.

- Los estudiantes pueden subir sus tareas (en cualquier formato de archivo) al servidor. Se registra la fecha en que se han subido.
- Se permite enviar tareas fuera de tiempo, pero el profesor puede ver claramente el tiempo de retraso.
- Para cada tarea en particular, puede evaluarse a la clase entera (calificaciones y comentarios) en una única página con un único formulario.
- Las observaciones del profesor se adjuntan a la página de la tarea de cada estudiante y se le envía un mensaje de notificación.
- El profesor tiene la posibilidad de permitir el reenvío de una tarea tras su calificación (para volver a calificarla).

f. Módulo de chat

- Permite una interacción fluida mediante texto síncrono.
- Incluye las fotos de los perfiles in la ventana de chat.
- Soporta direcciones URL, emoticonos, integración de HTML, imágenes, etc.
- Todas las sesiones quedan registradas para verlas posteriormente, y pueden ponerse a disposición de los estudiantes.

g. Módulo de consulta

- Es como una votación. Puede usarse para votar sobre algo o para recibir una respuesta de cada estudiante (por ejemplo, para pedir su consentimiento para algo).
- El profesor puede ver una tabla que presenta de forma intuitiva la información sobre quién ha elegido qué.
- Se puede permitir que los estudiantes vean un gráfico actualizado de los resultados.

h. Módulo de foro

- Hay diferentes tipos de foros disponibles: exclusivos para los profesores, de noticias del curso y abiertos a todos.
- Todos los mensajes llevan adjunta la foto del autor.
- Las discusiones pueden verse anidadas, por rama, o presentar los mensajes más antiguos o los más nuevos primeros.
- El profesor puede obligar la suscripción de todos a un foro o permitir que cada persona elija a qué foros suscribirse de manera que se le envíe una copia de los mensajes por correo electrónico.
- El profesor puede elegir que no se permitan respuestas en un foro (por ejemplo, para crear un foro dedicado a anuncios).
- El profesor puede mover fácilmente los temas de discusión entre distintos foros.
- Las imágenes adjuntas se muestran dentro de los mensajes.
- Si se usan las calificaciones de los foros, pueden restringirse a un rango de fechas.

i. Módulo de diario

- Los diarios constituyen información privada entre el estudiante y el profesor.
- Cada entrada en el diario puede estar motivada por una pregunta abierta.
- La clase entera puede ser evaluada en una página con un único formulario, por cada entrada particular de diario.
- Los comentarios del profesor se adjuntan a la página de entrada del diario y se envía por correo la notificación.

j. Módulo de cuestionario

- Los profesores pueden definir una base de datos de preguntas que podrán ser reutilizadas en diferentes cuestionarios.
- Las preguntas pueden ser almacenadas en categorías de fácil acceso, y estas categorías pueden ser publicadas para hacerlas accesibles desde cualquier curso del sitio.

- Los cuestionarios se califican automáticamente, y pueden ser recalificados si se modifican las preguntas.
- Los cuestionarios pueden tener un límite de tiempo a partir del cual no estarán disponibles.
- El profesor puede determinar si los cuestionarios pueden ser resueltos varias veces y si se mostrarán o no las respuestas correctas y los comentarios.
- Las preguntas y las respuestas de los cuestionarios pueden ser mezcladas (aleatoriamente) para disminuir las copias entre los alumnos.
- Las preguntas pueden crearse en HTML y con imágenes.
- Las preguntas pueden importarse desde archivos de texto externos.
- Los cuestionarios pueden responderse varias veces.
- Los intentos pueden ser acumulativos, y acabados tras varias sesiones.
- Las preguntas de opción múltiple pueden definirse con una única o múltiples respuestas correctas.
- Pueden crearse preguntas de respuesta corta (palabras o frases).
- Pueden crearse preguntas tipo verdadero/falso.
- Pueden crearse preguntas de emparejamiento.
- Pueden crearse preguntas aleatorias.
- Pueden crearse preguntas numéricas (con rangos permitidos).
- Pueden crearse preguntas de respuesta incrustada (estilo cloze) con respuestas dentro de pasajes de texto.
- Pueden crearse textos descriptivos y gráficos.

k. Módulo de recurso

- Admite la presentación de cualquier contenido digital, Word, Powerpoint, Flash, vídeo, sonidos, etc.
- Los archivos pueden subirse y manejarse en el servidor, o pueden ser creados sobre la marcha usando formularios web (de texto o HTML).
- Se pueden enlazar contenidos externos en web o incluirlos perfectamente en la interfaz del curso.
- Pueden enlazarse aplicaciones web, transfiriéndoles datos.

l. Módulo de encuesta

- Se proporcionan encuestas ya preparadas (COLLES, ATTLS) y contrastadas como instrumentos para el análisis de las clases en línea.
- Los informes de las encuestas están siempre disponibles, incluyendo muchos gráficos. Los datos pueden descargarse con formato de hoja de cálculo Excel o como archivo de texto CVS.
- La interfaz de las encuestas impide la posibilidad de que sean respondidas sólo parcialmente.
- A cada estudiante se le informa sobre sus resultados comparados con la media de la clase.

m. Módulo de taller

- Permite la evaluación de documentos, y el profesor puede gestionar y calificar la evaluación.
- Admite un amplio rango de escalas de clasificación posibles.
- El profesor puede suministrar documentos de ejemplo a los estudiantes para practicar la evaluación.
- Es muy flexible y tiene muchas opciones.

### 3.6.2 Características de WBT Manager

Las características de WBT Manager, herramienta que cumple con la especificación definida por la AICC y SCORM 1.2, según sus realizadores, son las siguientes: [25]

- Ayuda a los administradores e instructores de entrenamiento web, a distribuir y monitorear fácilmente el entrenamiento basado en Web.
- Administración fácil de usar:
  - Navegación en vista de árbol.
  - Interfase Drag & Drop para el registro de estudiantes.

- Permite a los administradores locales registrar estudiantes y asignar estudiantes a los cursos.
  - Descentraliza el proceso de registro
- Controla el acceso de los estudiantes a los materiales de entrenamiento basados en el Web.
- Los estudiantes pueden hacer seguimiento de su proceso de aprendizaje.
- Soporta los cursos compatibles con los estándares CMI AICC-Internet.
- Soporta las lecciones compatibles con la especificación SCORM.
- Soporte para entrenamiento de aulas de clase (instructor-led) y exámenes.
- Interfaces personalizadas disponibles para las herramientas de autoría de contenido basado en Web no compatibles con la especificación AICC.
- Completamente soportado bajo las bases de datos Microsoft Access, Microsoft SQL Server y Oracle.
- Seguimiento por curso, departamento, estudiante y organización.
- Soporta programas con certificación.
- Permite la facturación interna y externa.
- Permite más de 30 reportes estándar predefinidos.
- Reportes personalizados, fácilmente creados con herramientas de escritura de reportes.
- Incluye una herramienta para estructurar el contenido web en cursos personalizados o introducida por software courseware (herramientas de autoría de cursos) WBT existente.
- Página principal y marco superior personalizable.
- La auto-inscripción a cursos se puede permitir para que los estudiantes la realicen automáticamente a través del listado de cursos.
- Catálogo de cursos. El módulo de estudiante cuenta con un botón Catálogo, el cual le muestra las clases de certificación y cursos disponibles en el sistema. Los cursos son organizados por categoría.
- Seguimiento de lecciones fuera de línea. Un curso o lección puede ser etiquetado como fuera de línea. El curso o lección aparece en el menú del estudiante pero no pueden ser ejecutados. Esta característica se puede usar

- para dar soporte a cursos presenciales donde los resultados son enteramente manuales.
- Modo de navegador en pantalla completa (kiosk). Una etiqueta en la definición del curso le indica al módulo del estudiante que visualice el curso en pantalla completa.
  - Auto completado de cursos o lecciones. Las lecciones o cursos pueden configurarse para que se dirijan hacia un estado Completado, una vez el estudiante inicia su lección o curso.
  - Los administradores locales pueden crear administradores. Los administradores del nivel de organización pueden adicionar/editar y eliminar administradores dentro de sus organizaciones a través de la interfase web local de administrador.
  - Soluciones escalables
  - Se integra fácilmente con las herramientas de autoría de terceros.
  - Interfase de scripts, para la construcción de soluciones e-commerce.

Es posible acceder a una demo en esta dirección (<http://www.ielearning.com/wbt/demos/II/wbtmanagerdemo1.lrm>), y se puede apreciar que incluye módulos definidos en las características de Moodle, como la administración de cursos, de estudiantes y de seguridad. Pero dejando como módulos externos las herramientas de foros, Chat, comunicación, tareas y evaluación entre otros, dependiendo éstos de las herramientas de autoría y sus capacidades. Las herramientas de autoría de cursos virtuales como authorware, dreamweaver courseware, toolbox, macromedia flash, ofrecen elementos pedagógicos para crear contenidos virtuales, incluso evaluaciones y quizzes online, pero no cuentan con herramientas tan poderosas como los módulos de Moodle, por ejemplo.

### 3.6.3 Características de BlackBoard

Una de las herramientas que mejor integra las herramientas de comunicación y demás elementos pedagógicos indispensables en la enseñanza virtual y el cumplimiento de las especificaciones AICC y SCORM, es BlackBoard, que cuenta con una gran variedad de características como:

- Administración y distribución de contenidos provee a los instructores con flexibilidad y control.
- Sistema de administración de exámenes, diseñado para mejorar la creación de evaluaciones y proveer flexibilidad en la entrega de pruebas y encuestas.
- Diseñado basándose en la realimentación de los usuarios y los tests de desempeño, el Gradebook mejora la productividad del instructor.
- Nueva funcionalidad que le permite a los instructores administrar electrónicamente la colección y organización de las tareas a través de la interfase Gradebook.
- Tableros de discusión y una nueva herramienta de salón de clase virtual, permite colaboración dinámica y comunicación en el ambiente de aprendizaje.
- Sistema de administración empresarial robusto, que permite a las instituciones manejar el crecimiento exitoso de su sistema, reduciendo sobrecostos administrativos.
- Manejo de los datos para la integración con la información de los estudiantes, administración de identidades y sistemas de autenticación.
- Las instalaciones virtuales, proveen la capacidad de crear ambientes múltiples y diferentes de Blackboard en el mismo sistema de software y hardware.
- Construido con la arquitectura de Bloques Constructivos de blackboard, lo que permite la personalización, extensión e integración de la plataforma.
- Certificación ADL Content Player Building Block, que sigue su tradición de cumplir con las especificaciones SCORM 1.2, demostrando el compromiso de Blackboard para satisfacer las necesidades del mercado de educación virtual.

También, es indispensable mostrar las características un poco más técnicas de la plataforma, como son:

#### Administración de cursos:

- Capacidades de administración de cursos enfocadas en una efectiva creación y configuración de cursos (Asistente para la creación de cursos, Plantillas de

cursos), así como herramientas para migración semestre a semestre (Copia de cursos, Reciclado de cursos) y archivado (Importación/Exportación de cursos, Archivado de curso, Backup de curso).

Autoría de contenido:

- El editor visual de texto, provee una interfase de texto enriquecido, incluyendo WYSIWYG (What You See Is What You Get) y comprobación de escritura, para crear contenido eficiente.
- La edición rápida, permita a un instructor rápidamente cambiar de la vista del estudiante a la vista de curso del instructor.
- Los instructores pueden incluir contenido de e-learning, creado en herramientas de autoría externas como Macromedia Dreamweaver, Microsoft Frontpage, o cualquier herramienta de autoría compatible con SCORM.

Entrega adaptativa:

- Provee al instructor la habilidad de crear rutas de aprendizaje personalizadas a través del contenido del curso y las actividades. Ítems de contenido, discusiones, exámenes, tareas y otras actividades, pueden ser entregadas a los estudiantes, basados en una serie de criterios incluyendo: fecha/hora, nombre de usuario, grupo, institución, rol o grado.

Creador del programa del curso:

- Otorga la habilidad a los instructores de fácilmente crear el programa de un curso, subiendo un programa existente o usando la funcionalidad de creador del programa de contenido y el plan de lecciones.

Unidades de Aprendizaje:

- Permite a los instructores crear lecciones secuenciadas y controlar el momento en que los estudiantes pueden pasar a través de ellas, o tener la posibilidad de seleccionar lecciones individuales de la tabla de contenido. Los estudiantes pueden guardar su ubicación en la unidad de aprendizaje y regresar posteriormente.

Contenido del texto guía (Cartuchos del curso):

- Todos los grandes publicadores de contenidos educativos crean contenidos pre-empacados y materiales de curso en el formato de Cartuchos de Curso Blackboard, como suplementos a sus libros del curso. El contenido del curso puede tener multimedia, exámenes, bloques de preguntas y enlaces adicionales tales como aplicaciones de aprendizaje interactivo, etc. Los cartuchos de material pueden ser personalizados una vez descargados dentro de un sitio.

Herramientas de enseñanza y aprendizaje:

Se cuenta con una gran variedad de herramientas diseñadas para el soporte específico de actividades de enseñanza / aprendizaje como:

- El Glosario, que es una herramienta para crear listas de definición personalizables.
- El Tablero Eléctrico, que es una herramienta de anotaciones en línea, que permite a los estudiantes tomar notas a medida que trabajan en su material e Información del personal docente, que contiene información detallada sobre las horas y medios de contactos con los instructores y asistentes.

Administración de la información personal:

- El calendario permite al usuario manejar y visualizar eventos específicos del curso, programados por el instructor del curso, además como los eventos institucionales y personales.

- La herramienta de Tareas, permite a los instructores asignar tareas con prioridad y fechas de entrega para los estudiantes (individuos o grupos) y seguir su progreso.
- Los mensajes de Blackboard facilitan una comunicación parecida al correo electrónico dentro de un curso sin utilizar ni direcciones ni servidores de correo externos.

#### Tablero de discusión:

- El tablero de discusión permite discusiones asincrónicas y anidadas. Los instructores pueden fijar múltiples foros alrededor de diferentes tópicos y ubicar estos foros en áreas de contenido apropiadas dentro del curso. Los instructores pueden determinar cuándo los estudiantes pueden modificar, eliminar, enviar anónimamente, incluir archivos adjuntos y otras opciones. Los foros pueden ser organizados por hilo, autor, fecha o asunto y son completamente referenciables.

#### Herramientas de colaboración. Salón de clase Virtual:

- La herramienta de colaboración, diseñada para una interacción sincrónica en vivo, soporta un ambiente de Chat basado en texto, y además un salón de clase virtual. Los instructores pueden programar las sesiones de colaboración usando cualquier ambiente. Adicionalmente al Chat basado en texto, el salón de clase virtual posee un tablero colaborativo, navegación web en grupo (web touring) y pregunta / respuesta privada. Puede correr en modo lectura o en modo de participación abierta. Los usuarios pueden levantar la mano para solicitar control y participar activamente en la sesión. Todas las sesiones de Chat, pueden ser registradas y archivadas.

#### Proyectos grupales:

- Para facilitar la colaboración entre compañeros, los instructores pueden usar las herramientas de grupos para formar múltiples grupos de estudiantes. Cada

grupo puede tener asignada su área de intercambio de archivos, tablero de discusión, salón de clase virtual y la herramienta de correo grupal para enviar correo a todos los miembros del grupo. Los estudiantes pueden pertenecer a múltiples grupos simultáneamente, así un profesor puede asignar grupos diferentes para diferentes tareas o proyectos.

#### Exámenes y encuestas:

- Los instructores pueden publicar en línea, exámenes automáticamente calificables y encuestas. Ellos pueden crear exámenes a su gusto, o utilizar bancos de preguntas personales, institucionales, o comercialmente creados. Los tipos de preguntas son 10, como falso verdadero, selección múltiple, respuesta múltiple, ordenamiento, apareamiento, completar, respuesta corta, ensayo, subir archivo y respuesta binaria. Las preguntas de los exámenes pueden ser dadas todas al tiempo o una a la vez, puede tener control de tiempo y los exámenes pueden ser tomados varias veces o una sola vez.

#### Tareas:

- Permite a los instructores crear tareas en las cuales los estudiantes pueden enviar sus respuestas. Los instructores pueden seguir las tareas de los estudiantes y descargar las formas de envío de una clase completa simultáneamente a través de la libreta de calificaciones (Gradebook). Los instructores pueden calificar las tareas y realimentar a cada estudiante que haya sido visto en línea verificando sus calificaciones.

#### Libreta de calificaciones:

- Los instructores pueden almacenar el resultado de desempeño de un estudiante en la libreta de calificaciones del curso. Los puntajes de los exámenes son automáticamente almacenados en esta libreta. La libreta de calificaciones

- soporta escalas de calificación personalizadas, peso de calificación, análisis de ítems y múltiples vistas de la libreta.
- Con permiso del instructor, los estudiantes pueden ver sus propias notas (pero no las de los demás)

#### Tablero de reportes y desempeño:

- El tablero de desempeño, provee una vista del progreso del estudiante e indica cuándo los estudiantes han revisado ítems específicos del contenido. El seguimiento de contenido provee estadísticas de uso (filtrable por usuario o rango de fechas).
- Similarmente, las estadísticas del curso proveen datos de utilización para todo el curso.
- El reporte avanzado del sistema mantiene una base de datos paralela para permitir a los administradores ejecutar reportes sin impactar el desempeño del sistema.

#### Escalabilidad empresarial:

- Basado en servidores robustos estándar de la industria, servidores de aplicación y bases de datos, el sistema Blackboard tiene una habilidad probada de escalar de cientos a miles de usuarios activos. El soporte de balance de cargas, permite una fácil configuración de servidores de aplicación adicionales para permitir un crecimiento sin problemas.

#### Soporte Multilinguaje:

- Permite a las instituciones ejecutar múltiples lenguajes en el mismo sistema. Los instructores puede fijar el lenguaje del curso independientemente del lenguaje que corre todo el sistema. Adicionalmente al soporte de la mayoría de los lenguajes Europeos, blackboard soporta los grupos de caracteres multi-byte como el Japonés y el Chino.

Estándares:

- Compatibilidad e interoperabilidad con los estándares de la industria es una capacidad fundamental de los productos de Blackboard.
- Cumple con los estándares de interoperabilidad de (IMS, SIF, OKI, etc.); especificaciones de contenido (IMS, SCORM, NLN, etc.), privacidad (FERPA), accesibilidad (Section 508), y metadata (IMS, Dublin Core, etc.).

Quizás la herramienta más completa de todas es Blackboard, y actualmente está siendo utilizada en Colombia por dos grandes instituciones estatales importantes como son el SENA y la Fiscalía General de la Nación, para sus planes de capacitación virtual y remota de sus funcionarios como es el caso de la fiscalía y de cualquier colombiano que así lo desee, como lo hace el SENA.

Finalmente, la tendencia actual es la utilización de herramientas de administración de cursos virtuales, también llamados ambientes e-learning, que no sólo cumplan con los requerimientos de las instituciones en cuanto a sus metodologías propias, sino que satisfagan la evolución y crecimiento de los estándares y normatividades internacionales al respecto.

La competencia por la utilización de las plataformas de administración de la educación virtual en Colombia, en manos de WebCT hasta el momento, obliga a buscar otras alternativas que pueden aprovechar las instituciones de educación que desean entrar en este medio de enseñanza, que sean o más económicas como Moodle o se encuentren en los lineamientos definidos por los estándares internacionales de educación virtual como BlackBoard.

## 4. LABORATORIOS INTERACTIVOS REMOTOS

---

### 4.1 LABORATORIOS REMOTOS. ANTECEDENTES

En el mundo se han presentado varias propuestas de laboratorios remotos, llamados también laboratorios virtuales o precisando aún más laboratorios interactivos remotos – LIR), que se distinguen de las aplicaciones de simulación de tareas o eventos.

Existen varias propuestas y grupos de investigación sobre el tema principalmente en Europa y Norteamérica. El grupo Lablink [26] en Europa, propone un intercambio de laboratorios virtuales en diferentes áreas como: telemetría y telecontrol de dispositivos a través de Internet, control de ruido activo, tele-laboratorio de control de maquinaria, laboratorios en automatización, que permite compartir recursos de hardware en áreas comunes de la electrónica a todos los estudiantes participantes [27].

Uno de los ejemplos en Telecontrol, se aprecia en la Figura 2, donde se permite el Telecontrol de un brazo robótico, proyecto localizado en Alemania en el VVL Automation (Verbund Virtuelles Labor - Laboratorios Virtuales Unidos en Automatización).

<http://vvl6.fh-konstanz.de/VVL/DE/Overview.html> o en la página <http://141.18.3.218/VVL/index.htm>

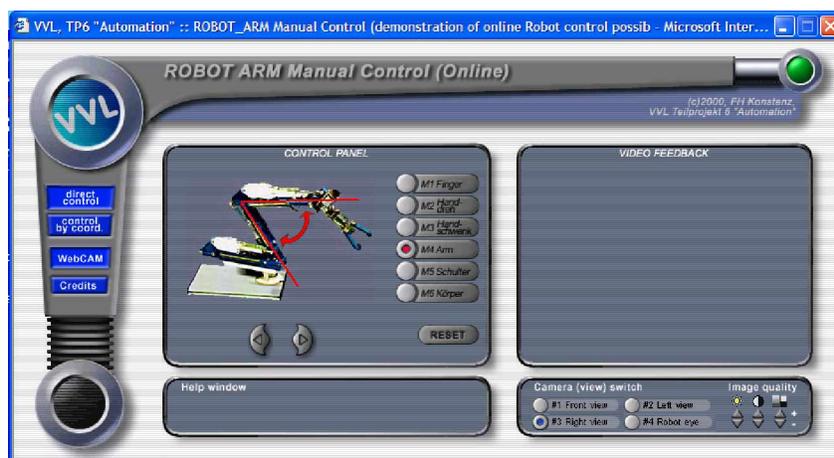


Figura 2. Telecontrol de un brazo robótico.

Por otro lado, en telemetría, se puede referenciar el proyecto del ISIB, Figura 3.



Figura 3. Osciloscopio Virtual

Ambos laboratorios, permiten interacción con el estudiante, pero este último se enfoca más en la telemetría y comunicación con un osciloscopio real que además puede ser telecontrolado para modificar sus parámetros.

Así mismo, se han logrado desarrollos locales en el tema, como por ejemplo el trabajo adelantado en la Universidad Nacional sede Manizales, llamado desarrollo de un sistema virtual para acceso remoto de instrumentación por medio de interfaz gpib, el cual aplica conceptos de instrumentación virtual y define los lineamientos para desarrollar laboratorios virtuales para dispositivos que cuenten con interfaz gpib [28]. Otro trabajo corresponde al denominado Laboratorio Interactivo Remoto Via Internet [29] orientado al área de señales y laboratorios virtuales, que brinda la posibilidad de ofrecer una interacción mayor con los estudiantes remotos mediante hardware propietario que permite adquisición de señales del lado del cliente en tiempo real y la transfiere a un servidor que se encarga de almacenarla para posterior análisis. Dichas señales son respuesta a una práctica producto del esfuerzo pedagógico entregado por parte de los profesores del área.

Por último está el trabajo Desarrollo de un sistema de monitoreo y manejo remoto sobre internet de un difractor de rayos x bruker d8 advanced" [30], que consiste en un sistema remoto de adquisición de espectros de difracción de rayos X, utilizando dos difractómetros BRUKER D8 ADVANCED. Mediante el sistema desarrollado se pueden

obtener los resultados del análisis de la composición de materiales y sustancias diversas de forma cuantitativa y cualitativa, pudiéndose realizar análisis de materiales a temperaturas de hasta 1100 °C, utilizando para ello la cámara de temperatura con la cual cuenta uno de los difractómetros.

#### 4.2 TELEMETRÍA Y TELECONTROL

Los fabricantes de dispositivos electrónicos como osciloscopios, multímetros, generadores de señales, etc., han sacado al mercado, dispositivos de instrumentación tradicionales programables o controlables, los cuales mediante lenguajes definidos por los fabricantes o estándares industriales, permiten comunicación desde plataformas de cómputo para realizar tomas de mediciones o acciones de control mediante software.

La telemetría o medición remota complementa la funcionalidad de los equipos, ampliando las capacidades del software o creando interfaces de interconexión, para permitir la realización de una medición o el control de dispositivos desde centros de cómputo localizados en otros lugares, ya sea en redes locales, redes WAN, redes públicas como Internet, la red telefónica pública, etc.

Las aplicaciones de telemetría y telecontrol son altamente utilizadas en procesos industriales donde el hombre ponga en peligro su vida, como la manipulación remota de sustancias químicas, la medición de variables físicas altamente peligrosas (vulcanología), la desactivación y detonación de artefactos explosivos a distancia, entre muchas otras.

Pero, además, a nivel residencial también se viene aplicando el concepto de telemetría y telecontrol, con el manejo de sistemas de monitoreo remoto de las residencias, sistemas de seguridad controlables remotamente, y aplicaciones de domótica.

#### 4.3 ESTÁNDARES Y PROTOCOLOS DE COMUNICACIÓN

Los estándares y protocolos de comunicación más utilizados para la telemetría y el telecontrol, son los manejados a través de las interfaces seriales, como por ejemplo RS-

232, RS-422, RS-485, el GPIB. Actualmente, la tendencia es el aumento de la velocidad de transmisión, como el bus USB, pero empaquetando estándares y lenguajes conocidos.

#### 4.3.1 RS232

El puerto serie RS-232C es la forma más comúnmente usada para realizar transmisiones de datos entre computadores [31]

El RS-232C consiste en un conector tipo DB-25 de 25 pines, aunque es normal encontrar la versión de 9 pines DB-9. En cualquier caso, los PCs no suelen emplear más de 9 pines en el conector DB-25. Dependiendo de la velocidad de transmisión empleada, es posible tener cables de hasta 15 metros [32]

#### 4.3.2 RS422 / RS485

La norma TIA/EIA-422-B más conocida como RS-422 se utiliza cuando se requieren velocidades mayores de transmisión como las que ofrece RS232, es necesario utilizar un sistema de transmisión diferencial, para evitar los efectos del ruido que aparecen con tensiones en modo común en las salidas del emisor o a la entrada del receptor [33].

La norma RS 422 fue definida por la EIA para este propósito permitiendo velocidades de transmisión de hasta 10 Mbit/s y hasta una longitud de cable de 1.200 m. Los dispositivos emisores que cumplen esta norma son capaces de transmitir señales diferenciales con un mínimo de 2 V. sobre un par de líneas trenzadas terminadas con una impedancia de  $100\Omega$ .

Los receptores deben ser capaces de detectar una señal diferencial de 200 mV. en presencia de una señal común de 7 V. La ventaja de esta norma con respecto a la RS-232 es que en aplicaciones de bus, permite que un solo emisor pueda comunicar con varios receptores aunque tiene la limitación de que los restantes receptores deben estar en estado de alta impedancia para no cargar al bus.

Esta norma permite la configuración de tres montajes básicos:

- Configuración punto a punto
- Configuración muti-carga
- Configuración muti-punto

### Configuración Punto a Punto

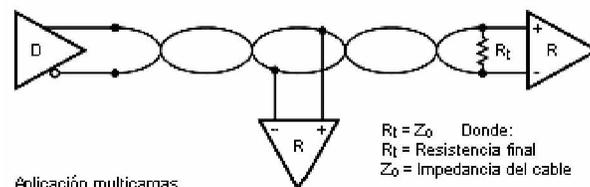
Esta configuración solo permite un controlador de dispositivo (emisor) y un solo sistema receptor. Las aplicaciones punto a punto pueden ser resueltas con normas como TIA/EIA-232-E porque esta es la configuración más popular para la RS-232.

Sin embargo, no se restringen las normas del diferencial cuando la RS-422 se utiliza como aplicación punto a punto.

### Configuración Multi-carga

La segunda configuración, Multi-carga, es un controlador de dispositivo (emisor) con dos o más receptores normalmente conectados en estrella terminando con una impedancia de carga.

Para la RS-422, el número máximo de receptores es de 10 unidades, si la impedancia de entrada del receptor ( $R_{in}$ ) es igual a 4 K es considerado 1 unidad de carga. Si la  $R_{in}$  de un receptor es igual a 8 K $\Omega$  entonces ese receptor es considerado como 1/2 unidad de carga. Por consiguiente un controlador de dispositivo de RS-422 puede manejar 10 receptores con la unidad de carga y 20 receptores con una  $R_{in}=8k\Omega$ .



Fuente: Free Serial Communication Software for RS232, RS422, RS485

Figura 4. Aplicación multicargas

## Configuración Multi-punto

El último tipo de configuración que puede realizarse con la norma RS-422 es multi-punto, que usa dos o más controladores de dispositivo (emisores) conectados a uno o más receptores (ver la Figura 5).

Normalmente no se diseñan los controladores de dispositivo RS-422 para este tipo de configuración. Sin embargo, un sistema multi-punto con RS-422 puede ser logrado si se solucionan ciertos problemas. Los tres problemas son las diferencias de potencial posibles entre los dispositivos, la competencia entre ellos, y la capacidad de control de éstos. Por tanto, para esta configuración se recomienda utilizar si es posible la norma RS-485.

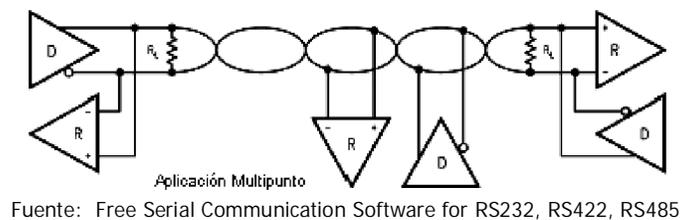


Figura 5. Aplicación Multipunto

## El estándar RS-485

La norma RS-485 se emplea fundamental para conexiones multi-punto en la industria.

La RS-485 es la única que permite una red de nodos múltiples con comunicación bidireccional con un solo par de cables trenzados, no todos los estándares combinan esta capacidad con el buen rechazo al ruido, con excelente velocidad de transmisión de datos, con gran longitud del cable de interconexión, y la robustez general del estándar.

Por estas razones, existe una gran variedad de uso de las aplicaciones con RS-485 para la transmisión de datos entre aparatos en sectores como:

- Automoción
- Informática

- Robótica
- Repetidores celulares
- Fabricantes de PLCs
- Fabricantes de Sinópticos

La RS-485 va dirigida a necesidades más amplias de las que alcanza la RS-422, ésta cubre las aplicaciones con un solo transmisor y múltiples receptores. La RS-485 es de bajo costo, bidireccional, multi-punto, interconexión con fuerte rechazo del ruido, buena tasa y rapidez de transmisión de datos, alta velocidad en la transmisión de datos y un rango del modo común ancho.

La norma especifica las características eléctricas de transmisores y receptores para la transmisión diferencial multi-punto de datos.

#### Aspectos técnicos de la Norma

Los siguientes conceptos técnicos influyen en la aplicación de la norma:

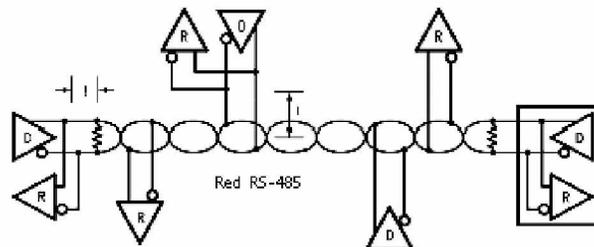
- La forma de los nodos
- Las configuraciones
- La media de interconectores
- Velocidad de los datos y la longitud del cable
- Terminales y adaptadores
- Diferencial único y parámetros de RS-485
- Blindajes y tierras
- Modo de protección
- Especial-función transmisor-receptor
- Relación fallo-seguridad

#### La forma de los Nodos

El estándar RS-485 permite su uso en redes múltiples de gran velocidad si se tiene en cuenta las siguientes recomendaciones:

1. Cada bus o red no debe tener más de 32 cargas.
2. El control direccional de los repetidores es complejo, pero se puede solucionar por hardware, por consiguiente, una estimación algo conservadora es que, sin usar los transceptores especiales, una red puede incluir 32 transceptores.

En la Figura 6 se puede ver una red RS-485 para transferir datos de forma bidireccional sustentada en un solo par de cables trenzados. Una red típica incluye nodos múltiples. Cada transceptor incluye un emisor diferencial D, y un receptor diferencial R, siendo la longitud de los cabos l.



Fuente: Free Serial Communication Software for RS232, RS422, RS485

Figura 6. Red RS-485

#### 4.3.3 GPIB

GPIB es un sistema de bus, para las aplicaciones de medición especialmente diseñado para la interconexión de computadores a instrumentos [34].

Este sistema interconectado tiene todas las características requeridas para crear un sistema de medición. El control remoto de instrumentos es un aspecto de estas características, pero hay algunas más importantes como el intercambio de datos y la capacidad de respuesta en tiempo real, sólo por nombrar algunas.

Las propiedades del bus son las siguientes:

- Se pueden conectar hasta 15 instrumentos a un computador.
- Velocidad de transmisión: 1Mbps

- Longitud del cable: 20 m entre el computador y un dispositivo o 2 m entre cada dispositivo.
- Modo de transmisión: 8 bits en paralelo.
- Intercambio: Intercambio de 3 cables, la recepción de cada byte es reconocida.
- Existen líneas adicionales para operaciones especiales, si se requieren:
  - SRQ - Service ReQuest.
  - IFC - InterFace Clear ofrece una función de reset.
  - REN - Remote ENable deshabilita la operación manual del dispositivo. El dispositivo es controlado por el controlador.
  - EOI - End Or Identify indicador de finalización en las transmisiones.
  - ATN - ATtentioN, diferencia los mensajes de control de los datos transmitidos.

En la medida en que aumenta la velocidad de las interfaces y el requerimiento de información, se observa una marcada tendencia de los fabricantes a no trabajar más con el estándar IEE488.1 pero sí a simularlo bajo interfaces más rápidas como USB 2(IEC625.1 y IEC625.2), FIREWIRE (IICP – Industrial and Instrumentation Control Protocol) y ETHERNET (VXI-11).

## 4.4 LENGUAJES DE PROGRAMACIÓN DE INSTRUMENTOS

### 4.4.1 Lenguaje estándar SCPI (Standard Commands for Programmable Instruments)

Por muchos años, los fabricantes de instrumentos han trabajado en la estandarización de las interfaces eléctricas y mecánicas entre instrumentos y computadores. Poco se había hecho, para la estandarización de mensajes enviados a través de estas interfaces. Consecuentemente, los fabricantes de instrumentos desarrollaron una amplia variedad de instrucciones para sus instrumentos, obligando a los usuarios a memorizar un nuevo vocabulario para cada instrumento. La proliferación de instrucciones, causó a los usuarios un excesivo desgaste de tiempo aprendiendo a programar los instrumentos, hizo el mantenimiento de los programas de prueba muy difícil e hizo complicado actualizar los sistemas de pruebas cuando se hacía disponible un nuevo equipo [35].

Los comandos estándar para instrumentos programables (Standard Commands for Programmable Instrumentation – SCPI) define una serie de comandos estándares para controlar los dispositivos de instrumentación programables de un sistema de instrumentación.

Un sistema de instrumentación es una colección de dispositivos de prueba y medición, conectados a través de un bus de comunicación con un computador controlador llamado el sistema controlador. Un sistema de instrumentación puede incluir dispositivos stand-alone (que funcionan solos) como los instrumentos IEEE-488, o tarjetas de instrumentación como un rack tipo VXIbus.

El sistema controlador direcciona comandos a uno o más instrumentos en el bus. Estos comandos son llamados mensajes de programa. Los instrumentos pueden enviar mensajes de respuesta nuevamente al controlador. El mensaje de respuesta, puede ser un resultado de una medición, configuración del instrumento, mensaje de error, etc. Cuando un mensaje de programa genera directamente una respuesta, se llama una consulta. En el pasado, los comandos que controlaban una función particular de un dispositivo, variaban entre instrumentos con capacidades similares. SCPI provee un

lenguaje uniforme y consistente para el control de los instrumentos de prueba y medición. Los mismos comandos y respuestas controlan las funciones correspondientes en los instrumentos compatibles con SCPI, sin importar el fabricante o el tipo de instrumento.

#### Intercambiabilidad de instrumentos

El 100% de intercambiabilidad de instrumentos SCPI sólo se puede garantizar si los instrumentos tienen exactamente las mismas funciones de medición con las mismas especificaciones, sin embargo, SCPI provee un alto grado de compatibilidad entre instrumentos, de manera que los instrumentos de diferentes fabricantes puedan ser intercambiables en muchas aplicaciones.

Uno de los objetivos fundamentales de SCPI es proveer una forma simple de ejecutar operaciones sencillas de configuración y transmisión de información entre instrumentos.

Los comandos de medición son signal oriented (orientados a la señal) en el sentido de que ellos especifican una medición en términos del resultado deseado y en términos de la señal a ser medida. Un usuario puede especificar características de la medición de la señal, tales como el valor esperado de señal o la resolución de la medición, adicionando parámetros al comando. Basado en estos parámetros, el instrumento selecciona la configuración apropiada.

El núcleo del estándar SCPI es la referencia de comandos; una lista agrupada de definiciones para todos los mensajes de programa. Estas definiciones especifican la sintaxis y semántica para cada mensaje SCPI.

## 5. MÉTODO DE ADMINISTRACIÓN Y CONSTRUCCION DE LABORATORIOS INTERACTIVOS REMOTOS

---

Este capítulo está centrado en una Metodología (Método) de Diseño de LIR, en donde se toma como punto especial la definición de un sistema de administración que sea independiente del sistema que controla el LIR como tal. El énfasis del método está en definir aquellos aspectos concretos que corresponden a una adecuada comunicación e inter-relación entre los dos sistemas (administración y operativo del LIR), de manera que ambos puedan ser desarrollados independientemente el uno del otro, pero sabiendo que se complementan para ejecutar y controlar todas las actividades del LIR.

El método que se propone tiene como base en su propuesta el desarrollo de 5 modelos:

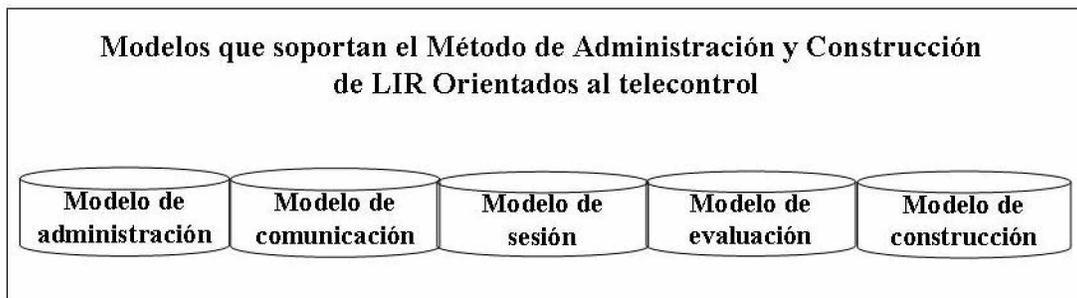


Figura 7. Modelos que soportan el Método de Administración y Construcción de un LIR orientado al telecontrol

Estos modelos surgen de distintos requerimientos y necesidades para poder coordinar y controlar todos los actores, actividades, datos, registros, laboratorios, sesiones de práctica y demás elementos. Después de una selección y agrupación se identificaron los

componentes de cada uno de los cinco modelos, los cuales tienen sus nombres según la información o tarea que han de modelar o ilustrar.

#### Modelo de Administración

En este modelo se plantea todo lo concerniente al registro de información y al acceso a la misma además de la manera como es almacenada; de los tipos de usuarios y sus tareas y de la organización de las prácticas.

#### Modelo de Comunicación

Contempla todo lo que deben involucrar tanto el módulo de administración como el operativo para que exista un adecuado entendimiento entre ambos.

#### Modelo de Sesión

Este modelo ilustra todo lo que se debe tener en cuenta del lado del módulo de administración y del módulo operativo del LIR para que se active una sesión de práctica, desde su inicio hasta su final, incluyendo inclusive los imprevistos por fallas debido a caídas del sistema o por salidas o abandonos de sesión inadecuados por parte del usuario final (estudiante/practicante).

#### Modelo de Evaluación

El modelo de evaluación contempla la manera como se registran las notas de un LIR de manera tal que queden registradas del lado del módulo de administración y no del

módulo operativo. También define la manera como los tutores acceden a las calificaciones o como deben realizar las mismas en caso tal de que ellos deban revisar lo ejecutado en las prácticas por los estudiantes y según su juicio hacer la calificación correspondiente.

## 5.1 Modelo de Administración

Tipos de Usuarios: en total son tres (3) los tipos de usuarios planteados para interactuar con los dos módulos (administración y operativo). El administrador general, los tutores y los estudiantes.

De estos tres tipos de usuarios el administrador general y los tutores interactúan solo con el módulo de administración, ya que ninguno de estos usuarios tiene la necesidad de hacer una práctica con un LIR específico, y en caso tal de que alguno de ellos necesite hacerla solo bastaría con registrarlo como un estudiante ficticio dentro de un grupo que tenga acceso al LIR correspondiente.

El administrador general tiene varias tareas, entre ellas registrar nuevos LIR y hacer las actualizaciones de los mismos. El administrador debe ser una persona cuyo sitio de trabajo debe estar localizado en donde se encuentre el módulo de administración, por ejemplo, si se habla de una institución educativa, lo ideal podría ser escoger al encargado del centro de informática (se supone además que allí estaría el servidor con el módulo de administración). En esencia esta persona debe estar pendiente de todos los datos de los diferentes LIR, que además se asume podrían estar localizados en el mismo centro de informática, aunque no es requisito que sea así. Básicamente el

administrador debe estar pendiente de un cambio de dirección IP de un LIR, o de la ruta/path en donde se encuentre instalado y obviamente de los demás datos de identificación.

El administrador también debe estar presto a registrar tutores, grupos y estudiantes, así como de asignar tutores a grupos; además debe estar pendiente de los ajustes pertinentes que se deba realizar a cualquiera de estos registros.

También entra dentro de las tareas del administrador el consultar los turnos de práctica asignados a un LIR y el estado de dichas prácticas.

Los tutores pueden registrar estudiantes a los grupos que están vinculados y pueden consultar notas de los mismos. En caso tal que un LIR genere documentación de la práctica realizada por un estudiante, el tutor puede acceder a dicha documentación para hacer la calificación correspondiente.

Los estudiantes interactúan con los dos módulos. Inicialmente deben estar registrados tanto en el sistema como en un grupo para poder acceder al sistema, ya que para el ingreso se les solicita su ID y contraseña. El ID es propio de cada estudiante y la contraseña es la registrada para el grupo. Así entonces se tiene que todos los estudiantes de un mismo grupo poseen la misma contraseña pero diferente id. Cuando un estudiante ingresa al sistema se conecta inicialmente con el módulo de administración para hacer la autenticación correspondiente, luego si desea hacer la reserva de un turno de práctica lo puede hacer en el mismo módulo de administración,

igual si desea consultar sus notas. Si el estudiante va a realizar una práctica, entonces selecciona la opción correspondiente dentro del módulo de administración y este pasa la solicitud de iniciar una práctica al módulo operativo del LIR, en adelante el estudiante interactúa con el módulo operativo.

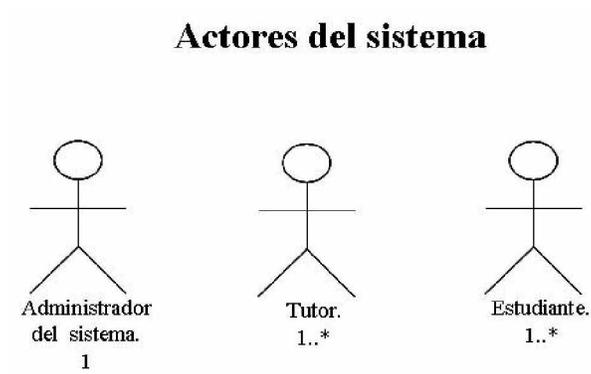


Figura 8. Actores del Sistema

Casos de Uso de los diferentes actores:

## Diagrama de Casos de Uso Administrador del Sistema

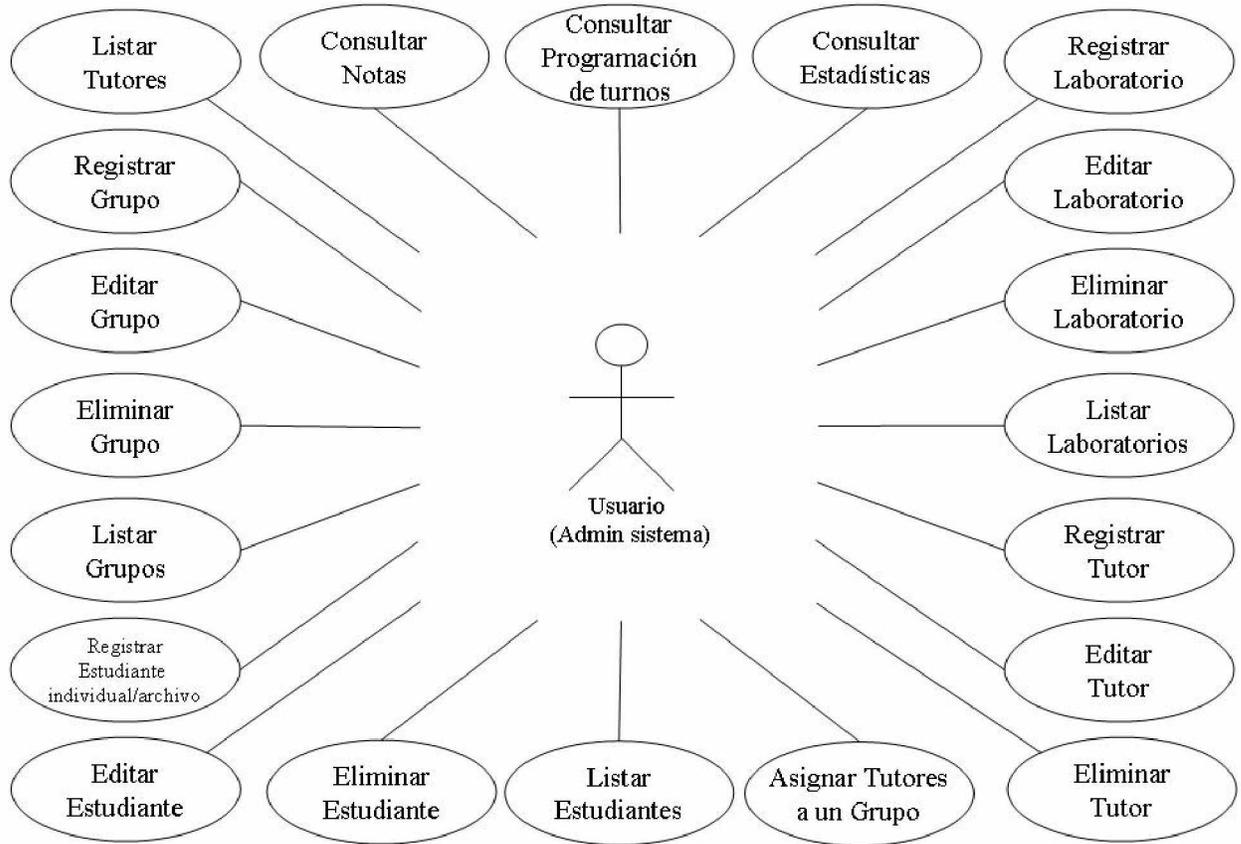


Figura 9. Diagrama de casos de uso Administrador del Sistema

## Diagrama de Casos de Uso Tutor

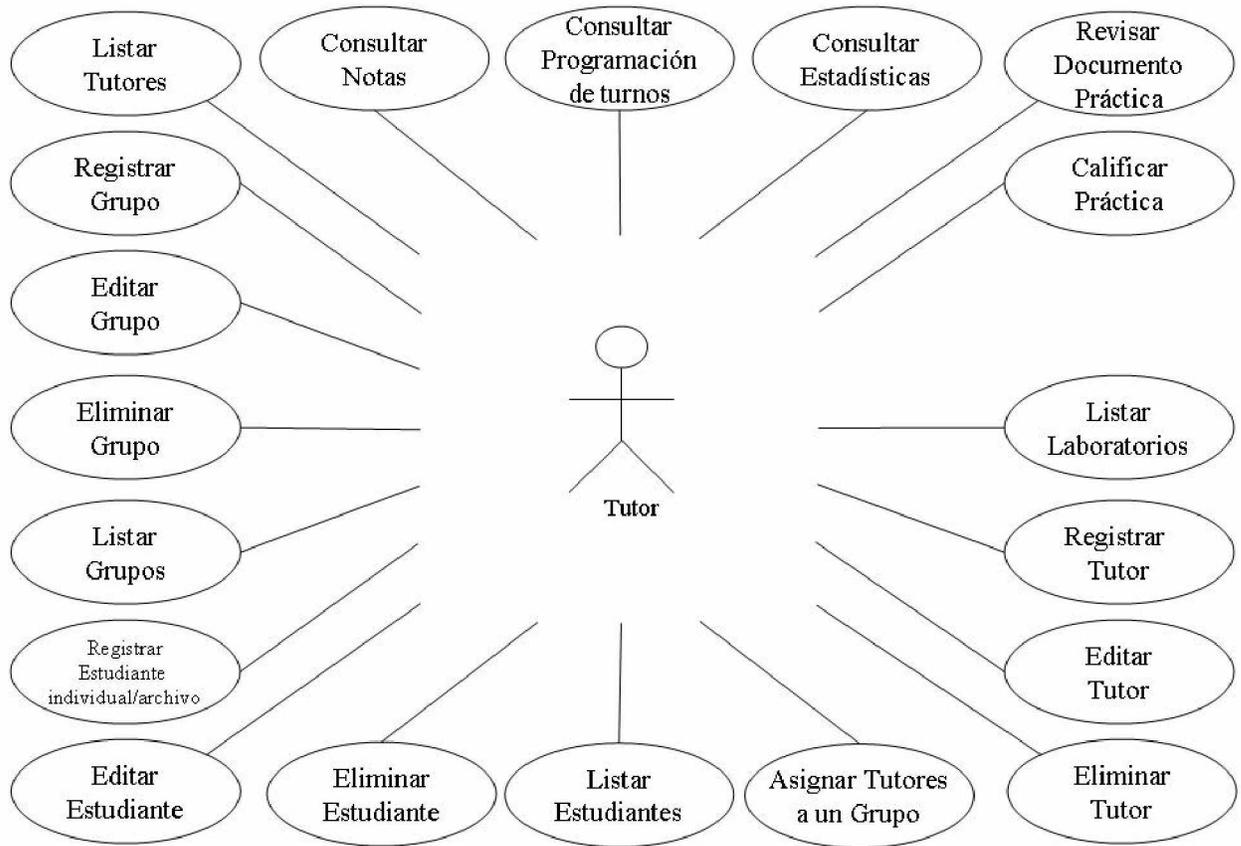


Figura 10. Diagrama de Casos de uso Tutor

## Diagrama de Casos de Uso Estudiante

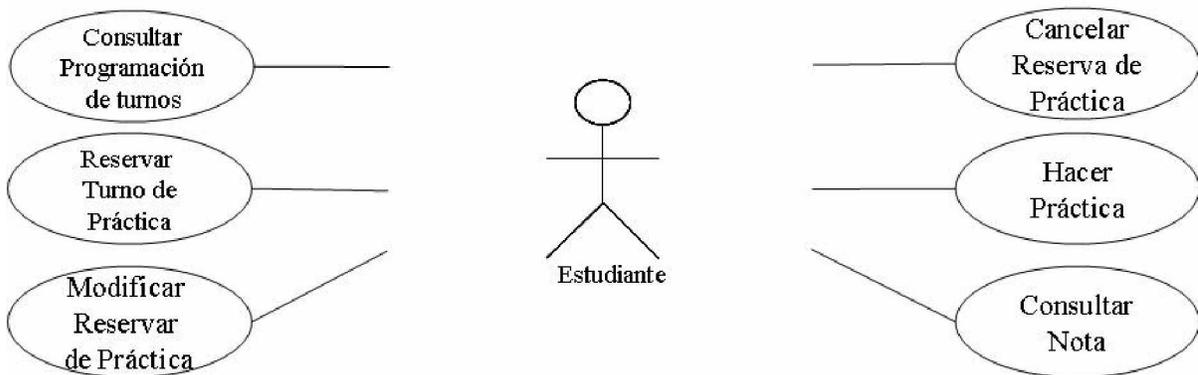


Figura 11. Diagrama de Casos de Uso Estudiante  
La descripción de cada caso de uso, con su correspondiente diagrama de interacción y contratos se puede leer en el Anexo B.

En los anteriores diagramas se relacionan los casos de uso de las actividades que debería contemplarse para cualquier sistema como el que se propone, pero cabe decir que en una implementación específica pueden aparecer otros casos de uso que tendrían que ver con actividades complementarias, entre las cuales se pueden mencionar la participación de foros de discusión de un LIR o para un grupo, el manejo de correo interno propio del sistema, publicación de mensajes, entre otros.

Registro de información y modo de almacenamiento: Los datos básicos que se plantean son los que corresponden al módulo de administración ya que cada LIR tendrá los datos que sean necesarios según los requerimientos para su adecuado funcionamiento.

Se propone que los datos sean almacenados en tablas dentro de una base de datos relacional para mayor comodidad en la elaboración de búsquedas cálculos y demás operaciones que sean necesarias al interior de la base de datos.

A continuación se presenta el diagrama E-R (entidad-relación), que sirve como diseño de la base de datos. Este es un diseño base a partir del cual se debe comenzar a incorporar otros datos (tablas y campos), dependiendo de los complementos que se vayan a adicionar en una implementación cualquiera.

Para comodidad en la lectura del diagrama, en cada recuadro se identifica solo la entidad (tabla), más no sus atributos. Para tener una mejor descripción de cada una de

las tablas se presenta más adelante el diccionario de datos que refleja toda la información pertinente a la base de datos.

## Diagrama Entidad-Relación

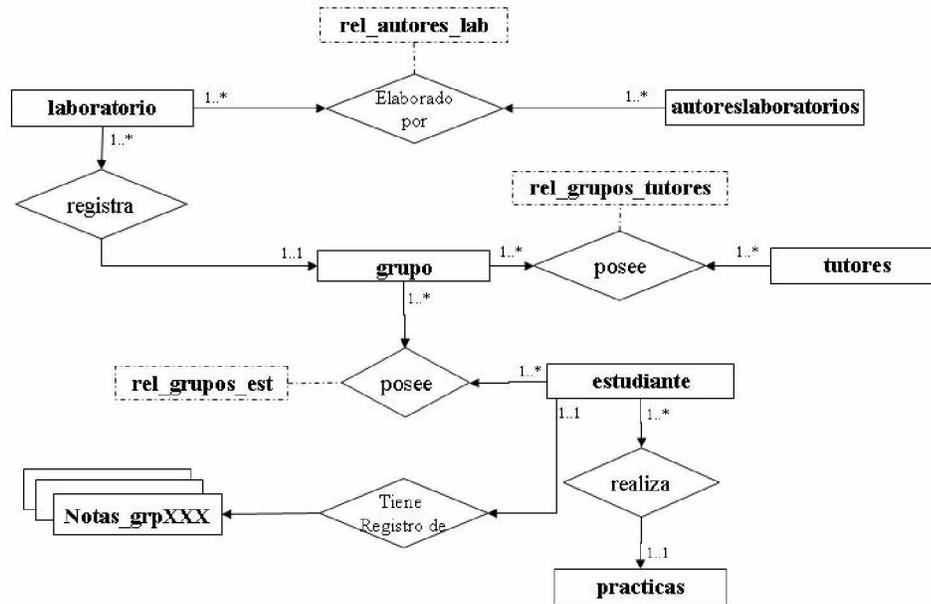


Figura 12. Diagrama Entidad Relación

En el anterior Diagrama E-R, se observa la entidad notas\_grpXXX, la cual se ha especificado de una manera diferente puesto que hace referencia a un conjunto de tablas, por eso se diseña la multiplicidad, pero además se generan dinámicamente, esto se hace en el momento de registrar un grupo y detectar a que laboratorio pertenece el grupo, ya que es el laboratorio el que define si se manejan o no notas y cuántas. Esto debe tenerse en cuenta cuando se modifique ya sea el número de notas de un laboratorio o el valor del campo auto evaluación de la tabla laboratorio.

El nombre de las tablas de la entidad notas\_grpXXX es como se especifica, solo que las XXX se deben reemplazar por el Id del grupo a que pertenecen las notas, de esta manera se asegura que no existan tablas repetidas.

#### Diccionario de datos

##### Laboratorio

Ilave	Campo	Tipo/longitud	Descripción/Validación
Primaria	id_laboratorio	Entero	Número de identificación del LIR
	ip	Char 15	Dirección IP del LIR
	puerto	Entero	Puerto de enlace del LIR, por defecto 80
	path	Char 60	Path para acceder al LIR dentro del equipo relacionado en la IP
	nombre	Char 60	Nombre del LIR
	ubicación	Char 30	Nombre de la Dependencia en donde se encuentra el LIR
	area_aplicacion	Char 30	Area de aplicación del LIR
	duracion_practica	Entero	Tiempo de la práctica dada en minutos. Si el LIR incluye la realización de varias prácticas se toma el tiempo de la más extensa.
	numero_notas	Entero	Número de notas que entrega el LIR
	documentación	Char20	Archivo que documenta el LIR. El usuario puede tener cualquier nombre de archivo en cualquier formato, pero al subir el archivo al sistema el nombre se conforma así: documento_labId, done Id es el correspondiente al LIR. Físicamente el archivo se almacena en el directorio docs_lab.
	autoevaluacion	Entero	Define si el LIR automáticamente hace la calificación de las prácticas, si genera un archivo de documentación de cada práctica para que el tutor lo evalúe o si no hay notas

##### Tutores

Ilave	Campo	Tipo/longitud	Descripción/Validación
Primaria	id_tutor	Char 9	Número de identificación del tutor dentro del sistema
	nombre	Char 35	Nombre del tutor
	contrasenia	Char 6	Contraseña para ingresar al sistema
	e-mail	Char 40	E-mail del tutor.

### Autoreslaboratorios

Ilave	Campo	Tipo/longitud	Descripción/Validación
Primaria	id_autor	Char 9	Número de identificación del autor dentro del sistema
	nombre	Char 20	Nombre(s) del autor
	apellido	Char 20	Apellidos del autor
	sitio_trabajo	Char 30	Sitio de trabajo del autor
	e_mail	Char 40	E-mail del autor
	telefono	Char 20	Teléfono del autor

### Rel\_autores\_laboratorios

Ilave	Campo	Tipo/longitud	Descripción/Validación
Foránea	laboratorio	Entero	Número de identificación del laboratorio dentro del sistema
Foránea	autor	Char 9	Número de identificación del autor dentro del sistema

### Grupos

Ilave	Campo	Tipo/longitud	Descripción/Validación
Primaria	id_grupo	Entero	Número de identificación del grupo dentro del sistema
	descripción	Char 40	Descripción del grupo
	contrasenia	Char 6	Contraseña con que ingresan los estudiantes inscritos en el grupo
	documentación	Char 20	Archivo que documenta el grupo. El usuario puede tener cualquier nombre de archivo en cualquier formato, pero al subir el archivo al sistema el nombre se conforma así: documento_grpId, done Id es el correspondiente al grupo. Físicamente el archivo se almacena en el directorio docs_gru.
Foránea	laboratorio	Entero	E-mail del autor

### Rel\_grupos\_tutores Grupos

Ilave	Campo	Tipo/longitud	Descripción/Validación
Foránea	grupo	Entero	Número de identificación del grupo dentro del sistema
Foránea	tutor	Char 9	Número de identificación del tutor dentro del sistema

### Estudiantes

Llave	Campo	Tipo/longitud	Descripción/Validación
Primaria	id_estudiante	Char 12	Identificación del estudiante dentro del sistema. (se sugiere su código)
	nombre	Char 20	Nombre del estudiante
	apellido	Char 20	Apellido del estudiante
	e_mail	Char 40	E-mail del estudiante

### Rel\_grupos\_est

Llave	Campo	Tipo/longitud	Descripción/Validación
Foránea	grupo	Entero	Número de identificación del grupo dentro del sistema
Foránea	estudiante	Char 12	Identificación del estudiante dentro del sistema.

### practicass

Llave	Campo	Tipo/longitud	Descripción/Validación
Primaria	id_practica	Entero	Consecutivo de identificación de la práctica dentro del sistema
Foránea	estudiante	Char 12	Identificación del estudiante dentro del sistema.
Foránea	laboratorio	Entero	Identificación del laboratorio en que se ha reservado la práctica.
	fecha	Fecha	Fecha para la que se ha reservado la práctica
	hora	Entero	Hora para la que se ha reservado la práctica
	minuto	Entero	Minuto para el que se ha reservado la práctica
	estado	Entero	Estado de la práctica: 1 practica sin realizar, 2 iniciada pero sin terminar, 3 práctica concluida.

### notas\_grpXXX

Llave	Campo	Tipo/longitud	Descripción/Validación
Foránea	id_estudiante	Char 12	Identificación del estudiante dentro del sistema.
	nota1	Char 12	Nota del estudiante de su práctica.

### Acceso a la información

Para poder ingresar al sistema cada usuario debe emplear su id y contraseña. El administrador del sistema es solo uno aunque en el mundo real existan varios, ya que

no hay necesidad de diferenciarlos y se dispondrá de una opción para modificar la contraseña y el ID del administrador por motivos de seguridad.

El administrador del sistema puede acceder a la información de los laboratorios, autores, grupos, tutores, estudiantes y podrá obtener las estadísticas de las prácticas, pero no puede hacer reserva de un turno de práctica.

Los tutores, que pueden ser de 1 a N en el sistema, solo pueden acceder a la información de los grupos en que están inscritos y de los correspondientes estudiantes que pertenecen a dichos grupos. Pueden ver las notas de los estudiantes y además deben hacer revisión de los documentos (archivos) de práctica de los estudiantes para registrar la nota correspondiente en el archivo de notas.

Los estudiantes solo pueden acceder a su información personal lo que incluye datos básicos de identificación y notas y también puede hacer reservación de turnos y elaborar prácticas. Un estudiante no podrá ver el horario de prácticas ni reservar turnos en un LIR que no corresponda al grupo en que esté inscrito.

#### Administración de prácticas.

Las prácticas solo pueden ser reservadas y realizadas por estudiantes, por dicho motivo estas opciones no aparecen en las interfaces del administrador y de tutores.

Cada que se hace la reserva de un turno de práctica, este queda en un estado de práctica - sin iniciar -. Cuando el estudiante ingresa al sistema para realizar la práctica,

el estado pasa a - práctica iniciada pero sin terminar - y cuando haya finalizado a un estado de - practica finalizada -. Esto permite llevar un control sobre los turnos realizados y reservados, por ejemplo, de esta manera se puede determinar para un LIR cuántos turnos hay reservados en un rango de fechas o cuántos turnos fueron reservados pero no se realizó la práctica, o cuántas prácticas completas se hicieron o cuántas prácticas incompletas se presentaron. Estos resultados se deben de obtener en varias categorías: para un LIR específico (clasificado por grupos), para un grupo específico (clasificado por estudiantes), para un estudiante y en general (sin clasificar). Los resultados estadísticos a obtener son solo estadísticos descriptivos y básicamente corresponden a un conteo (frecuencia) y el porcentaje de cada dato dentro de la categoría correspondiente.

Cuándo un estudiante ingrese a la opción de hacer reserva de turno, este solo podrá hacer reserva en el laboratorio que le corresponde a su grupo. La identificación del grupo se hace a través de una variable de sesión. Esto se logra en el momento en que el estudiante ingresa al sistema, ya que para ello debe proporcionar su ID y la contraseña del grupo, así que por medio de la contraseña se identifica el grupo.

El administrador puede consultar cualquiera de las opciones estadísticas, mientras que un tutor puede consultar las estadísticas de los grupos en que se encuentra registrado. Un estudiante no tiene acceso a estadísticas.

El módulo operativo del LIR cada que se inicia un turno de práctica debe verificar el tiempo de duración de la misma, en caso tal de que se haya cumplido el tiempo y el

estudiante no haya terminado su sesión, la práctica debe ser cancelada puesto que puede existir una reserva en el período de tiempo siguiente. En este caso el turno de práctica se debe registrar con el estado de no terminada.

## 5.2 Modelo de comunicación

El modelo de comunicación ilustra la manera como el módulo de administración y el módulo operativo de un LIR se deben de comunicar, esto con el fin de que exista un adecuado entendimiento entre ambos y que la información pertinente a una sesión de práctica sea manejada adecuadamente.

Una de las maneras en que este sistema completo se podría desarrollar es definiendo una arquitectura cliente/servidor; en donde se tendrían dos servidores, uno estaría del lado del módulo de administración y otro estaría del lado del módulo operativo de un LIR. Teniendo en cuenta esta idea, sería necesario entonces desarrollar unos demonios tanto del lado del administrador como del módulo operativo de manera que estuvieran atentos a cualquier solicitud de un cliente y hacer toda la transmisión de datos a través de Sockets, o empleando Corba o RMI. Por lo tanto para la comunicación entre ambos módulos y el cliente se tendría que definir los datagramas empleados para el envío y confirmación de recepción de datos y los protocolos de comunicación pertinentes para identificar una petición de inicio de sesión y una identificación de fin de sesión, transmisión de datos (información de usuario, notas, estado de práctica, entre otros). Según lo anterior, un cliente (usuario estudiante), para ingresar al sistema tendría que emplear una aplicación diseñada para dicho fin, ya que solo dicha aplicación estaría en capacidad de emplear los protocolos de comunicación definidos para una adecuada

transmisión de datos dentro de todo el sistema. Así entonces el usuario se vería obligado a instalar la aplicación en el computador donde va a desarrollar una práctica, lo cual puede llegar a ser un cuello de botella para acceder a las prácticas ya sea porque el cliente no posee computador propio o porque donde se encuentra no le es permitido por una u otra razón hacer la instalación requerida.

Otra forma de desarrollar el sistema propuesto es a través de la implementación de un sitio Web, que sería el módulo del administrador y que a través de este sitio se haga conexión con el sitio Web de los diferentes LIR. De esta manera un cliente (usuario estudiante), se podría comunicar con el sitio Web del administrador solamente empleando un navegador, el recomendado por el desarrollador del sitio (Se da por hecho que hoy en día cualquier computador tiene instalado un navegador).

En el presente proyecto se toma la segunda opción para el desarrollo, puesto que con esto se logra que los usuarios del sistema no tengan que hacer instalación alguna de aplicaciones que pueden tornar el proceso de prácticas en algo engorroso y dispendioso por las tareas previas no propias de la práctica.

Así entonces el modelo de comunicación debe basarse en el empleo de variables de sesión que sean capaces de almacenar la información necesaria de las diferentes sesiones de trabajo y que soporten todos los datos para el proceso de realización de prácticas y demás tareas secundarias dentro de todo el sistema.

También se debe contemplar en el desarrollo de un módulo operativo el hecho de que la dirección IP del módulo de administración debe estar registrada y almacenada en una tabla o en algún sitio específico, de manera que dicha dirección se pueda acceder para su consulta o modificación en caso de ser necesario. Esto es importante ya que en varios puntos del código fuente del módulo operativo se hace referencia dicha IP y de esta manera se logra optimizar el mantenimiento de la aplicación en caso de algún cambio. Queda a discreción de los creadores de un LIR si el acceso a dicha IP se hace por métodos no convencionales, es decir sin el empleo de la aplicación y accediendo directamente a las tablas, lo cual no es aconsejable; o si por el contrario se contempla una opción de configuración que puede ser accedida por un súper usuario del LIR, el cual tendría que ingresar al sistema como un estudiante más pero con un ID que identifique que es el súper usuario. Obviamente dicho súper usuario tendría que estar registrado en el sistema de administración para poder acceder tanto al sistema como al Laboratorio.

Teniendo en cuenta que se propone el desarrollo de un sitio Web para el módulo de administración, se hace necesario decir que en el momento de desarrollar un módulo operativo de un LIR, se debe asegurar que solo permita conexiones desde el módulo de administración, ya que es este el que le administra su agenda de prácticas y lleva el registro de las mismas.

## Diagrama de Comunicación para realizar una práctica Cliente-Módulo de Administración-Módulo Operativo LIR

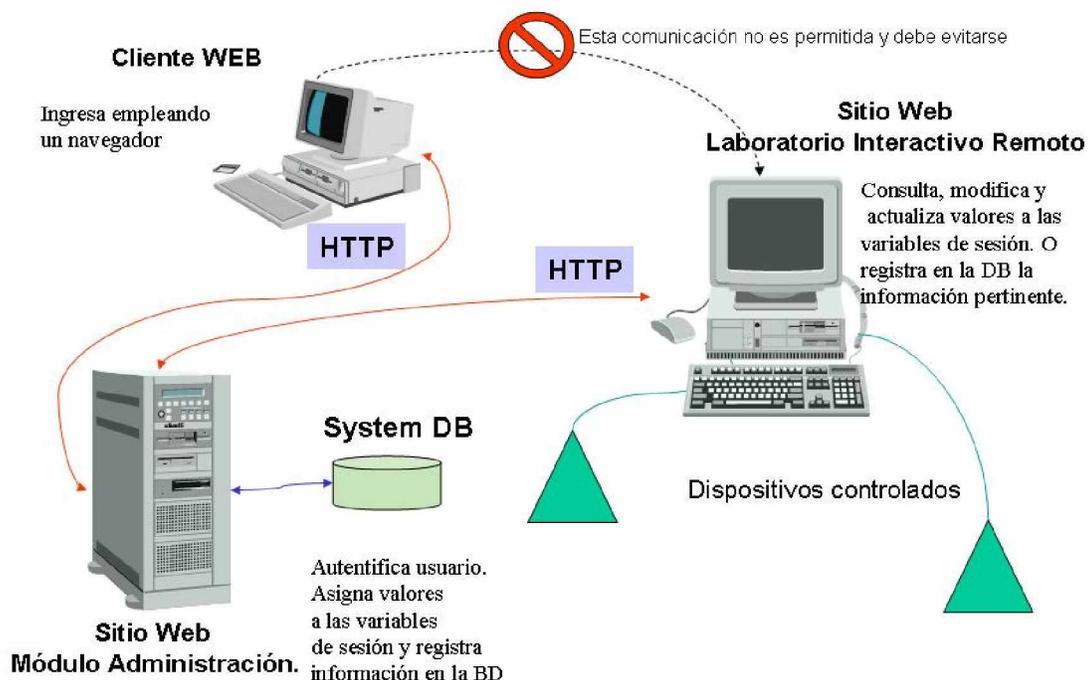


Figura 13. Diagrama de comunicación para realizar una práctica Cliente-Módulo de administración-Módulo operativo LIR.

Para los casos de uso diferentes al de realización de una práctica, la comunicación solo se da entre el cliente y el sitio Web del módulo de administración y dicha comunicación se logra a través de un navegador.

### 5.3 Modelo de sesión

El modelo de sesión incluye principalmente todas las especificaciones que los módulos de administración y el operativo de un LIR deben tener en cuenta para que se pueda realizar de manera adecuada una práctica, que es el proceso de comunicación más complejo dentro del sistema. Por tal motivo no se describen ni se aterrizan especificaciones para otras situaciones, salvo en aquellas donde sea necesario.

Cuando un usuario (estudiante), desea realizar su práctica, el primer paso es hacer una reservación de turno en el LIR correspondiente. A partir de dicha reservación el estudiante puede acceder a su práctica. Cuando el estudiante ingresa al sistema, tanto su Id como su contraseña y el id de su grupo se almacenan en variables de sesión, `sesion_id`, `sesion_contrasenia` y `sesion_id_grupo` respectivamente, esto se hace siempre sin importar la acción que se vaya a realizar. Igual ocurre si ingresa un tutor o el administrador del sistema pero estos no definen valor para la variable `sesion_id_grupo`. Estas variables de sesión quedan disponibles para ser empleadas en cualquier momento mientras no haya terminado la sesión.

Cuando el usuario en el módulo de administración selecciona hacer práctica, este módulo hace conexión con el sitio Web del LIR a través del IP, el puerto y la ruta que tiene registrados en el sistema y antes de enlazar con el LIR define una variable de sesión llamada `sesion_estado`, para identificar el estado de dicha sesión de práctica, la cual se debe de iniciar con el valor de 2 (dos), que significa práctica iniciada pero sin terminar, y también se define la variable de sesión `sesion_id_practica`, que toma el valor del id de la práctica para posteriormente poder acceder a su registro y actualizar los datos correspondientes. En la BD el valor de la variable estado, correspondiente al registro de práctica, se debe modificar igualmente a 2.

Otras variables de sesión que se inician cuando se va a iniciar una práctica son `sesion_practica_archivo`, `sesion_notas` y `sesion_tiempo`. `Sesion_practica_archivo` asume el valor de la ruta y el nombre de archivo que debe

emplear el módulo del LIR para guardar (transmitir) el archivo de documentación de práctica, si es que lo genera. `Sesion_notas` se inicia en un valor nulo o vacío y sirve para almacenar el conjunto de notas generadas por el LIR. Y `sesion_tiempo` es una variable de sesión que le indica al LIR el tiempo máximo que puede durar dicha sesión de práctica, el valor de la variable está dada en minutos y es el resultado de la diferencia entre la hora de culminación de la práctica y la hora en que se hace conexión con el LIR menos 3 (minutos). La resta de 3 minutos se hace por seguridad, para que el LIR realmente se encuentre libre entre una sesión de práctica y otra, además para que se disponga del tiempo necesario, si es del caso, para que los dispositivos automáticamente se ubiquen en posiciones iniciales para la siguiente práctica y no vayan a presentarse inconvenientes para el desarrollo de la misma. En caso tal de que un LIR necesite más de 3 minutos de disponibilidad entre una sesión de práctica y otra, este puede variar el valor de `sesion_tiempo` según lo requiera.

Lo primero que debe verificar el módulo del LIR es que se está haciendo una petición de conexión desde el módulo de administración, de lo contrario debe de rechazar la petición, luego el LIR simplemente debe seguir su flujo normal de eventos.

Cuando el estudiante concluya su práctica, este debe salirse adecuadamente del sistema para lo cual se sugiere que la interfaz del LIR sea bastante sugestiva al respecto y que indique adecuadamente como realizar dicha acción. En el momento en que se termina la sesión se debe cambiar el valor de la variable `sesion_estado` a 3 que significa práctica concluida, y a continuación el LIR se comunica y pasa el control a la página de salida del

módulo de administración para terminar la sesión de práctica y hacer el registro del estado de la misma en la BD.

Se sugiere para no dejar nombres sin establecer que la página de salida del módulo de administración se llame `fin_practica_logout.xxx`, de esta manera este nombre debe ser asumido por todos los desarrolladores de LIR y no habrá problemas de acceso a la misma puesto que los módulos operativos deben poseer almacenada la dirección URL del módulo de administración.

Si el LIR genera un archivo de documentación de la práctica realizada por el estudiante, dicho archivo debe ser grabado directamente en la dirección y con el nombre especificados en la variable de sesión `sesion_practica_archivo`.

En caso tal de que por algún motivo una sesión de práctica falle y/o el navegador del usuario se bloquee, la continuación de la práctica queda a discreción del módulo operativo. Esto se debe a que en el registro de práctica, en la BD del módulo de administración; el valor de la variable estado debe encontrarse en 2, práctica iniciada pero sin terminar, lo cual no inhabilita al estudiante a volver a establecer comunicación con el LIR siempre y cuando se encuentre dentro del rango de tiempo reservado para el turno; pero es el LIR el que según su secuencia de práctica y registro de información el que define si se puede continuar con una sesión de práctica interrumpida previamente. Por lo tanto el LIR se debe encargar de llevar un registro para especificar la continuidad de una práctica. Se debe recordar que las prácticas en un LIR solo se pueden realizar de a una a la vez, no se pueden presentar prácticas en paralelo puesto que no se debe

permitir que dos estudiantes manipulen simultáneamente un dispositivo. Según esto como el LIR puede identificar el usuario y la práctica a través de las variables de sesión `sesion_id` y `sesion_id_practica`, solo tendría que comparar el valor de dichas variables y las que posee en su registro, que identifican los valores de la práctica anterior, y en caso tal de ser iguales y no poder hacer continuidad de la práctica, entonces le notifica el hecho al usuario y hace comunicación con la página de salida del módulo de administración, `fin_practica_logout.xxx`. Se aclara que en el registro de la práctica en la BD del módulo de administración, el estado debe quedar en 2, ya que no se concluyó la práctica.

Otra utilidad de las variables `sesion_id` y `sesion_practica` es para la reubicación a sus posiciones iniciales de los dispositivos del LIR, puesto que si se ha caído la comunicación previamente y el LIR automáticamente no pudo ajustar las posiciones, con estas variables podrá identificar si hay una continuidad con la práctica anterior o si se trata de una práctica nueva y se debe tomar el tiempo necesario para ajustar los dispositivos.

## Diagrama de interacción para realizar una práctica Cliente-Módulo de Administración-Módulo Operativo LIR

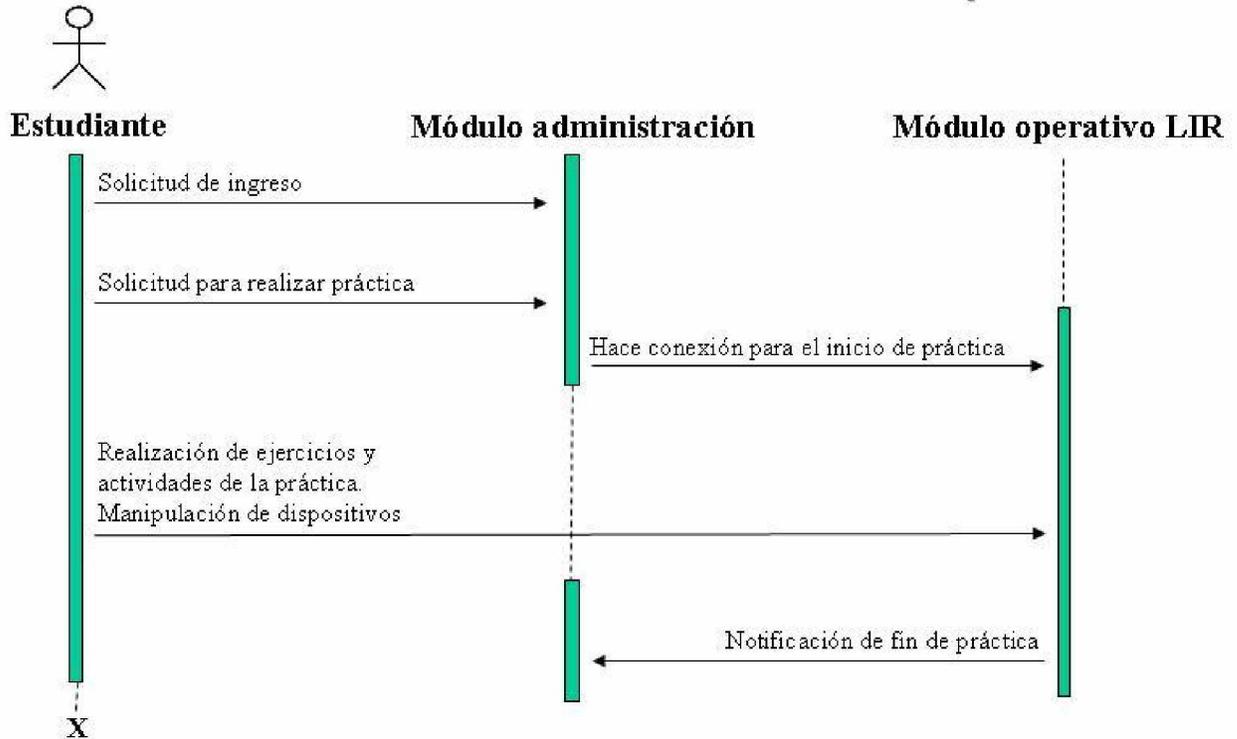


Figura 14. Diagrama de interacción para realizar una práctica Cliente-Módulo de administración-Módulo operativo LIR.

### Contratos

A continuación se presentan los contratos asociados al proceso de realización de práctica:

Nombre: Solicitud de ingreso.

Responsabilidades: verifica el nombre de usuario y contraseña para permitir el ingreso al sistema.

Excepciones: Si la verificación de id y contraseña es inválido no se da ingreso al sistema y se presenta una notificación de Error de inicio de sesión.

Precondiciones:

El estudiante debe estar registrado en el sistema y además debe estar vinculado al grupo que corresponde la contraseña.

Poscondiciones:

Se asignó a la variable de sesión `sesion_id`, el valor del id del estudiante.

`sesion_id`  $\beta$  `document.frm_login.id_estudiante.value`

Se asignó a la variable de sesión `sesion_contrasenia`, el valor de la contraseña del grupo.

`sesion_contrasenia`  $\beta$  `document.frm_login.password.value`

Nombre: Solicitud para hacer práctica.

Responsabilidades: Presentar al usuario el LIR en el cual posee reservación de turno de práctica y permitir pasar a la misma.

Precondiciones:

El estudiante debe tener reservado el turno de práctica

Poscondiciones:

Se generó un vínculo al LIR de práctica.

Nombre: Hacer conexión para el inicio de práctica.

Responsabilidades: Hacer conexión con el LIR correspondiente a la práctica e iniciar las variables de sesión con los valores indicados.

Excepciones: Si el LIR genera un archivo con información de la práctica, se hace la asignación de la dirección y nombre del archivo en la variable de sesión `sesion_practica_archivo`.

`sesion_practica_archivo` ∈ "dominio/path/nombre\_archivo"

Precondiciones:

Debe haberse generado el vínculo para ingresar al módulo operativo del LIR.

Poscondiciones:

Se asignó a la variable de sesión `sesion_estado`, el valor 2, que significa "práctica iniciada pero sin terminar".

`sesion_estado` ∈ 2

Se asignó a la variable de sesión `sesion_notas`, un valor nulo o vacío.

`sesion_notas` ∈ ""

Se asignó a la variable de sesión `sesion_practica_archivo`, la ruta y nombre del archivo donde se debe enviar el archivo de documentación.

`sesion_practica_archivo` ∈ "dominio/path/nombreArchivo"

Se asignó a la variable de sesión `sesion_tiempo`, el tiempo máximo de duración de la práctica dado en minutos.

`sesion_tiempo` ∈ `horaFinalizacion - horaIngreso - 3`

El control de la sesión es asumido por el módulo operativo del LIR.

Nombre: Realización de la práctica.

Responsabilidades: Desarrollo de todas las actividades de la práctica.

Excepciones: Si es un reingreso del estudiante con el mismo id de práctica y no es posible continuar con la práctica, no se permite continuar y se da paso a la página de fin de practica, `fin_practica_logout.xxx`, sin actualizar el valor de la variable `sesion_estado`.

Precondiciones:

La petición de conexión con el LIR debe provenir del módulo de administración.

Poscondiciones:

La variable `sesion_notas`, debe poseer todas las notas generadas por el LIR.

Se asignó a la variable `sesion_estado`, el valor 3, que significa "práctica finalizada".

### sesion\_estado B 3

Si el LIR genera un archivo de práctica, el LIR se debió transmitir a la ruta especificada en la variable `sesion_practica_archivo`.

Nombre: Notificación de fin de práctica.

Responsabilidades: Hacer conexión con la página de fin de práctica `fin_practica_logout.xxx`, del módulo de administración.

Precondiciones:

Se debe haber realizado la práctica en su totalidad.

Se debe haber validado la estructura de la variable `sesion_notas`.

Poscondiciones:

El control de la sesión fue asumido por el módulo de administración.

En el registro correspondiente a la practica se actualizó el valor de la variable estado al valor de la variable `sesion_estado` (que si es una salida normal debe valer 3).

`practica.estado = sesion_estado.`

En el registro de notas correspondiente al grupo estudiante se almacenó la o las notas respectivas del estudiante.

$\forall nota[i] \in sesion\_notas \rightarrow notas\_grpXXX.nota[i] = nota[i]$

WHERE `notas_grpXXX.id_estudiante = sesion_id`

Resumen de variables de sesión:

Nombre	Descripción
<code>sesion_id</code>	Identifica el id del usuario que ha ingresado al sistema.
<code>sesion_contrasenia</code>	Identifica la contraseña del usuario que ha ingresado al sistema.
<code>sesion_grupo</code>	En el caso de un estudiante se almacena el id del grupo al que pertenece. En el caso del administrador o de un tutor la variable queda con un valor nulo o vacío.
<code>sesion_id_practica</code>	Identifica el id de la práctica activa.
<code>sesion_estado</code>	Identifica el estado actual de la práctica activa. Puede ser 2 "práctica iniciada pero sin terminar", o 3 "práctica finalizada".
<code>sesion_practica_archivo</code>	Identifica la ruta y nombre donde se debe almacenar el archivo de documentación de práctica.
<code>sesion_notas</code>	Almacena las notas generadas por el LIR en el formato: <code>#:numNota,nota,numNota,nota,...</code> Ver modelo de evaluación.
<code>sesion_tiempo</code>	Identifica el tiempo máximo que puede durar la sesión de práctica, se obtiene con la fórmula: <code>sesion_tiempo = horaFinalización - horaIngreso - 3.</code>

#### 5.4 Modelo de evaluación

El modelo de evaluación ilustra la manera como se comunican las correspondientes notas de una práctica desde el módulo operativo de un LIR al módulo de administración, puesto que estas deben quedar almacenadas en la BD del administrador. De igual forma ilustra la manera como se envía un archivo de documentación de práctica para que sea revisado posteriormente por un tutor.

Para la notificación de notas, el módulo operativo del LIR cuenta con la variable de sesión `sesion_notas`; la cual es una variable alfanumérica cuyo contenido se debe conformar bajo un formato específico para una adecuada transmisión de notas.

El contenido de la variable `sesion_notas` se ciñe al formato:

`#:numNota,nota,numNota,nota,.....`

Donde

- `#,` identifica la cantidad de notas que está transmitiendo. Es un solo dígito pero podrían ser dos, aunque no se cree que una práctica genere más de nueve (9) notas.
- `;` es el separador entre la cantidad de notas (`#`) y el primer consecutivo `numNota`.
- `numNota,` identifica el consecutivo de la nota que va a continuación (`nota`), dentro del registro de notas. No necesariamente tiene que ser un consecutivo.
- `nota,` identifica la nota correspondiente a `numNota` a la izquierda. El contenido debe terminar en un valor de nota y no en ningún otro.

Se toma la coma (,) como el separador entre numNota y nota y viceversa.

En el contenido no debe haber espacios ni ningún otro carácter extraño.

Se asume como separador de decimales el punto.

Ejemplo de valor de sesion\_notas: si un LIR genera un total de cuatro (4) notas, y para ello se deben realizar dos (2) prácticas; en donde en la primera práctica se transmiten la nota1 y la nota2 y en la segunda se transmiten la tres y cuatro, entonces al finalizar la primera práctica sesion\_notas posee un valor como este:

2:1,4.5,2,5.0

Donde

- 2, indica que se transmiten dos notas.
- :, es el separador entre la cantidad de notas (2) y el primer consecutivo numNota (1).
- 1, indica que a continuación va el valor de nota1.
- 4.5, es el valor de nota1 dentro del registro de notas del estudiante.
- 2, indica que a continuación va el valor de nota2.
- 5.0, es el valor de nota2 dentro del registro de notas del estudiante.

Y para la segunda práctica

2:3,2.5,4,4.0

Donde

- 2, indica que se transmiten dos notas.
- :, es el separador entre la cantidad de notas (2) y el primer consecutivo numNota (3).
- 3, indica que a continuación va el valor de nota3.
- 2.5, es el valor de nota3 dentro del registro de notas del estudiante.
- 4, indica que a continuación va el valor de nota4.
- 4.0, es el valor de nota4 dentro del registro de notas del estudiante.

Un archivo de práctica posee el registro de lo que un estudiante realizó en su práctica como resultados parciales, valores de variables, listado de programas, etc. El formato del archivo es definido por los desarrolladores del LIR, ya que el módulo de administración solo se encarga de guardarlo en un directorio específico y permite que este sea descargado por el tutor para su correspondiente revisión, y es el tutor quien debe poseer el software adecuado para realizar la revisión. La nota que se asigna se ingresa en el archivo de notas correspondiente al grupo al que pertenece el estudiante.

Para que el módulo operativo pueda hacer transmisión del archivo de documentación debe empear la variable de sesión `sesion_practica_archivo`, cuyo contenido se conforma de la siguiente manera:

`ruta/nombreArchivo`

Donde

`ruta`, es la url donde se debe enviar el archivo, incluye el dominio y el path donde se almacena.

nombreArchivo, es el nombre que se le da al archivo; el cual se genera de la siguiente manera:

id\_grupo\_id\_estudiante\_id\_practica.XXX

Donde

Id\_grupo, es el id del grupo del estudiante. Se puede tomar de sesion\_grupo.

Id\_estudiante, es el id del estudiante. Se puede tomar de sesion\_id.

Id\_practica, es el id de la práctica. Se puede tomar de sesion\_id\_practica.

XXX, es la extensión del archivo.

Entre id\_grupo, id\_estudiante y id\_practica va el caracter de subrayado (\_).

## 5.5 Modelo de Construcción de LIRs

El modelo de construcción de LIR se centra en aquellos aspectos específicos que debe cumplir cualquier LIR para que pueda ser administrado por el módulo de administración planteado en el método propuesto. En el modelo se plantean las características, requerimientos, restricciones y límites entre otros aspectos. Por su característica especial para la presentación de este modelo se destina el capítulo siguiente para su descripción.

## 6. MODELO DE CONSTRUCCIÓN DE LIRS

Es de gran importancia determinar una guía de construcción de laboratorios interactivos remotos, que tenga en cuenta los requerimientos que el desarrollador debe contemplar al diseñar un LIR. El modelo de construcción propone principalmente, una guía básica con los elementos principales necesarios para poner en funcionamiento un laboratorio interactivo remoto y que pueda ser inscrito en un servidor de administración como se plantea en la metodología. Se deben seguir las etapas de Diseño y Construcción para lograr implementar el modelo. La Figura 15 muestra un diagrama de un LIR.

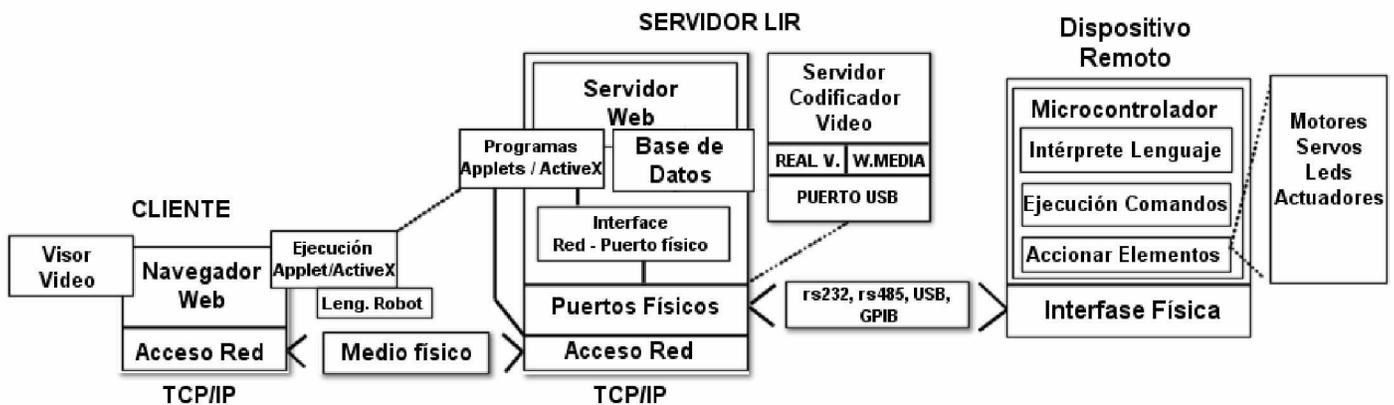


Figura 15. Diagrama de un LIR General

### 6.1 DISEÑO DE UN LABORATORIO INTERACTIVO REMOTO

Para comenzar el diseño del laboratorio se deben tener en cuenta los siguientes elementos:

#### a. Objetivos del laboratorio

El laboratorio debe tener un objetivo general claramente descrito, el cual se debe presentar al estudiante antes de realizar la práctica como tal. Los objetivos específicos se pueden presentar a medida que el laboratorio vaya sorteando

- etapas o si el diseñador lo prefiere puede plantearlos junto con el objetivo general.
- b. Descripción de tareas  
En la descripción de tareas del laboratorio, se definen las herramientas a utilizar para lograr desarrollar los objetivos y se plantean las principales funciones o tareas que debe desempeñar el laboratorio. De acuerdo a los criterios descritos en los objetivos, es muy importante adaptarse a las plataformas de software y hardware disponibles para lograr el fin del laboratorio.
  - c. Metodología de las prácticas  
La metodología de las prácticas, puede basarse en una ya existente, o plantear un desarrollo metodológico propio para lograr los objetivos de las mismas. Es bueno definir los requerimientos pedagógicos para que los estudiantes cumplan con los objetivos del área temática.
  - d. Restricciones y límites  
Es aquí donde se determinan los alcances del laboratorio, las restricciones y los límites técnicos del mismo. Es de gran importancia, plantear las posibles extensiones del proyecto para realizar una continuación del laboratorio si es necesario para futuros grupos de investigación que puedan hacerse cargo de los mismos.

## 6.2 CONSTRUCCIÓN DEL LABORATORIO

Para la construcción de un laboratorio interactivo remoto, se debe contemplar principalmente, de acuerdo a su diseño, las diferentes herramientas que permitan su implementación, es decir, cada uno de los elementos, protocolos, interfaces, dispositivos, etc., necesarios para llevar a cabo los objetivos.

Por ejemplo, si el laboratorio debe utilizar una interfase Web, es necesario utilizar alguno de los ambientes Web disponibles en el mercado, de acuerdo a las características de diseño del mismo, o si por otro lado, el objetivo es que se utilice una red ATM, para transportar los datos entre el equipo controlador y un elemento de instrumentación, se debe utilizar ese tipo de interfase.

Es así, como de acuerdo a los bloques funcionales, se ha decidido listar las herramientas más importantes en cada una de las etapas del laboratorio. Todo esto debe ser completamente compatible con el modelo de comunicación que se debe manejar entre el LIR y el módulo de administración.

#### 6.2.1 Ambientes Web (servidores).

Una de las principales características de los LIR, es su acceso a través de protocolos TCP/IP (Internet e Intranet en general). El ambiente Web que se seleccione debe contemplar entre sus características la posibilidad de manejar variables de sesión (claves en el modelo de comunicación con el sistema de administración), también que sea dinámico, es decir que permita la utilización de código embebido dentro de páginas estáticas; debe tener la capacidad de conectarse con las bases de datos más empleadas, además servir contenidos estándar como HTML y XML entre otros.

En la actualidad los ambientes más destacados que cumplen estas características son:<sup>6</sup>

- J2EE (Java, jsp, servlets)
- Coldfusion, de Macromedia
- ASP de Microsoft
- PHP (Preprocessed Hipertext Pages)

Sin hacer a un lado muchos de los proveedores que en la actualidad planean ofrecer los requerimientos mínimos anteriormente mencionados.

#### 6.2.2 Bases de Datos

Los motores de bases de datos, cada vez más poderosos, como mínimo deben ofrecer una conectividad abierta mediante ODBC. Depende ya de los programas CGI que se ejecutan en el servidor web, de enlazarse con la base de datos mediante conectores propios o externos. La elección de la base de datos depende principalmente de los recursos y licencias que se dispongan.

---

<sup>6</sup> Ver <http://www.javasoft.com>, <http://www.macromedia.com>, <http://www.microsoft.com/>, <http://www.php.net>

Las más utilizadas son:<sup>7</sup>

- MySql
- PostgreSQL
- Microsoft SqlServer
- Oracle
- Microsoft Access

De igual manera, si las restricciones del servidor web y su lenguaje CGI lo permite, puede utilizarse cualquier base de datos que cuente con conectividad ODBC.

### 6.2.3 Aplicaciones de ejecución sobre un navegador.

Este software hace referencia a los programas precompilados que se descargan en un navegador y se ejecutan utilizando las librerías locales de la máquina cliente que los ha descargado. Se utilizan para liberar de procesos pesados al servidor descargando esta responsabilidad en la máquina cliente.

Los más conocidos son:

- Applets en Java.
- Controles Active X.

### 6.2.4 Funciones de programación

Hacen referencia a aquellas funciones escritas en las páginas y que son ejecutadas por el navegador empleando intérpretes de diferentes lenguajes. Son muy utilizadas en la navegación web, por su rapidez y facilidad de uso. Con estas funciones se tiene control del modelo de objetos de un navegador (DOM), brindando acceso a cada uno de los elementos de una página web, permitiendo dar interactividad a la navegación.

Los más empleados son:

---

<sup>7</sup> Ver <http://www.mysql.com>, <http://www.postgresql.com>, <http://www.microsoft.com>, <http://www.oracle.com>,

- Javascript
- VBscript

#### 6.2.5 Software de enlace entre interfaces de hardware

Son programas encargados de comunicar dos dispositivos de hardware para permitir la transferencia de datos entre ellos, por ejemplo un software de enlace entre la interfase RS232 con un puerto Ethernet o la interfase GPIB con Ethernet.

La selección que se realice depende de las características del proyecto y cualquiera es aceptable y no acarrea consecuencias para el sistema.

Es importante resaltar que los resultados esperados en cuanto a desempeño, velocidad y rendimiento dependen exclusivamente del sistema diseñado.

## 7. MARCO EXPERIMENTAL

---

Al poner a prueba la metodología propuesta se hace obligatorio desarrollar un prototipo de módulo de administración y un prototipo de LIR, en donde se pueda constatar todo lo que se ha planteado en los diferentes modelos, para que estos dos módulos se puedan comunicar entre si y puedan coordinar adecuadamente una sesión de práctica, realizando la interpretación adecuada de datos y los ajustes de los mismos, de manera que cuando se finalice la práctica el módulo de administración pueda actualizar la BD con la información pertinente.

En el desarrollo del prototipo del módulo de administración se contempla la implementación de todas las opciones que permitan hacer registro de usuarios como tutores y estudiantes, así mismo el registro de laboratorios y grupos y principalmente el manejo de las agendas de laboratorios y el desarrollo de prácticas. No se contempla para este prototipo la implementación de todas las demás tareas de comunicación entre usuarios como foros, correos, chats, manejo de pizarras de notas entre otras actividades, puesto que no son tareas principales dentro de la metodología sino que hacen parte de todo el componente pedagógico que debe conformar una plataforma de administración robusta y nuestra intención es poner a prueba lo planteado en los modelos y la comunicación entre los módulos y no la plataforma completa ya que esto hará parte de los desarrollos a futuro.

Para el desarrollo del prototipo de módulo de administración se emplea PHP ya que este es uno de los ambientes de desarrollo propuesto dentro del modelo de construcción debido a que permite código embebido dentro de las páginas estáticas, la conexión con una BD desarrollada en MySQL y lo más importante soporta el manejo de variables de sesión.

El prototipo de LIR es una aplicación que permite a los estudiantes experimentar el control remoto de un dispositivo móvil, con el fin de practicar la programación de dicho dispositivo empleando sus propias instrucciones, ya que la aplicación exige al estudiante

definir su propio lenguaje y luego emplearlo en la programación. El estudiante podrá visualizar a través de video los desplazamientos del móvil y su objetivo final será ubicarlo en alguno de los sitios indicados por las líneas trazadas sobre la plataforma de trabajo ya sea con algún elemento en la pinza o sin él. En caso de realizarlo con un elemento en la pinza se asume una práctica de mayor complejidad, por lo que el estudiante primero debe realizar ejercicios sin tomar elementos con la pinza y luego sí con ella cuando se tenga mayor destreza en la manipulación. El laboratorio en sí sirve como un sitio de entrenamiento en la manipulación de dispositivos y como un espacio de aprendizaje para la programación de dispositivos móviles.

El prototipo de LIR a desarrollar tiene como nombre Práctica de Programación y Control de Dispositivos Robóticos. Para su desarrollo se emplea PHP y Java Applets. Para la comunicación y control de los dispositivos se utiliza el puerto RS232 del PC. Además es necesario utilizar una interfase lógica entre TCP/IP y RS232, en este caso un aplicativo de software gratuito encargado de tal propósito. Para la adquisición de video se utiliza un codificador y una cámara de video por USB.

Para la comunicación del cliente con el LIR, se define un lenguaje estilo SCPI con instrucciones predefinidas que ejecutan las órdenes necesarias para mover el dispositivo y sus diferentes partes. Para facilitar la práctica de programación es necesario desarrollar un interprete y compilador que se encargue de traducir los comandos creados por el usuario y convertirlos en las versiones del lenguaje comprensible por el dispositivo móvil.

Para poder confrontar lo realizado en los prototipos con lo expuesto en la metodología, se plantea presentar los fragmentos de código en donde las variables de sesión asumen cambios en sus valores, de acuerdo a lo presentado en los diagramas de interacción (secuencia) y contratos que describen los diferentes procesos.

Según lo anterior para las pruebas es necesario:

Para el módulo de administración:

- Servidor - PC de buenas características.
- Servidor Web con precompilador PHP instalado.
- Mysql como motor de base de datos.

Para el LIR Práctica de Programación y Control de Dispositivos Robóticos es necesario:

- Servidor - Pc de buenas características.
- Servidor Web con precompilador PHP instalado.
- Mysql como motor de base de datos
- Puertos necesarios para la comunicación con el dispositivo: RS232 – TCP/IP (Ethernet o Telefónico).

Elementos para la construcción del móvil

- Servomotores
- Motores paso a paso
- Solenoides
- Tarjeta Electrónica con los diferentes componentes electrónicos como: Microcontrolador, Relé, Transistores, resistencias, diodos, entre otros.

Para un entendimiento más amplio del LIR referirse al Anexo L, en donde se incluye el análisis y diseño correspondiente.

## 7.1 IMPLEMENTACIÓN DEL MODELO DE ADMINISTRACIÓN DE LABORATORIOS INTERACTIVOS REMOTOS (LIRS)

Tipos de Usuarios: En total existen tres tipos de usuarios el administrador, tutores y estudiantes.

Cada usuario posee su propia carta de navegación, la cual se presenta a continuación:

**Diagrama Navegacional.  
Administrador del Sistema**

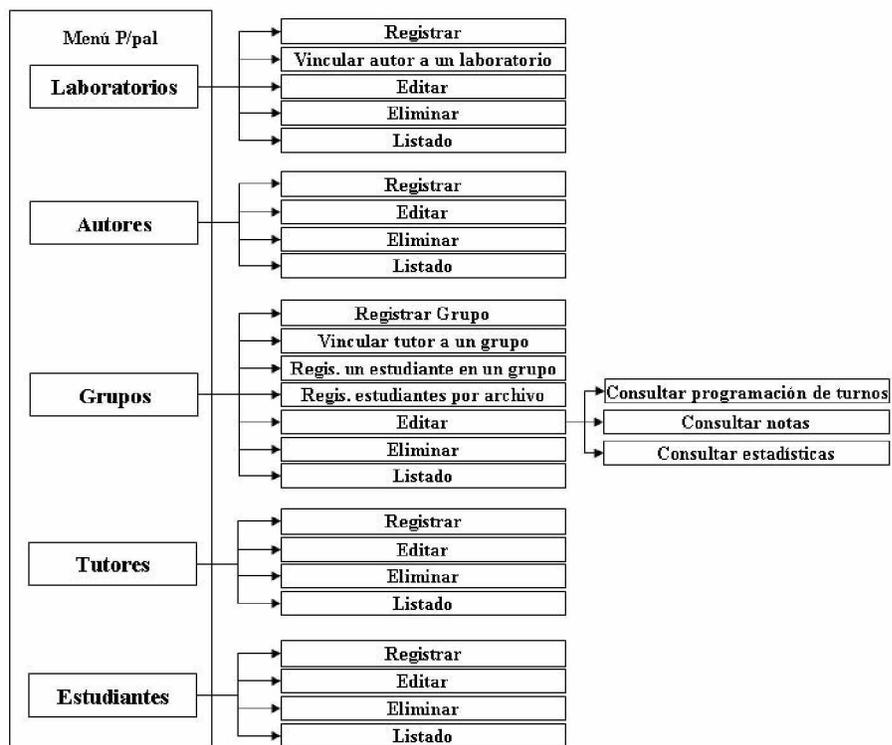


Figura 16a. Diagrama Navegacional Administrador del Sistema

**Diagrama Navegacional.  
Tutor**

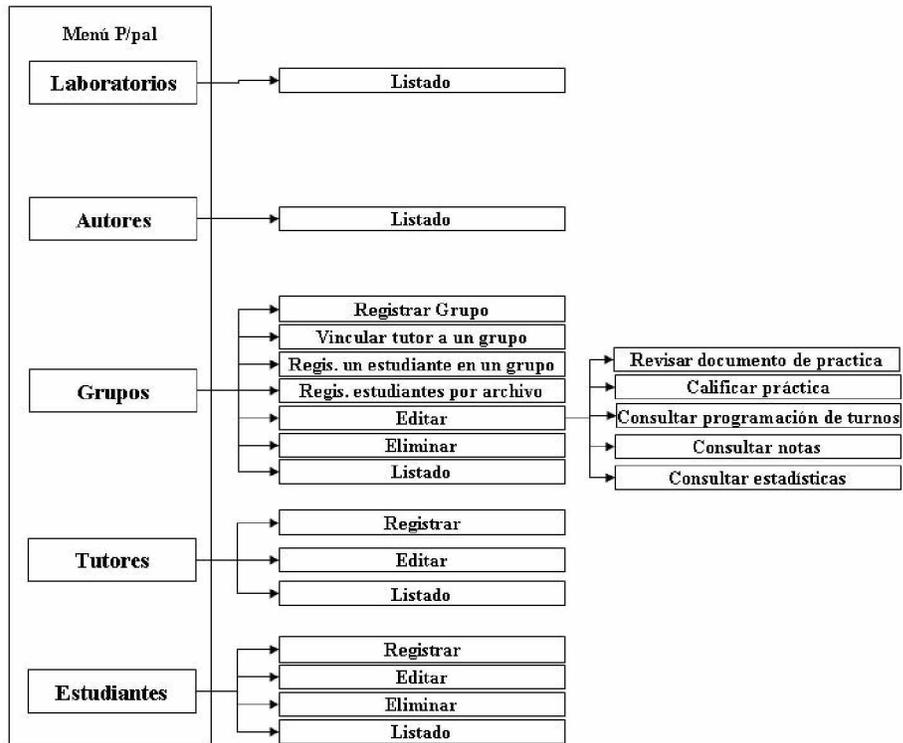


Figura 16b. Diagrama Navegacional Tutor

**Diagrama Navegacional.  
Estudiante**

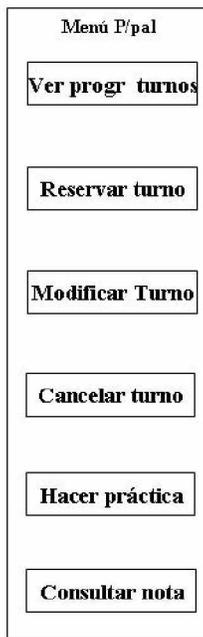


Figura 16a. Diagrama Navegacional Estudiante

Registro de información y modo de almacenamiento: para probar la metodología, en el servidor del módulo de administración se implementa una base de datos denominada SALIR, la cual además da el nombre a toda la aplicación, esta base de datos fue desarrollada con MySql. SALIR soporta toda la información del sistema y posee permisos de acceso para que los navegadores de Internet puedan acceder al registro, modificación y consulta de los datos pertinentes, pero es la aplicación en si la que debe proveer la parte de seguridad en cuanto a las diferentes opciones a las cuales tiene acceso cada usuario.

Para identificar toda su estructura de SALIR, se recurre al resultado de un Dump de MySqlFront, en donde se puede observar claramente el nombre de las tablas y de cada uno de sus campos, los cuales deben de coincidir con lo planteado en el modelo.

Dump de MySqlFront de la Base de Datos SALIR: ver Anexo C.

## 7.2 IMPLEMENTACIÓN DEL MODELO DE COMUNICACIÓN DE LABORATORIOS INTERACTIVOS REMOTOS (LIRS)

Como ya se ha mencionado, el módulo de administración se implementa en un servidor con sistema operativo LINUX, en donde además se encuentra la base de datos SALIR. También posee instalado el Apache como servidor de navegación en páginas Web. Allí también se encuentra instalada la aplicación correspondiente al módulo de administración de LIR. La aplicación se denomina SALIR y se ha desarrollado en PHP ya que cumple con las características y necesidades planteadas en el método. La página de ingreso se denomina index.php y como el resto de la aplicación se encuentra en el directorio salir; por lo tanto, para acceder a la misma desde un navegador se debe digitar la dirección del dominio seguida de /salir/index.php.

El módulo operativo del LIR desarrollado se encuentra instalado en otro equipo servidor el cual posee sistema operativo Windows XP, 256 MB de memoria RAM, procesador a 1GHz, disco duro de 40GB, tarjeta de red ethernet, puerto serial RS-232 y puerto USB 2.0, como requerimientos de hardware y como requerimientos de software, servidor apache 2.X con soporte PHP, codificador de video por difusión, interfase entre TCP/IP y RS-232 y motor de bases de datos MySQL. La aplicación del LIR se encuentra instalada en el directorio /laboratorio, a partir de la raíz web y fue desarrollada en PHP con inclusión de JAVA Applets.

El acceso al LIR solo lo puede hacer un estudiante que se encuentre registrado en el sistema SALIR y que además se encuentre inscrito en un grupo activo. Para ello el estudiante debe previamente hacer una reservación de turno y en el momento correspondiente al turno de práctica debe ingresar al módulo de administración SALIR y seleccionar la opción correspondiente a realizar práctica.

En este momento SALIR le dará paso al sitio Web del LIR y podrá realizar la práctica. Además SALIR se encarga de iniciar las variables de sesión con los valores correspondientes para que el LIR pueda hacer una adecuada sesión de práctica.

Cuando el estudiante haya finalizado su práctica y seleccione dentro del LIR la opción para terminar la práctica, el LIR hace conexión con la página de fin de sesión de práctica que se encuentra en el servidor de administración y esta página se encarga de tomar las variables de sesión y hacer los ajustes necesarios en la base de datos.

### 7.3 IMPLEMENTACIÓN DEL MODELO DE SESIÓN DE LABORATORIOS INTERACTIVOS REMOTOS (LIRS)

En esta sección se pretende mostrar como ha sido el manejo de las variables de sesión durante una práctica. Para ello se presentan un diagrama y los diferentes segmentos de código en donde se hace uso de las variables de sesión tanto en la aplicación SALIR como en la aplicación del LIR implementado.

La presentación de las variables de sesión se hará de manera secuencial según el desarrollo normal de una práctica y para hacer el seguimiento correspondiente se toma el diagrama de interacción correspondiente al caso de uso Hacer práctica y los contratos involucrados en el mismo, pero insertando las poscondiciones de cada uno de los mensajes, tal como se puede observar a continuación:

## Manejo de variables de sesión

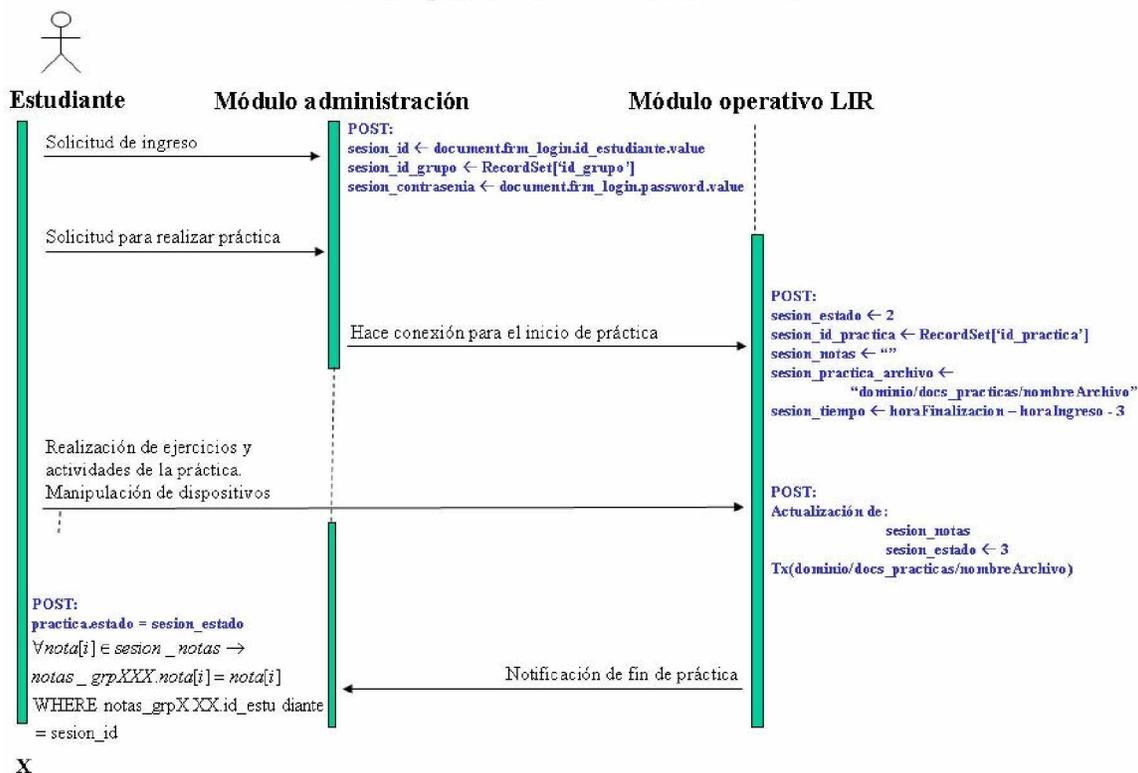


Figura 17. Manejo de Variables de Sesión

La primera acción que se realiza para desarrollar una práctica es la solicitud de ingreso al sistema SALIR por parte de un estudiante, allí se verifica que se encuentre registrado y en un grupo activo. El acceso del estudiante se procesa en un archivo denominado `sesion_inicio_est.php` el cual se presenta completo en el Anexo D (las líneas más importantes se encuentran resaltadas – negrita -).

Luego el estudiante hace la solicitud para hacer la práctica, para lo cual se recuerda que ha debido reservar turno con anticipación. En el momento que el estudiante da clic para pasar al sitio Web del LIR todas las variables de sesión deben llevar los valores correspondientes, esto se ilustra en el segmento de código del Anexo E, donde se visualiza el código del archivo `hacer_practica.php`. Las líneas más importantes se encuentran resaltadas.

Durante la práctica se deben ajustar los valores de algunas variables, como `sesion_notas` que debe poseer la cantidad, consecutivo y valor de cada una de las notas

reportadas, bajo el formato establecido en el modelo de evaluación. También la variable `sesion_estado` se debe actualizar a 3 cuando todas las actividades de la práctica hayan finalizado.

También durante la práctica, en caso tal de que el LIR no entregue notas sino un archivo donde registre lo realizado en la práctica para la posterior evaluación del tutor, se debe hacer transmisión de dicho archivo a la ruta especificada en la variable de sesión `sesion_practica_archivo`.

En el Anexo F se puede visualizar el segmento de código donde el LIR recibe las variables de sesión y en el Anexo G dos segmentos de código donde se observa la actualización de las variables `sesion_notas` y `sesion_estado`. También se puede observar otro segmento de código donde se hace la transmisión del archivo en caso tal de no haberse generado notas.

## 7.4 IMPLEMENTACIÓN DEL MODELO DE CONSTRUCCIÓN DE LABORATORIOS INTERACTIVOS REMOTOS (LIRS)

Para la implementación del modelo de construcción de LIR, se decidió elaborar un laboratorio interactivo remoto en el área del Telecontrol, que pudiera ser utilizado por los estudiantes de la Universidad Autónoma de Manizales, principalmente de las carreras de ingeniería.

### 7.4.1 Diseño del Laboratorio

Para el diseño del laboratorio, se deben seguir los elementos propuestos en el modelo de construcción de LIR.

Para ello, se debe seguir los pasos siguientes:

- Definir Objetivos del LIR
- Describir las tareas del laboratorio

- Definir la metodología de las prácticas
- Determinar las restricciones y límites

### Objetivos del LIR

#### Objetivo General

Diseñar e implementar un laboratorio interactivo remoto que permita la realización de prácticas remotas de la manipulación de un dispositivo robótico a través de un navegador Web, mediante instrucciones simples definidas por el usuario.

#### Objetivos específicos

- Diseñar e implementar un compilador de instrucciones para la manipulación del dispositivo robótico a través de un navegador Web.
- Implementar un sistema de monitoreo mediante video del dispositivo robótico a través de un navegador Web.
- Implementación del modelo de comunicación con el sistema de administración SALIR.

### Descripción de tareas del laboratorio

El LIR, tiene como finalidad permitir a los estudiantes la realización de prácticas remotas referentes a la manipulación y monitoreo de un dispositivo robótico (en este caso un vehículo) a través de un navegador Web, mediante instrucciones sencillas, que son definidas por los mismos estudiantes.

En el siguiente diagrama se describe el funcionamiento del laboratorio.

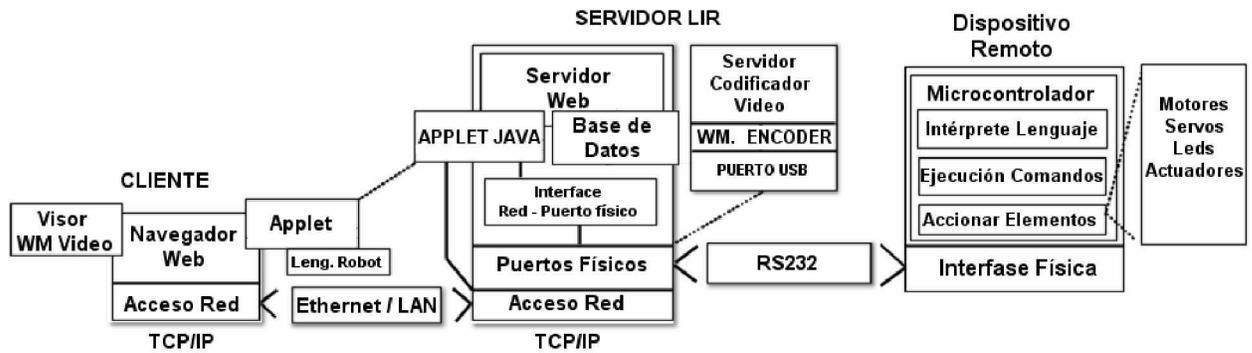


Figura 18. Diagrama del LIR de prueba implementado

Modo de operación: El laboratorio debe manejar un modo de operación con un esquema cliente / servidor, mediante protocolos TCP/IP, ya sea en un ambiente LAN o en un ambiente EXTRANET - INTERNET (teniendo en cuenta las limitaciones).

Servidor Web: Debe poseer la capacidad de servir páginas dinámicas al equipo cliente (estudiante), ubicado remotamente. Es imprescindible la utilización de un servidor Web que cumpla los requisitos establecidos por los modelos de comunicación, sesión y evaluación. En este caso se selecciona PHP.

Programas de ejecución en el cliente: El dispositivo cliente, debe descargar programas, para ser ejecutados en su máquina. Es necesario definir las políticas de seguridad de acceso locales para que se establezca la conexión entre las dos máquinas. Para el laboratorio, se utiliza Java Applets, con el java plugin como ambiente de ejecución.

Interfases físicas: De acuerdo al diagrama anterior, en este caso se establecen canales de conexión (SOCKETS) entre el cliente y el servidor para manejar la interfase (TCP/IP – RS-232) con el dispositivo robótico. En otros casos, es probable que se utilicen interfaces diferentes (p.ej TCP/IP – USB).

Bases de Datos: Se puede utilizar cualquier motor de bases de datos compatible con el servidor web seleccionado, que permita utilizar funciones predefinidas en el mismo para interactuar con los datos de la base. En este caso MySQL, que permite total compatibilidad con el servidor web y el lenguaje PHP.

Adquisición de Video: En el diagrama se utiliza un dispositivo de captura de video mediante puerto USB, el cual es compatible con los codecs de video más utilizados. En

este caso, se utiliza la codificación y el formato de Microsoft, para realizar la transmisión del mismo. Se selecciona Windows Media Encoder.

### Metodología de los laboratorios

Como los laboratorios diseñados para las asignaturas presenciales, es necesario concebir una metodología para los laboratorios remotos. Generalmente, las instituciones definen su propio planteamiento metodológico para llevar una práctica organizada y que cumpla con los requisitos del currículo exigidos por dicha asignatura.

En este caso, la metodología que puede utilizarse para el diseño de las prácticas de laboratorio, debe contar con los elementos siguientes:

- 1 Objetivos
- 2 Fundamentos teóricos
- 3 Descripción del lenguaje a definir.
- 4 Informe

Con estos elementos, el estudiante podrá realizar su práctica de acuerdo a los criterios definidos por el profesor. La evaluación de la práctica depende principalmente del profesor, quien se encarga de revisar el contenido de la práctica para verificar si se cumplieron las metas de la misma, o tener en cuenta ciertos modelos de evaluación con los que puede contar el laboratorio mismo y basado en ello otorgar una cuantificación de la práctica.

### Restricciones y límites

El presente laboratorio cuenta con las siguientes restricciones:

- Debe estar asociado al dominio en el cual se instala el módulo administrador
- Sólo permite una conexión simultánea
- La comunicación con el dispositivo remoto se debe limitar a la velocidad del puerto serial del pc.

- La transmisión de video se limita a los protocolos utilizados para difusión y los retardos previstos por los codificadores.
- La velocidad de ejecución de las instrucciones depende principalmente de los procedimientos desarrollados en el microcontrolador.

#### 7.4.2 Construcción del LIR

Para la etapa de construcción del laboratorio, y de acuerdo a las características y objetivos del mismo, se debe comenzar a abordar los diferentes bloques funcionales.

Basándonos en el objetivo general Diseñar e implementar un laboratorio interactivo remoto que permita la realización de prácticas remotas de la manipulación de un dispositivo robótico a través de un navegador Web, mediante instrucciones simples definidas por el usuario, se resalta principalmente que este laboratorio específico, debe ofrecer como mínimo una conectividad mediante el protocolo http o Web que hace parte de la serie TCP/IP.

Es aquí donde se empieza a encontrar las necesidades técnicas del laboratorio, en este caso ya se ha encontrado por lo menos que el laboratorio debe contar con un servidor web.

Las necesidades y requerimientos del servidor web, dependen principalmente de las características definidas en el modelo de comunicación, que finalmente se centran en la capacidad de utilización de las variables de sesión por parte de los navegadores, para transferir información entre páginas web.

En este caso, es posible utilizar cualquier servidor web que cuente con dicha habilidad, como son la serie de servidores Apache (PHP), IIS (PHP – ASP), Apache Jakarta Tomcat, y Macromedia ColdFusion, principalmente. Acá se ha seleccionado el servidor Apache con soporte (PHP).

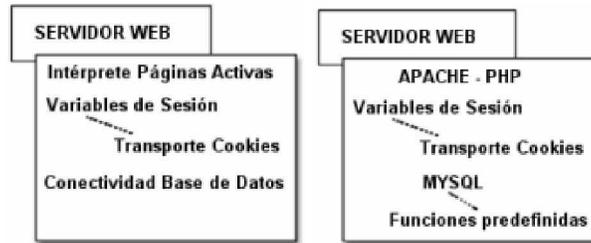


Figura 19. Servidor Web

Por otra parte, los objetivos específicos del laboratorio que se muestran a continuación,

- Diseñar e implementar un compilador de instrucciones para la manipulación del dispositivo robótico a través de un navegador Web.
- Implementar un sistema de monitoreo mediante video del dispositivo robótico a través de un navegador Web.
- Implementación del modelo de comunicación, sesión y evaluación con el sistema de administración SALIR.

Para la realización de cada una de estas acciones, se va a comentar en detalle los requerimientos.

Esto quiere decir que, en el primer caso, para poder desarrollar un compilador de instrucciones para la manipulación del dispositivo robótico a través de un navegador Web, o por lo menos ejecutado en un navegador web, es necesario utilizar una de las tecnologías que permiten realizar esta tarea, como son: Applets de Java o Controles ActiveX.

En nuestro caso, para realizar la experimentación, se implementa un applet de java con la versión 1.4.3, para ser publicado y utilizado mediante el Java Plug-In (Herramienta adicional de Sun Javasoft) que mejora el rendimiento y seguridad de los programas en lenguaje Java ejecutados en los navegadores.

Dicho Applet, debe ser descargado y ejecutado en la máquina del cliente, e inmediatamente establece una conexión TCP/IP (a través de SOCKETS) con el servidor que alberga el laboratorio.

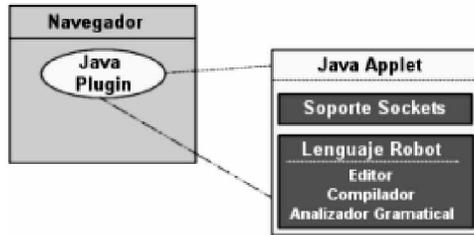


Figura 20. Esquema Applet

El compilador desarrollado en JAVA, brinda al estudiante un editor gráfico para la implementación de su programa de control, el cual permite resaltar las palabras reservadas del lenguaje de programación por él mismo definido. Además, cuenta con tres clases básicas que permiten la compilación de los programas, su respectivo análisis gramatical y su correspondiente traducción al lenguaje de máquina, comprensible por el dispositivo remoto (en este caso un lenguaje de marcas tipo SCPI). En el anexo J, se describe con mayor precisión el desarrollo de este software.

El servidor debe contar con una aplicación interfase entre TCP/IP y RS232, cuyo objetivo es establecer la conexión entre el puerto ethernet y el puerto serial de la máquina, de manera que al establecer una conexión TCP/IP, el aplicativo interconecta el socket con el puerto serial, permitiendo el envío y recepción directas hacia el puerto serial. Esto se logra con un aplicativo de software gratuito descargado de Internet de la dirección: <http://www.tcptoser.de> desarrollado por Carsten Gottbehuet.

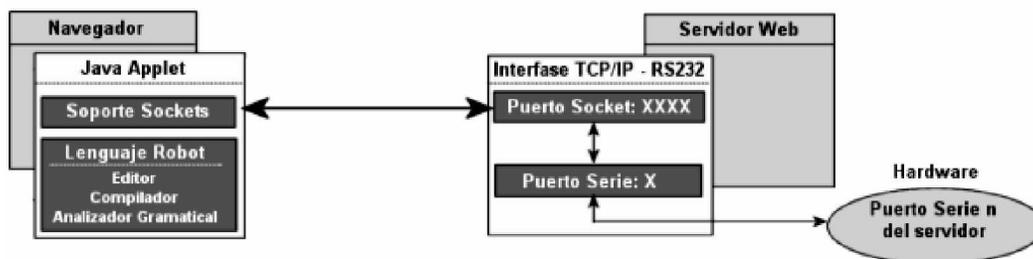


Figura 21. Esquema TCP/IP – RS232

La implementación del sistema de monitoreo, se desarrolló mediante la utilización de una herramienta de software de uso público conocida como Windows Media Encoder, que permite realizar difusiones de video y audio a través de redes TCP/IP. En este caso,

la señal se capturó de una cámara WebCam, marca KOZUMI, que maneja una resolución aceptable para la prueba.

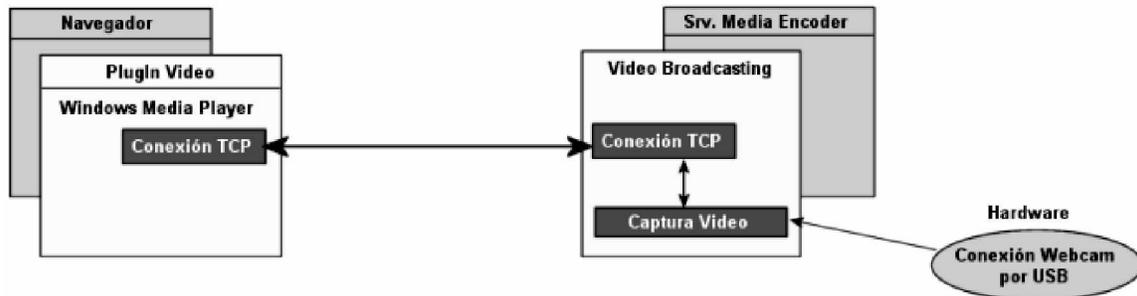


Figura 22. Esquema diagrama Video.

## 7.5 PRUEBAS DE RENDIMIENTO

Dado que la implementación de las aplicaciones para probar la metodología se hizo en la red LAN de alta velocidad de la Universidad Autónoma de Manizales, las pruebas de rendimiento realizadas podrían ser insuficientes para determinar una prueba de funcionamiento completa, cubriendo sólo una parte de los escenarios posibles de la aplicación de la metodología. Es así como se ampliaron los escenarios de pruebas para incluir las conexiones conmutadas y los accesos WAN desde otro país, en este caso España.

Las pruebas de rendimiento más relevantes, aplicadas al sistema, incluyen el análisis de las siguientes variables:

- Velocidad de transmisión de paquetes
- Tráfico y Análisis de congestión.

Todas las pruebas se realizaron con software de análisis de protocolos y software propietario de los dispositivos de la red, como switches y enrutadores, en el caso de la Universidad Autónoma de Manizales. En el Anexo H, se encuentran las tablas detalladas con los valores de las pruebas.

### 7.5.1 Velocidad de transmisión de paquetes

La velocidad de transmisión de paquetes, hablando en un contexto TCP/IP en este caso, depende de qué tipo de protocolo se está enviando o recibiendo entre las dos estaciones, es así como las pruebas de velocidad se separan de acuerdo a las estaciones involucradas. En este caso la prueba que se realiza es una transacción completa entre la estación cliente que realiza la petición hasta que es entregada la información por parte del servidor, no se realiza por paquetes individuales TCP.

Pruebas de la conexión cliente – sistema de administración SALIR

#### Prueba de protocolo HTTP

Escenario 1. Red Lan de la Universidad Autónoma de Manizales.

La primer prueba se realiza utilizando el servidor Apache 2.032 con soporte PHP, utilizando 1, 5 y 10 conexiones simultáneas y concurrentes al sistema, tomando la medición del protocolo http solamente. Los datos promedio de dichas conexiones se muestran en la siguiente tabla:

<b>Conexiones concurrentes</b>	<b>Tiempo de respuesta (msecs)</b>
1	45,9375
5	46,4375
10	46,125

Por otra parte, los datos promedio utilizando el servidor Jakarta Tomcat 5.0, una vez fueron compilados los archivos jsp, fueron los siguientes<sup>8</sup>:

<b>Conexiones concurrentes</b>	<b>Tiempo de respuesta (msecs)</b>
1	57,6875
5	58,1875
10	58,5

---

<sup>8</sup> La primera vez que se ejecutan los archivos jsp, el servidor debe compilar las clases y generar un código ejecutable. Esta compilación hace que las páginas al ser ejecutadas por primera vez se demoren hasta un 1500% más de lo habitual, dependiendo de la capacidad del equipo utilizado.

Escenario 2. Conexión remota desde España. Banda ancha 1Mbps.

Se realiza la misma prueba, a diferencia de la posibilidad de realizar conexiones concurrentes que en este caso serán de 1, 3 y 5 respectivamente. Los datos promedio de dichas conexiones se muestran en la siguiente tabla:

<b>Conexiones concurrentes</b>	<b>Tiempo de respuesta (msecs)</b>
1	884,4375
3	891,0625
5	911,1875

Los datos promedio utilizando el servidor Jakarta Tomcat 5.0, fueron los siguientes:

<b>Conexiones concurrentes</b>	<b>Tiempo de respuesta (msecs)</b>
1	908,125
3	919,1875
5	936,3125

Escenario 3. Conexión telefónica. Velocidad 36.6 Kbps

De igual manera, se realizan las mismas pruebas.

<b>Conexiones concurrentes</b>	<b>Tiempo de respuesta (msecs)</b>
1	972,3125
3	984,6875
5	993,5

Los datos promedio utilizando el servidor Jakarta Tomcat 5.0, fueron los siguientes:

<b>Conexiones concurrentes</b>	<b>Tiempo de respuesta (msecs)</b>
1	999,5625
3	1013,625
5	1018,5625

Pruebas de la conexión cliente – Laboratorio Interactivo Remoto

Prueba de protocolo HTTP

En este caso, el LIR ha sido implementado con Apache 2.0.32 con soporte PHP, y los datos registrados no se alejan mucho de las pruebas realizadas con el sistema de administración SALIR en este mismo escenario. Por las restricciones de seguridad y de administración del sistema SALIR, sólo se permite una conexión simultáneamente al laboratorio, ya que depende de los turnos reservados en el mismo.

Escenario 1. Red Lan de la Universidad Autónoma de Manizales

Conexiones concurrentes	Tiempo de respuesta (msecs)	Tiempo de respuesta Segundos
1	45,6875	0.045

Escenario 2. Conexión Remota desde España. Banda Ancha 1Mbps.

Conexiones concurrentes	Tiempo de respuesta (msecs)	Tiempo de respuesta Segundos
1	2708,5625	2.7

Escenario 3. Conexión Telefónica. 36.6 Kbps

Conexiones concurrentes	Tiempo de respuesta (msecs)	Tiempo de respuesta Segundos
1	3619,0625	3.6

Prueba de protocolo TCP (Broadcast de Video – Windows Media Video)

La prueba de Difusión de video, depende principalmente del ancho de banda disponible por la estación cliente. El codec de video permite transmitir a diferentes velocidades según la conexión del cliente. Se seleccionaron las ratas de 300Kbps, 256Kbps, 150 kbps, 128Kbps, 64Kbps y 32Kbps para realizar la difusión y dependiendo del ancho de banda la selección es automática.

Escenario 1. Red Lan de la Universidad Autónoma de Manizales

Conexiones concurrentes	Rata de Transmisión de video	Tiempo de buffering (msecs)	Tiempo de buffering Segundos
1	300 kbps	13935,90708	14
5	256 kbps	14914,73714	15
10	150 kbps	14885,54988	15

Escenario 2. Conexión Remota desde España. Banda Ancha 1Mbps.

Conexiones concurrentes	Rata de Transmisión de video	Tiempo de buffering (msecs)	Tiempo de buffering Segundos
1	300 kbps	14319,54641	14
5	128 kbps	13940,24216	14
10	32 kbps	14730,57388	15

Escenario 3. Conexión Telefónica. 36.6 Kbps

Para la conexión telefónica, sólo pudo realizarse una prueba de conexión concurrente. Las demás conexiones ralentizaban enormemente la transmisión de video a 32Kbps, lo que hace no viable utilizar más de una conexión de video por estación conectada telefónicamente.

Conexiones concurrentes	Rata de Transmisión de video	Tiempo de buffering (msecs)	Tiempo de buffering Segundos
1	32 kbps	14930,77081	15

#### Prueba de protocolo TCP (Socket)

Las pruebas de comunicación entre el cliente y el socket del laboratorio interactivo no mostraron ninguna variación apreciable en la conexión desde los sistemas locales y remotos. El envío de las instrucciones hacia el robot se hace a una velocidad muy rápida en cualquiera de los escenarios.

Escenario	Tiempo de respuesta SOCKET (msecs)
1 (LAN UNIVERSIDAD)	1,62532
2 (CONEXION ESPAÑA)	2,3224
3 (CONEXION MODEM)	3,5768

#### 7.5.2 Pruebas de Tráfico y Congestión de la Red

De acuerdo a los escenarios anteriormente determinados, las pruebas de tráfico y congestión de la red dependen de las características propias del enlace establecido cliente – servidor. Se decidió realizar las pruebas de análisis de tráfico y congestión, ejecutando la aplicación en conjunto, en cuyo caso se utilizan las herramientas de difusión de video, conectividad a través de sockets y transferencia de hipertexto.

Se utilizaron herramientas de análisis de tráfico en los adaptadores implicados en la conexión, principalmente del lado del cliente.

### Escenario 1. Red LAN de la Universidad Autónoma de Manizales

#### Conexión Cliente – Laboratorio Interactivo Remoto

Se realizaron pruebas con 1, 2 conexiones simultáneas desde el mismo cliente, para determinar el ancho de banda disponible para dicho enlace.

Con una conexión, se obtiene los datos graficados en la figura 23:

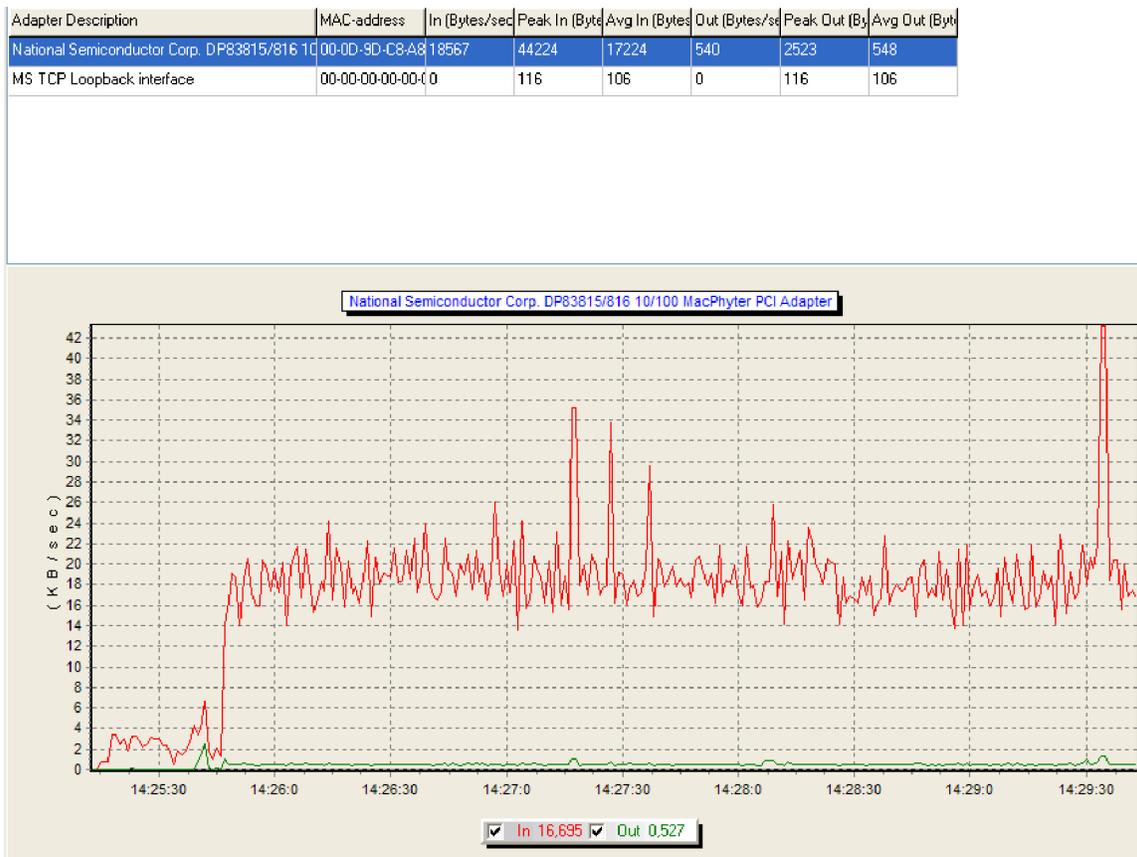


Figura 23. Tráfico: Cliente – LIR. Escenario 1 (una conexión)

La tasa de transmisión con un cliente, es de 256 kbps (indicación del codificador de video). Los datos en la gráfica están en KBps, así que debemos llevarlos a Kbps, para mostrar una sola referencia.

Rata Tx Cliente video	Paq. Promedio In / seg	Pico In / seg	Paq. Promedio Out / seg	Pico Out /seg
291 kbps	137.792 Kbps	353.792 Kbps	4.384 Kbps	20.184 Kbps

Con dos conexiones simultáneas, los datos de la figura 24:

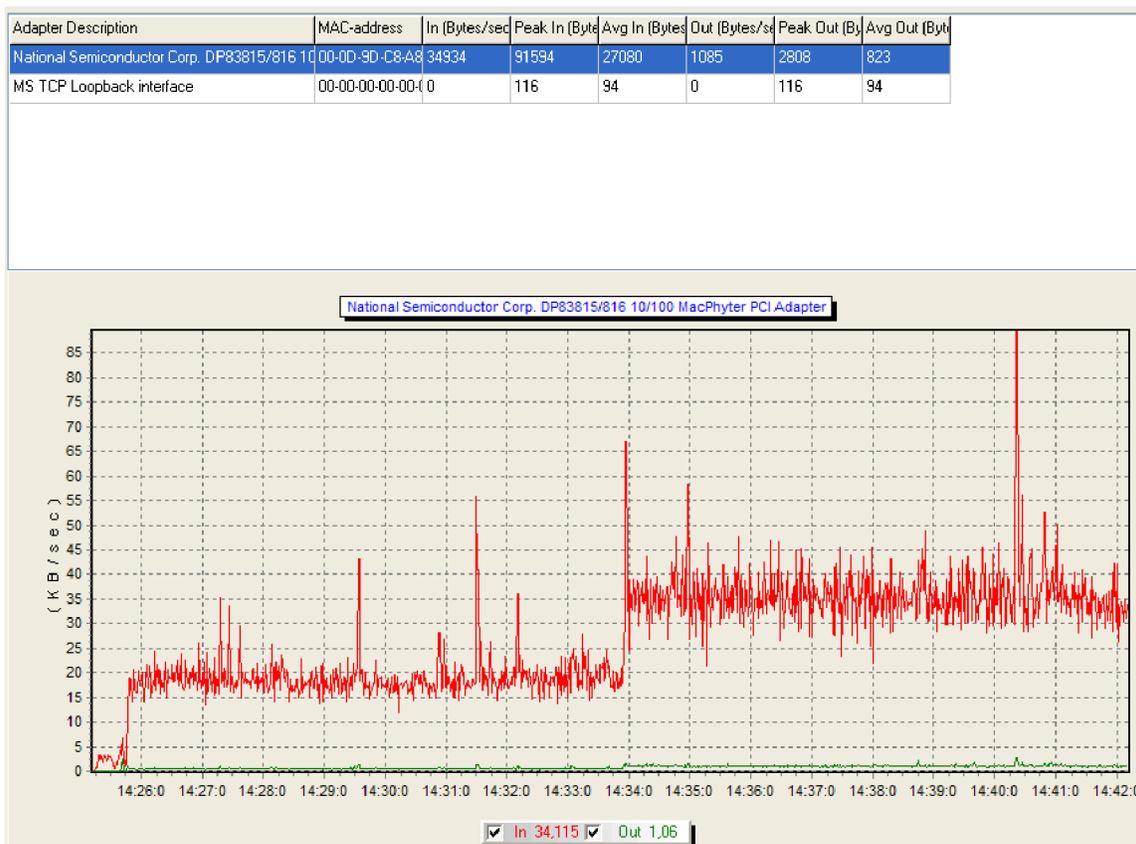


Figura 24. Tráfico: Cliente – LIR. Escenario 1 (dos conexiones)

La rata de transmisión para cada uno es de 121 kbps (según el reproductor de video de cada visualización).

Rata Tx Cliente video	Paq. Promedio In / seg	Pico In / seg	Paq. Promedio Out / seg	Pico Out /seg
121 kbps	216.064 Kbps	732.752 Kbps	6.584 Kbps	22.4 Kbps

Escenario 2. Conexión desde España. Banda Ancha 1Mbps

#### Conexión Cliente – Laboratorio Interactivo Remoto

Se realizaron las mismas pruebas con una y dos conexiones simultáneas respectivamente, para determinar el ancho de banda disponible para dicho enlace.

Con una conexión, Figura 25:

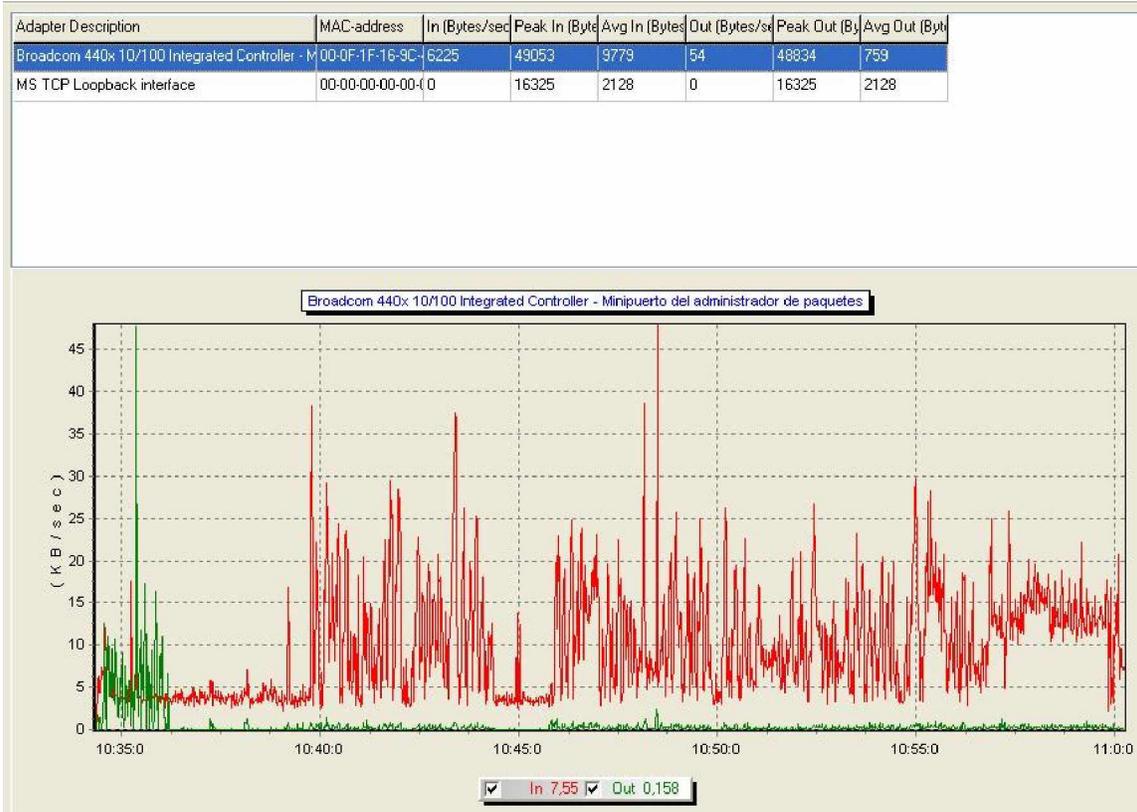


Figura 25. Tráfico: Cliente – LIR. Escenario 2 (una conexión)

Los datos obtenidos son:

Rata Tx Cliente video	Paq. Promedio In / seg	Pico In / seg	Paq. Promedio Out / seg	Pico Out /seg
291 kbps	78.232 kbps	392.424 Kbps	6.07 Kbps	390.672Kbps

Posteriormente, con dos conexiones, Figura 26:

Rata Tx Cliente video	Paq. Promedio In / seg	Pico In / seg	Paq. Promedio Out / seg	Pico Out /seg
291 kbps	104.400 kbps	582.424 Kbps	5.656 Kbps	390.672Kbps

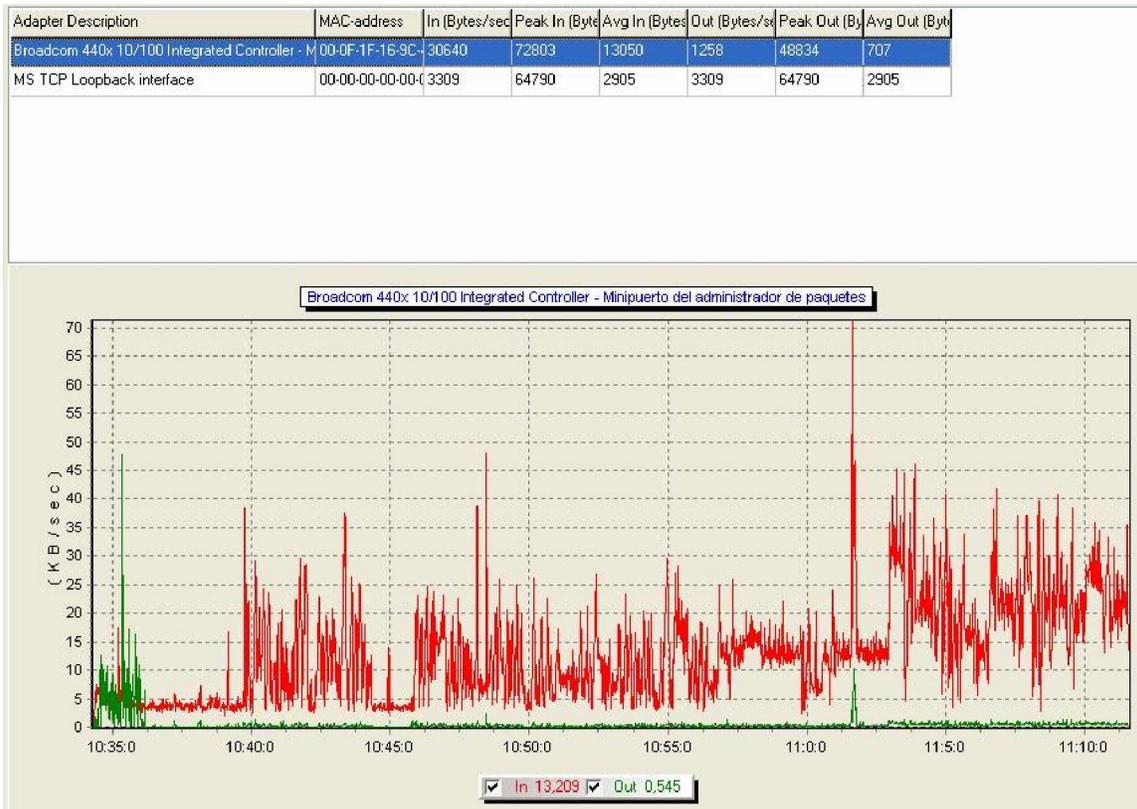


Figura 26. Tráfico: Cliente – LIR. Escenario 2 (dos conexiones)

Escenario 3. Conexión Telefónica. MODEM 33.6 Kbps

Finalmente, se realizaron pruebas semejantes bajo una conexión telefónica, obteniendo los siguientes resultados:

Para una conexión, Figura 27:

Rata Tx Cliente video	Paq. Promedio In / seg	Pico In / seg	Paq. Promedio Out / seg	Pico Out /seg
28 kbps	36.04 kbps	181.680 Kbps	1.368 Kbps	22.232 Kbps

Adapter Description	MAC-address	In (Bytes/sec)	Peak In (Byte)	Avg In (Bytes)	Out (Bytes/s)	Peak Out (B)	Avg Out (Byte)
National Semiconductor Corp. DP83815/816 1C	00-0D-9D-C8-A8 0	0	0	0	0	0	0
WAN (PPP/SLIP) Interface	00-53-45-00-00-00	5320	22710	4506	256	2779	171
MS TCP Loopback interface	00-00-00-00-00-00	58	58	38	58	58	38

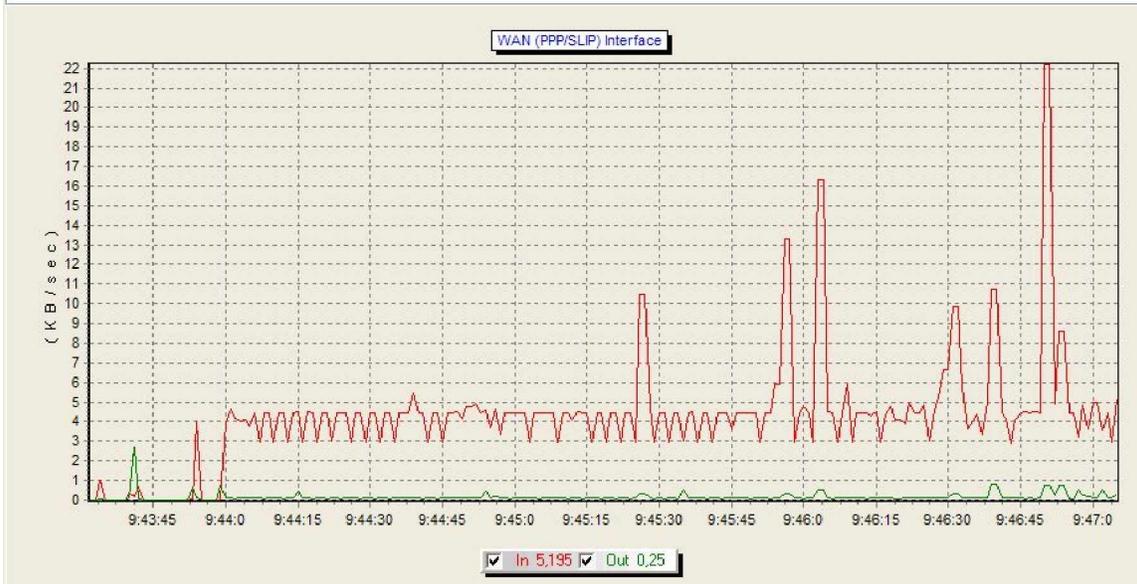


Figura 27. Tráfico: Cliente – LIR. Escenario 3 (Una conexión)

Para dos conexiones, Figura 28.

Los datos obtenidos son los siguientes:

Rata Tx Cliente video	Paq. Promedio In / seg	Pico In / seg	Paq. Promedio Out / seg	Pico Out /seg
28 kbps	35.424 kbps	181.680 Kbps	1.400 Kbps	22.232 Kbps

Adapter Description	MAC-address	In (Bytes/sec)	Peak In (Byte)	Avg In (Bytes)	Out (Bytes/sec)	Peak Out (By)	Avg Out (Byt)
National Semiconductor Corp. DP83815/816 10	00-0D-9D-C8-A8 0	0	0	0	0	0	0
WAN (PPP/SLIP) Interface	00-53-45-00-00-00	4542	22710	4428	80	2779	175
MS TCP Loopback interface	00-00-00-00-00-00	0	58	36	0	58	36

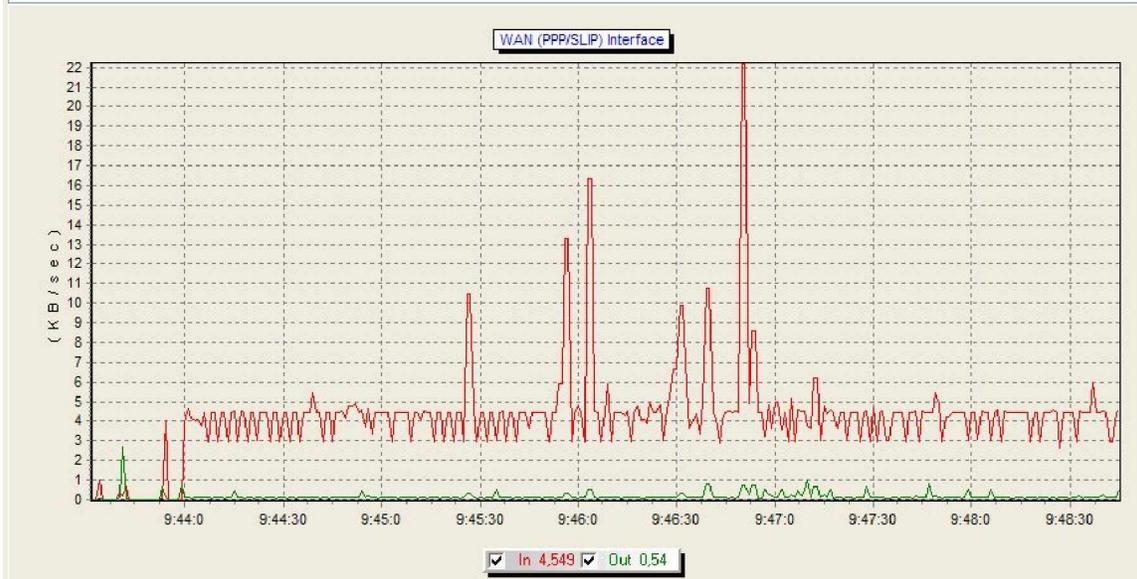


Figura 28. Tráfico: Cliente – LIR. Escenario 3 (Dos conexiones)

## 7.6 DISCUSIÓN DE RESULTADOS

Como se mencionó anteriormente, para la realización de las pruebas, las variables medidas dependen mucho del tipo de protocolo y en general de la información que se está transfiriendo, siendo la transmisión de video la que más ancho de banda consume o produce mayor congestión en la interfase de red, mientras que la velocidad de transmisión, por paquetes no es afectada sino por los escenarios en donde el enlace se realiza remotamente.

La transmisión de páginas php o jsp, en general se realiza a velocidades muy buenas en cualquiera de los escenarios, (entre 45 ms y 950 ms), lo que garantiza un flujo de información aceptable entre las estaciones.

La transmisión de video utilizando los codecs y tecnologías de multidifusión, permiten que el ancho de banda pase a un segundo plano, manipulando la calidad de la

transmisión, es decir, adaptando la rata de transmisión de acuerdo a la capacidad de ancho de banda del cliente. Es así como las transmisiones en la red LAN de la Universidad Autónoma de Manizales, en el peor de los casos, fue de 121 Kbps (Figura 24) por cliente para una transmisión de video continuo utilizando el Windows Media Encoder, y manejando un tiempo de almacenamiento de paquetes de aproximadamente 12 seg. Para el caso del Helix Server, el menor ancho de banda reportado para la LAN fue de 150 Kbps por cliente. Es decir, la transferencia de video, se adapta a las condiciones que se le presentan, y en promedio mantiene el mismo tiempo de almacenamiento o Buffering, de 12 segundos (Windows Media Encoder) y de 9 Segundos (Helix Producer / Helix Server). En un ambiente remoto, es probable que 9 o 12 segundos sea un tiempo demasiado extenso para el almacenamiento, ya que el retardo asociado en un ambiente de video es elevado, pero esto garantiza un flujo constante de la imagen.

Las tecnologías utilizadas para la difusión de video, fueron el Windows Media Server, con Windows Media Encoder y su tecnología asociada, WMV, y el Helix Server de Real Video, con protocolos RTP que funcionan mediante paquetes UDP. Ambos sistemas de codificación manejan Buffering entre 9 y 12 segundos.

Este inconveniente, puede ser solucionado utilizando otro tipo de protocolo o tecnología de difusión de video, que en este caso por el corto margen de tiempo no pudo ser tomado en cuenta.

En general, los aspectos técnicos relacionados con la comunicación entre el cliente y el laboratorio interactivo remoto, se encuentran dentro de los márgenes de aceptabilidad, que oscilan entre 7 y 12 segundos de espera de un cliente para comenzar a visualizar información en un sitio web.

## 8 CONCLUSIONES

---

Se propone un método de Administración y Construcción de Laboratorios Interactivos Remotos – LIR, que incluye los módulos de administración, comunicación, sesión, evaluación y construcción. Los modelos planteados dentro del método propuesto surgen de la categorización de los requerimientos y brindan las condiciones adecuadas en las funciones de acceso, control y seguridad de los LIR. Los modelos permiten la coordinación y la comunicación adecuada entre los actores, actividades y laboratorios; y a su vez, flexibilizan la realización de las prácticas docentes ofreciendo un ambiente integrado de fácil uso y alta velocidad.

El método propuesto de administración admite la inclusión o adaptación (en general y en sus diferentes modelos) de nuevos elementos tecnológicos en la administración y construcción de LIR. De esta manera, se brinda una alta flexibilidad y libertad en la implementación de laboratorios interactivos en las diferentes instituciones de educación, que tienen áreas de interés comunes en el tema de educación.

El método de administración propuesto se investigó, analizó y desarrolló en un sistema concreto de administración de laboratorios interactivos remotos sobre la red LAN de la Universidad Autónoma de Manizales (SALIR – Sistema de Administración de Laboratorios Interactivos Remotos), y posteriormente se aplicó el método de construcción planteado en un LIR enfocado al Telecontrol, lo que muestra la validez del método.

Las pruebas realizadas sobre el rendimiento del sistemas SALIR corresponden básicamente a velocidad de acceso y análisis de tráfico bajo diferentes escenarios de trabajo. Los resultados de las pruebas muestran que el método propuesto es adecuado para la administración de laboratorios interactivos remotos sobre IP. Según los datos obtenidos en el análisis mencionado, las diferencias en los escenarios de conexión son poco significativas en cuanto al transporte de protocolos livianos como http (menos de 1 segundo de tiempo de descarga en el peor de los casos), mientras que el transporte de

video exige la utilización de sistemas de difusión con ancho de banda adaptativo (entre 32 Kbps y 300 Kbps).

La implementación del método de Administración y Construcción de Laboratorios Interactivos Remotos en el presente trabajo, deja abiertas entre otras las siguientes tareas, para desarrollo en trabajo futuro en la Universidad Autónoma de Manizales: 1) Ampliación del compilador del LIR, para permitir instrucciones de control complejas con el fin de mejorar la interacción del móvil robótico con su medio, 2) Modificación del dispositivo robótico para brindarle autonomía de espacio, adicionando una interfaz inalámbrica de datos, 3) Adición de un sistema de video inalámbrico secundario en el móvil robótico, 4) Desarrollo de una herramienta de autor (autorware) compatible con el sistema de administración y con los estándares de e-learning para la adición de laboratorios interactivos remotos.

## 9 BIBLIOGRAFIA Y REFERENCIAS

---

- [1] FACUNDO, Angel. La Educación Superior Virtual en Colombia. Bogotá. UNESCO/IESALC, Febrero 2003. p. 6
- [2] <http://www.chi.itesm.mx/uv/historia.htm>.
- [3] La Red Mutis esta conformada por la Universidad Autónoma de Bucaramanga, la Universidad Autónoma de Manizales, la Corporación Universitaria del Tolima, la Universidad Minuto de Dios, la Universidad Tecnológica de Bolívar y la Corporación Autónoma de Occidente.
- [4] FACUNDO, Angel. La Educación Superior Virtual en Colombia. Bogotá. UNESCO/IESALC, Febrero 2003. p. 15
- [5] FACUNDO, Angel. La Educación Superior Virtual en Colombia. Bogotá. UNESCO/IESALC, Febrero 2003. p. 31
- [6] ZAPATA Zapata, Donna. Contextualización de la Enseñanza Virtual en la Educación Superior. Bogotá. ICFES 2002. p. 16.
- [7] FACUNDO, Angel. La Educación Superior Virtual en Colombia. Bogotá. UNESCO/IESALC, Febrero 2003. p. 29
- [8] FACUNDO Angel. Educación Virtual en América Latina y el Caribe: Características y Tendencias. Bogotá. UNESCO, Febrero 2002. p. 25
- [9] <http://www.mmi.unimaas.nl/>
- [10] FACUNDO Angel. Educación Virtual en América Latina y el Caribe: Características y Tendencias. Bogotá. UNESCO, Febrero 2002. p. 14
- [11] <http://moodle.org/doc/?file=background.html>
- [12] MEZA MARTINEZ, Jorge I. Implementación de aplicaciones web para el desarrollo rápido de intranets basadas en grupos de usuarios. Universidad Autónoma de Manizales. 2001.
- [13] <http://www.ielearning.com/wbt/courseware/index.cfm>
- [14] <http://www-10.lotus.com/ldd/today.nsf/lookup/DeployContentLMS>
- [15] <http://www.msglobal.org/specifications.html>
- [16] <http://www.adlnet.org>

- [17] <http://www.trainingfoundation.com/articles/default.asp?PageID=945>
- [18] <http://www.readygo.com/aicc/aic02/00aic02.htm>
- [19] <http://www.aicc.org>
- [20] HENAO ÁLVAREZ, Octavio., 1993. "El aula escolar del futuro". En: Revista Educación y Pedagogía, Vol. 4 (8-9), 87-96.
- [21] LEFLORE, D., 2000. "Theory supporting design guidelines for web-based instruction". En: Beverly Abbey (Ed.) Instructional and Cognitive Impacts of Web-Based Education. Hershey, PA: Idea Group Publishing.
- [22] UNIGARRO GUTIERREZ, Manuel Antonio. Educación Virtual: Encuentro Formativo en el Ciberespacio. Universidad Autonoma De Bucaramanga. 2001
- [23] <http://moodle.org/doc/?file=philosophy.html>
- [24] <http://moodle.org/doc/?file=features.html>
- [25] <http://www.ielearning.com/wbt/features/index.cfm>
- [26] El grupo LABLINK está compuesto por: ISIB (Institut Supérieur Industriel de Bruxelles – Bélgica), Silesian University of Technology (Polonia), VVL (Verbund Virtuelles Labor) , Hochschule Aalen (Universidad Aalen – Alemania), kauno kolegija (Colegio Kauno - Lituania), České vysoké učení technické (Universidad Técnica de Praga), KaHo Sint-Lieven Hogeschool (Universidad Católica de Sint-Lieven – Bélgica), Karel de grote Hogeschool (Universidad Karel de Grote – Bélgica) y el Centro de procesamiento paralelo en Bulgaria.
- [27] LABORATORY of Electronics. Telemetry and Telecontrol of devices through Internet. ISIB (Institut Supérieur Industriel de Bruxelles), Bruselas.
- [28] IBARRA Muñoz, José Jaime. Desarrollo de un Sistema Virtual para Acceso Remoto de Instrumentación por medio de Interfaz GPIB. Universidad Nacional de Colombia Sede Manizales. Manizales. 2002.
- [29] MARÍN Ballesteros, Hugo Andrés. Salgado Maldonado, Mauricio Alejandro. Laboratorio Interactivo Remoto Via Internet. Universidad Nacional de Colombia Sede Manizales. Manizales. 2002.
- [30] OLAYA Gómez, Nilson. Tovar Bustos, Edixon Fernando. Desarrollo de un sistema de monitoreo y manejo remoto sobre internet de un difractor de rayos x bruker d8 advanced. Universidad Nacional de Colombia Sede Manizales. Manizales. 2002.

- [31] EL LIBRO del RS232. Je Campbell. Edit. Anaya Multimedia
- [32] COMPLETE guide to RS232 and parallel conections. Martin D. Seyer. Prentice Hall. 1988
- [33] Free Serial Communication Software for RS232, RS422, RS485 and Modbus Devices: Troubleshooting the COM Port.  
<http://www.windmill.co.uk/serial.html>
- [34] <http://www.ines.de/gpibinfo.htm>
- [35] <http://www.scpiconsortium.org/>
- [36] ROBOT programming languages standardization in manufacturing environment Capizzi, A.; Messina, G.; Tricomi, G.; Intelligent Robots and Systems '93, IROS '93. Proceedings of the 1993 IEEE/RSJ International Conference on , Volume: 1 , 26-30 July 1993 Pages:488 - 492 vol.1
- [37] PRATT, Jerry 2000. Exploiting Inherent Robustness and Natural Dynamics in the Control of Bipedal Walking Robots. Ph.D. Thesis, Computer Science Department, Massachusetts Institute of Technology, Cambridge, Massachusetts, 2000

#### REFERENCIAS DE APOYO

- [r1] FREUND, E.; Ludemann-Ravit, B.; Stern, O.; Koch, T. Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on , Creating the architecture of a translator framework for robot programming languages. Volume: 1 , 2001. Pag:187 – 192.
- [r2] MOHAMED Califa and Rinky Lam. Web-Based Learning: Effects on Learning Process and Outcome. IEEE Transactions on Education. Vol 45. No 4. November 2002. Pag. 350 – 356
- [r3] FU, King Sun. Robotica: Control, deteccion, vision e inteligencia / K. S. Fu; R. C. Gonzalez; C. S. G. Lee. - Madrid: [McGraw-Hill](#), D. L. 1988. - XIII, 599 p.: il.
- [r4] ARKIN, Ronald. C. 1987 Motor Schema Based Mobile Robot Navigation, International Journal of Robotics Research 8.
- [r5] BROOKS, Rodney A. 1986. A Robust Layered Control System for a Mobile Robot. IEEE Transactions on Robotics and Automation.
- [r6] KOK Kiong Tan, Tong Heng Lee, and Chai Yee Soh. Internet-Based Monitoring of Distributed Control Systems. An Undergraduate Experiment. IEEE Transactions on Education. Vol 45. No 2. May 2002. Pag 128 – 134.

## Referencias de Laboratorios Virtuales

- [LabVir1] LABORATORIOS Virtuales. TEC de Monterrey, campus Cuernavaca.  
[http://www.mor.itesm.mx/~lsi/laboratorio\\_virtual.html??http://w3.mor.itesm.mx/~emoraes/REDII/redii\\_abajo.html](http://www.mor.itesm.mx/~lsi/laboratorio_virtual.html??http://w3.mor.itesm.mx/~emoraes/REDII/redii_abajo.html)
- [LabVir2] UNIVERSIDAD Nacional de Colombia Sede Manizales. Curso Interactivo de Física I.  
<http://www.manizales.unal.edu.co/cursofisica/>
- [LabVir3] LABORATORIO Virtual de Física de La universidad Nacional de Colombia sede Medellín  
<http://www.unalmed.edu.co/~daristiz/LABFIS/Principal/Labfis.htm>
- [LabVir4] SECCIÓN Tu Salud de VirtualLab.  
<http://scienceu.fsu.edu/espanol/content/virtuallab/healthyou/>
- [LabVir5] The ECOSSE Control HyperCourse. The Virtual Control Laboratory.  
<http://www.chemeng.ed.ac.uk/ecosse/control/course/map/>
- [LabVir6] UNIVERSIDAD Federal de Santa Catarina  
[http://www.lrv.eps.ufsc.br/port/menu/Principal\\_2.htm](http://www.lrv.eps.ufsc.br/port/menu/Principal_2.htm).
- [LabVir7] LABORATORIO virtual de eventos físicos.  
<http://physicsweb.org/vlab/>
- [LabVir8] NTNU Virtual Physics Laboratory  
<http://www.phy.ntnu.edu.tw/java/>
- [LabVir9] VIRTUAL Laboratories in Probability and Statistics  
<http://www.math.uah.edu/stat/>
- [LabVir10] UNIVERSIDAD de Valladolid. Departamento de Física de la Materia Condensada Cristalografía y Mineralogía.  
<http://goya.eis.uva.es/Simulacion.asp>
- [LabVir11] <http://www.cesvirtual.edu.co/mod/resource/view.php?id=1178>
- [LabVir12] <http://www.mor.itesm.mx/~lsi/redii/fasel.html> Control de Nomad
- [LabVir13] [http://www.mor.itesm.mx/~lsi/redii/telecontrol/control\\_distancia.html](http://www.mor.itesm.mx/~lsi/redii/telecontrol/control_distancia.html)
- [LabVir14] Laboratorio virtual de robótica móvil.  
[http://www.mor.itesm.mx/~lsi/redii/Lab\\_vir\\_der.html](http://www.mor.itesm.mx/~lsi/redii/Lab_vir_der.html)
- [LabVir15] BÉLGICA Laboratorio de electrónica. Telemetría y telecontrol de instrumentos. <http://lablink.isib.be/>

- [LabVir16] UNIVERSIDAD Aalen - Alemania. <http://www.fh-aalen.de/>
- [LabVir17] VERBUND Virtuelle Labor Alemania.  
<http://vvl6.fh-konstanz.de/VVL/DE/Overview.html>
- [LabVir18] COLEGIO Kauno - Lituania <http://www.kauko.lt/kaukas/kolegija.php>
- [LabVir19] UNIVERSIDAD Técnica de Praga. <http://www.cvut.cz/cz/>
- [LabVir20] UNIVERSIDAD Católica de Sint-Lieven – Bélgica. <http://www.kahosl.be/>
- [LabVir21] UNIVERSIDAD Karel de Grote – Bélgica. <http://www.iwt-kdg.be/>
- [LabVir22] CENTRO de procesamiento paralelo en Bulgaria. <http://www.acad.bg/>

## ANEXOS

---

ANEXO A. TABLA CON PRODUCTOS COMERCIALES CERTIFICADOS POR LA AICC



Products Certified Compliant to AGR-010 (Web-based CMI Guidelines)

<u>Vendor</u>	<u>Product Name(s)</u>	<u>Product Type</u>	<u>Version</u>	<u>Expires</u>
Thomson NETg Corporation	Adobe GoLive 6 Fundamentals	CBT Course (Web-based)	version 15012-0307-10	14-Sep-2006
Pathlore Software Corporation	Pathlore Learning Management System	CMI System (Web-based)	Release 6.0.00.197	16-Aug-2006
SkillSoft Corporation	See certification statement for a list of product names	CBT Course(s) (Web-based)	See certification statement	10-May-2006
ABIFORMAZIONE - Divisione di ABI SERVIZI S.p.A.	See certification statement for a list of product names	CBT Course(s) (Web-based)	Version 1.0	06-Apr-2006
Docent Inc.	Docent Learning Management Server	CMI System (Web-based)	Version 6.5 SP3	27-Feb-2006
Macromedia	Authorware	Authoring System (Web-based)	Version 7.0	30-Jan-2006
Question Mark Computing, Ltd.	Perception	Courseware Generation/ Assessment System (Web-based)	Version 3.4	15-Dec-2005
Maritz Learning	LIBRIX Performance Management System	CMI System (Web-based)	Version 4.5.5	08-Sep-2005
WBT Systems	TopClass	CMI System (Web-based)	Version 6.1.3	18-Aug-2005
THINQ	ThinQ Learning	CMI System	Version 5.2	04-Aug-2005

Learning Solutions	Management System	(Web-based)	with patch 5.2.0.TQ006	
Pathlore Software Corporation	Pathlore LMS	CMI System (Web-based)	Release: 5.5.02.96	04-Aug-2005
Plateau Systems, Ltd.	Plateau 4 LMS	CMI System (Web-based)	Version 4.1.3	18-Jul-2005
PCCW Ltd.	ConXerto Achieve	CMI System (Web-based)	Version 2.0	08-Dec-2004
DIDA GROUP S.p.A.	In.Form@	CMI System (Web-based)	Version 1.0.0	01-Nov-2004
Miraenet Co., Ltd.	SpeedLearn	CMI System (Web-based)	Version 1.0	25-Oct-2004

Fuente: <http://www.aicc.org/pages/cert.htm>

---



---

## ANEXO B. CASOS DE USO, DIAGRAMAS DE INTERACCIÓN Y CONTRATOS DEL MODELO DE ADMINISTRACIÓN

---

A continuación se presentan los casos de uso, los diagramas de interacción y los contratos para las actividades relevantes del sistema. Sin embargo como muchas de ellas son actividades básicas y triviales, solo se presenta el caso de uso de alto nivel y sus diferencias se notan principalmente en las precondiciones y poscondiciones de los contratos.

Caso de uso: Registrar Laboratorio.

Actores: Administrador del sistema.

Tipo: Secundario.

Descripción: El administrador debe hacer registro de un nuevo LIR dentro del módulo de administración, por lo cual toma los datos pertinentes del mismo para crear el registro en la BD.

### Curso normal de eventos

Administrador	Sistema
0. Selecciona la opción Laboratorio / Registrar	1. Presenta la interfaz para ingresar los datos.
2. Ingresa los datos correspondientes.	
3. Al terminar de digitar los datos da clic en Guardar.	4. Hace validación de los datos.
	5. Guarda el registro del Laboratorio en la tabla Laboratorio.
	6. Notifica que se ha guardado satisfactoriamente el registro.
7. Da clic en Aceptar.	8. Presenta nuevamente la interfaz para ingresar otro Laboratorio.
8. Da clic en Salir.	

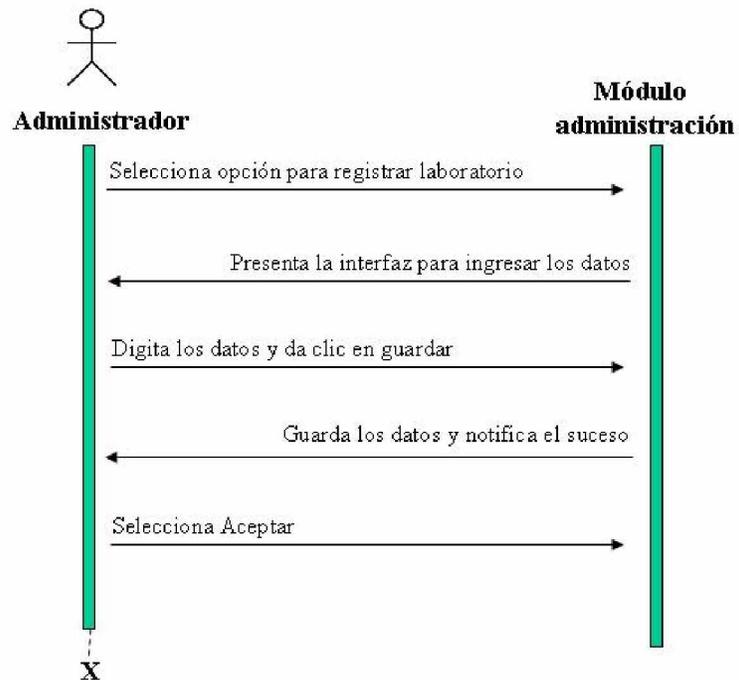
### Cursos alternos.

2. Si da clic en Salir, el sistema le presenta la página de portada y los datos no son grabados.

4. Si algún dato no es correcto se presenta un mensaje con una notificación del hecho y se pasa al paso 2.

8. Si desea ingresar los datos de otro Laboratorio va al paso 2.

## Diagrama de interacción: Registrar Laboratorio



### Contratos:

Nombre: Selecciona opción para registrar laboratorio.

Responsabilidades: Presentación de menú u opciones para escoger la opción registrar laboratorio.

Precondiciones:

El usuario autenticado debe ser el administrador del sistema.  
 $sesion\_id \in id\_administrador$

Poscondiciones:

El sistema presenta la interfase para ingresar los datos del LIR.

Nombre: Digitación de datos.

Responsabilidades: Capturar todos los datos correspondientes a un LIR y almacenarlos en la BD.

Precondiciones:

Selección de la opción registrar laboratorio.

Poscondiciones:

El id del laboratorio no debe estar registrado en el sistema.

$Id\_laboratorio \notin tabla\ Laboratorio$

El archivo correspondiente a la documentación del laboratorio se debe haber subido al directorio docs\_lab en el servidor, con el nombre doclab[id\_laboratorio].ext\_original\_de\_arch

Los campos obligatorios deben poseer la información correspondiente.

box\_id\_laboratorio.value ≠ vacío

box\_ip.value ≠ vacío

box\_puerto ≠ vacío

box\_path.value ≠ vacío

box\_duracion\_practica.value ≠ vacío

box\_numero\_notas.value ≠ vacío

box\_documentacion.value ≠ vacío

box\_autoevaluacion.value ≠ vacío

Se creó un registro en la tabla laboratorio con los datos correspondientes.

Presenta en pantalla la notificación del hecho.

Caso de uso: Editar Laboratorio.

Actores: Administrador del sistema.

Tipo: Secundario.

Descripción: El administrador necesita consultar o modificar los datos de un LIR.

Curso normal de eventos

Administrador	Sistema
0. Selecciona la opción Laboratorio / Editar.	1. Presenta la interfaz para seleccionar el LIR a editar.
2. Selecciona el LIR y oprime en Editar.	3. Presenta la interfaz con los datos del LIR
4. Digita los datos y luego da clic en Guardar.	5. Hace validación de los datos.
	6. Modifica el registro del Laboratorio en la tabla Laboratorio.
	7. Notifica que se ha guardado satisfactoriamente el registro.
8. Da clic en Aceptar.	9. Presenta nuevamente la interfaz para seleccionar otro LIR.
10. Da clic en Salir.	

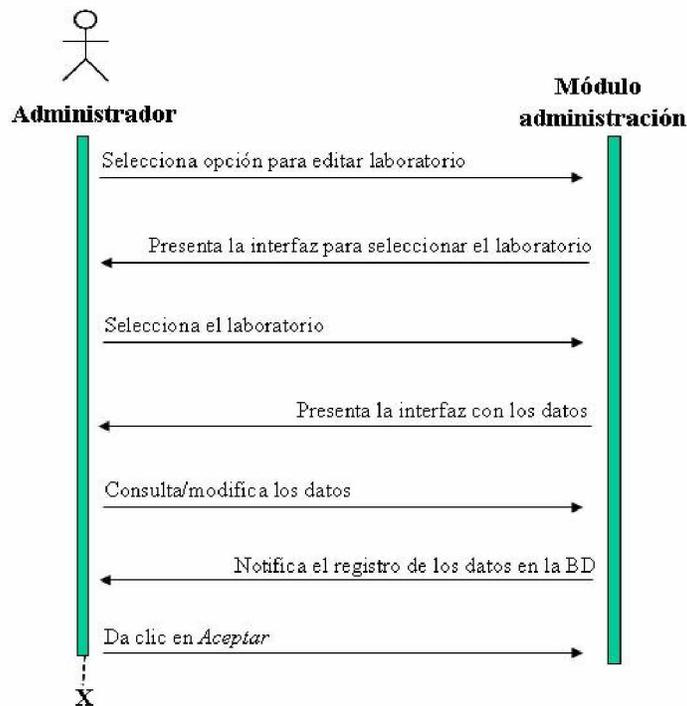
Cursos alternos.

2,4 Si da clic en Salir, el sistema le presenta la página de portada y los datos no son grabados.

4. Si algún dato no es correcto se presenta un mensaje con una notificación del hecho y se pasa al paso 4.

10. Si selecciona otro LIR va al paso 3.

## Diagrama de interacción: Editar Laboratorio



### Contratos

Nombre: Selecciona opción para editar laboratorio.

Responsabilidades: Presentación de menú u opciones para escoger la opción registrar laboratorio.

Precondiciones:

El usuario autenticado debe ser el administrador del sistema.  
 $sesion\_id \in id\_administrador$

Poscondiciones:

El sistema presenta la interfase para seleccionar el laboratorio.

Nombre: Selección de laboratorio.

Responsabilidades: permitir seleccionar un laboratorio y presentar sus datos.

Precondiciones:

Selección de la opción editar laboratorio.

Poscondiciones:

El sistema presenta la interfase para ingresar los datos del LIR.

Nombre: Consulta/modifica los datos.

Responsabilidades: Presentar/Capturar todos los datos correspondientes a un LIR.

Precondiciones:

Selección del laboratorio correspondiente.

El campo id\_laboratorio no es modificable.

Poscondiciones:

Si se ha modificado el archivo de documentación, este se debe haber subido al directorio docs\_lab en el servidor, con el nombre doclab[id\_laboratorio].ext\_original\_de\_arch, se debió reescribir el anterior.

Los campos obligatorios deben poseer la información correspondiente.

box\_id\_laboratorio.value  $\neq$  vacío

box\_ip.value  $\neq$  vacío

box\_puerto  $\neq$  vacío

box\_path.value  $\neq$  vacío

box\_duracion\_practica.value  $\neq$  vacío

box\_numero\_notas.value  $\neq$  vacío

box\_documentacion.value  $\neq$  vacío

box\_autoevaluacion.value  $\neq$  vacío

Si se modificó el número de notas de un laboratorio, todos los archivos de notas de los grupos que están relacionados con el mismo deben ajustarse para que coincida con la definición del laboratorio así:

Si ( num\_notas = 0  $\wedge$  num\_notasAjustado > 0) =>

Crear en el directorio notas el archivo notas\_grp[id\_grupo]  $\forall$  grupos del LIR

Si ( num\_notas > 0  $\wedge$  num\_notasAjustado = 0) =>

Borrar en el directorio notas el archivo notas\_grp[id\_grupo]  $\forall$  grupos del LIR

Si (num\_notas  $\neq$  0  $\wedge$  num\_notasAjustado  $\neq$  0  $\wedge$  num\_notas  $\neq$  num\_notasAjustado) =>

Ajustar en el directorio notas el archivo notas\_grp[id\_grupo]  $\forall$  grupos del LIR

Se modificó el registro en la tabla laboratorio con los datos correspondientes.

Presenta en pantalla la notificación del hecho.

Caso de uso: Eliminar Laboratorio.

Actores: Administrador del sistema.

Tipo: Secundario.

Descripción: Si un LIR ha quedado mal registrado es posible eliminar su registro.

Curso normal de eventos

Administrador	Sistema
0. Selecciona la opción Laboratorio / Eliminar.	1. Presenta la interfaz para seleccionar el LIR a editar.
2. Selecciona el LIR y oprime en Eliminar.	3. Valida que si se pueda eliminar.
	4. Presenta la interfaz con los datos del LIR
5. Verifica que es el LIR a eliminar y luego da clic en Confirmar eliminar.	6. Elimina el registro de la tabla laboratorio.
	7. Notifica que el registro ha sido eliminado
8. Da clic en Aceptar.	9. Presenta nuevamente la interfaz para seleccionar otro LIR.

10. Da clic en Salir.

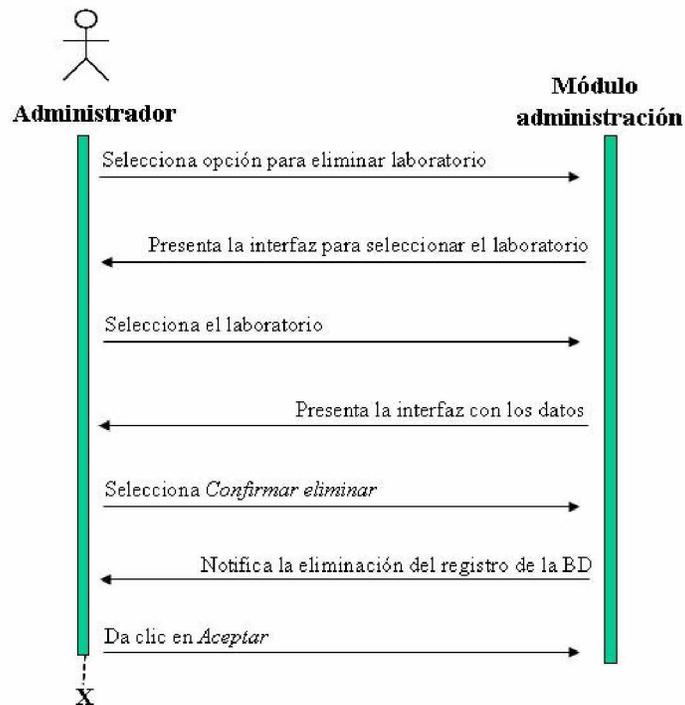
Cursos alternos.

2,4 Si da clic en Salir, el sistema le presenta la página de portada.

3 Si el LIR no se puede eliminar se notifica con un mensaje y se va al paso 2.

10. Si selecciona otro LIR va al paso 3.

## Diagrama de interacción: Eliminar Laboratorio



### Contratos:

Nombre: Selecciona opción para eliminar laboratorio.

Responsabilidades: Presentación de menú u opciones para escoger la opción eliminar laboratorio.

Precondiciones:

El usuario autenticado debe ser el administrador del sistema.

$sesion\_id \in id\_administrador$

Poscondiciones:

El sistema presenta la interfase para seleccionar el laboratorio.

Nombre: Selección de laboratorio.

Responsabilidades: permitir seleccionar un laboratorio y presentar sus datos.

Precondiciones:

Selección de la opción editar laboratorio.  
Poscondiciones:  
El sistema presenta la interfase para visualizar los datos del LIR.

Nombre: Selección Confirmar eliminar.  
Responsabilidades: Eliminar el registro de la base de datos.

Excepciones:  
Si el laboratorio seleccionado posee grupos relacionados no se puede eliminar y se notifica el hecho.

Precondiciones:  
Selección del laboratorio correspondiente.

Poscondiciones:  
Se ha borrado el registro en la tabla laboratorio  
`id_laboratorio`  $\notin$  `tabla_laboratorio`

Caso de uso: Listar Laboratorios.  
Actores: Administrador del sistema / Tutor / Estudiante  
Tipo: Secundario/  
Descripción: Se desea consultar un listado de los LIR registrados.

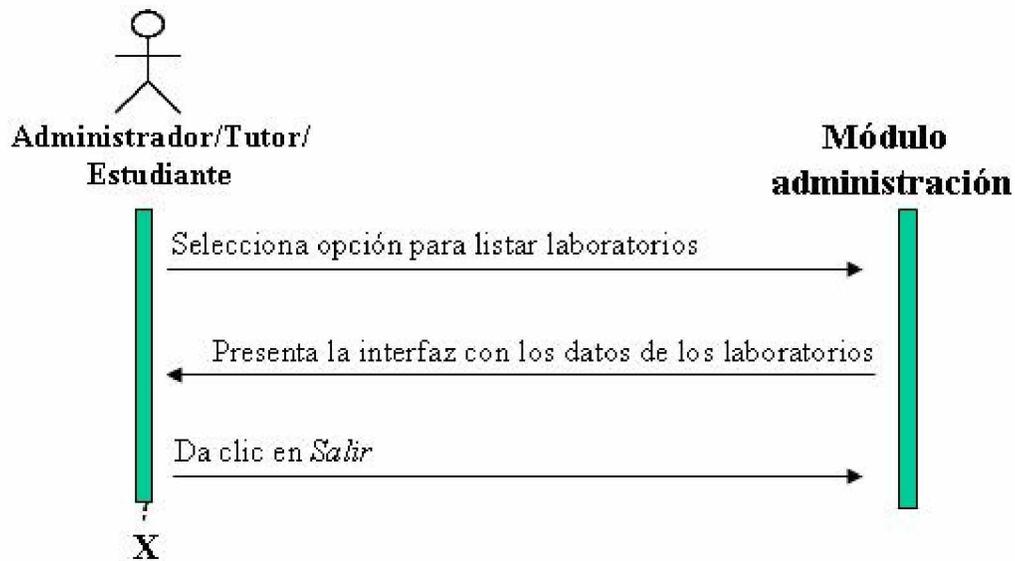
#### Curso normal de eventos

Administrador	Sistema
0. Selecciona la opción Laboratorio / Listar.	1. Presenta la interfaz con el listado de los LIR.
2. Da clic en Salir.	

#### Cursos alternos.

1, Si se desea descargar el archivo de documentación de un LIR, se oprime sobre el nombre del archivo y este se graba localmente.

## Diagrama de interacción: Listar Laboratorios



### Contratos:

Nombre: Selecciona opción para listar laboratorios.

Responsabilidades: Presentación de menú u opciones para escoger la opción listar laboratorio.

Excepciones:

Si el usuario desea descargar el archivo de documentación del laboratorio puede dar clic sobre el nombre del archivo y este se descarga en la máquina localmente.

Precondiciones:

El usuario debe haber sido autenticado  
 $sesion\_id \in id\_administrador \vee sesion\_id \in tabla\_tutores \vee sesion\_id \in tabla\_estudiantes$

Poscondiciones:

El sistema presenta la interfase con la información de los laboratorios.

$\forall$  laboratorio se creó un enlace para descargar su archivo de documentación.

Caso de uso: Registrar Tutor.

Actores: Administrador del sistema / Tutor.

Tipo: Secundario.

Descripción: Este caso de uso se presenta cuando es necesario registrar en el sistema un nuevo tutor.

Curso normal de eventos. Idem a Registrar Laboratorio pero en el paso 5 se guarda el registro en la tabla Tutores y en los pasos donde se menciona Laboratorio se asume Tutor.

Diagrama de interacción: Registrar tutor: posee la misma secuencia de registrar laboratorio.

### Contratos:

Nombre: Selecciona opción para registrar tutor.

Responsabilidades: Presentación de menú u opciones para escoger la opción registrar tutor.

Precondiciones:

El usuario autenticado debe ser el administrador del sistema o un tutor.  
sesion\_id ∈ id\_administrador ∨ sesion\_id ∈ tabla\_tutor"

Poscondiciones:

El sistema presenta la interfase para ingresar los datos del Tutor.

Nombre: Digitación de datos.

Responsabilidades: Capturar todos los datos correspondientes a un tutor y almacenarlos en la BD.

Precondiciones:

Selección de la opción registrar tutor.

Poscondiciones:

El id del tutor no debe estar registrado en el sistema.

Id\_tutor ∉ tabla\_Tutores

Los campos obligatorios deben poseer la información correspondiente.

box\_id\_tutor.value ≠ vacío

box\_nombre.value ≠ vacío

box\_contrasenia.value ≠ vacío

box\_e\_mail.value ≠ vacío

Se creó un registro en la tabla tutores con los datos correspondientes.

Caso de uso: Editar Tutor.

Actores: Administrador del sistema / Tutor.

Tipo: Secundario.

Descripción: El administrador o un Tutor necesitan consultar o modificar los datos de un Tutor.

Curso normal de eventos. Idem a editar Laboratorio pero en el paso 6 se modifica el registro correspondiente a la tabla de Tutores y en los pasos donde se menciona Laboratorio se asume Tutor

Diagrama de Interacción Editar Tutor: posee la misma secuencia de editar laboratorio.

## Contratos

Nombre: Selecciona opción para editar tutor.

Responsabilidades: Presentación de menú u opciones para escoger la opción editar tutor.

Precondiciones:

El usuario autenticado debe ser el administrador del sistema o un tutor.

$sesion\_id \in id\_administrador \vee sesion\_id \in tabla\_tutor$

Poscondiciones:

El sistema presenta la interfase para seleccionar el tutor.

Nombre: Selección de tutor.

Responsabilidades: permitir seleccionar un tutor y presentar sus datos.

Precondiciones:

Selección de la opción editar tutor.

Poscondiciones:

El sistema presenta la interfase para ingresar los datos de un tutor.

Nombre: Consulta/modifica los datos.

Responsabilidades: Presentar/Capturar todos los datos correspondientes a un tutor.

Precondiciones:

Selección del tutor correspondiente.

Poscondiciones:

Los campos obligatorios deben poseer la información correspondiente.

$box\_id\_tutor.value \neq vacio$

$box\_nombre.value \neq vacio$

$box\_contrasenia.value \neq vacio$

$box\_e\_mail.value \neq vacio$

Se modificó el registro en la tabla tutores con los datos correspondientes.

Presenta en pantalla la notificación del hecho.

Caso de uso: Eliminar Tutor.

Actores: Administrador del sistema /Tutor

Tipo: Secundario.

Descripción: Si un Tutor ha quedado mal registrado es posible eliminar su registro.

Curso normal de eventos. Idem a Eliminar Laboratorio, pero en el paso 6 se elimina el registro de la tabla Tutores y en los pasos donde se menciona Laboratorio se asume Tutor

Diagrama de interacción Eliminar tutor. Idem a eliminar laboratorio.

## Contratos:

Nombre: Selecciona opción para eliminar tutor.

Responsabilidades: Presentación de menú u opciones para escoger la opción eliminar tutor.

Precondiciones:

El usuario autenticado debe ser el administrador del sistema o un tutor  
 $sesion\_id \in id\_administrador \vee sesion\_id \in tabla\_tutores$

Poscondiciones:

El sistema presenta la interfase para seleccionar el tutor.

Nombre: Selección de tutor.

Responsabilidades: permitir seleccionar un tutor y presentar sus datos.

Precondiciones:

Selección de la opción editar tutor.

Poscondiciones:

El sistema presenta la interfase para visualizar los datos del tutor.

Nombre: Selección Confirmar eliminar.

Responsabilidades: Eliminar el registro de la base de datos.

Excepciones:

Si el tutor seleccionado posee grupos relacionados no se puede eliminar y se notifica el hecho.

Precondiciones:

Selección del tutor correspondiente.

Poscondiciones:

Se ha borrado el registro en la tabla tutor  
 $id\_tutor \notin tabla\_tutor$

Caso de uso: Listar tutores.

Actores: Administrador del sistema / Tutor / Estudiante

Tipo: Secundario/

Descripción: Se desea consultar un listado de los tutores registrados.

Curso normal de eventos. Idem a listar laboratorios

Diagrama de interacción: Listar laboratorios

Contratos:

Nombre: Selecciona opción para listar tutores.

Responsabilidades: Presentación de menú u opciones para escoger la opción listar tutores.

Precondiciones:

El usuario debe haber sido autenticado

$sesion\_id \in id\_administrador \vee sesion\_id \in tabla\_tutores \vee sesion\_id \in tabla\_estudiantes$

Poscondiciones:

El sistema presenta la interfase con la información de los tutores.

Caso de uso: Registrar Grupo.

Actores: Administrador del sistema / Tutor.

Tipo: Secundario.

Descripción: Este caso de uso se presenta cuando es necesario registrar en el sistema un nuevo grupo.

Curso normal de eventos. Idem a registrar laboratorio pero en el paso 5 se guarda el registro en la tabla Grupos y en los pasos donde se menciona Laboratorio se asume Grupo.

Diagrama de interacción: Registrar tutor: posee la misma secuencia de registrar laboratorio.

#### Contratos:

Nombre: Selecciona opción para registrar grupo.

Responsabilidades: Presentación de menú u opciones para escoger la opción registrar tutor.

Precondiciones:

El usuario autenticado debe ser el administrador del sistema o un tutor.

$sesion\_id \in id\_administrador \vee sesion\_id \in tabla\_tutor$

Poscondiciones:

El sistema presenta la interfase para ingresar los datos del grupo.

Nombre: Digitación de datos.

Responsabilidades: Capturar todos los datos correspondientes a un grupo y almacenarlos en la BD.

Precondiciones:

Selección de la opción registrar grupo.

Poscondiciones:

El id del grupo no debe estar registrado en el sistema.

$Id\_grupo \notin tabla\_Grupos$

El archivo correspondiente a la documentación del grupo se debe haber subido al directorio docs\_gru en el servidor, con el nombre docgrp[id\_grupo].ext\_original\_de\_arch

Los campos obligatorios deben poseer la información correspondiente.

$box\_id\_grupo.value \neq vacío$

$box\_contrasenia.value \neq vacío$

$box\_laboratorio \neq vacío$

Se creó un registro en la tabla grupos con los datos correspondientes.

Presenta en pantalla la notificación del hecho.

Caso de uso: Editar Grupo.

Actores: Administrador del sistema / Tutor.

Tipo: Secundario.

Descripción: El administrador o un Tutor necesitan consultar o modificar los datos de un Grupo.

Curso normal de eventos. Idem a editar Laboratorio pero en el paso 6 se modifica el registro correspondiente a la tabla de Tutores y en los pasos donde se menciona Laboratorio se asume Tutor

Diagrama de Interacción Editar Tutor: posee la misma secuencia de editar grupo.

### Contratos

Nombre: Selecciona opción para editar grupo.

Responsabilidades: Presentación de menú u opciones para escoger la opción editar grupo.

Precondiciones:

El usuario autenticado debe ser el administrador del sistema o un tutor.

$sesion\_id \in id\_administrador \vee sesion\_id \in tabla\_tutor$

Poscondiciones:

El sistema presenta la interfase para seleccionar el grupo.

Nombre: Selección de grupo.

Responsabilidades: permitir seleccionar un grupo y presentar sus datos.

Precondiciones:

Selección de la opción editar grupo.

Poscondiciones:

El sistema presenta la interfase para con los datos de un grupo.

Nombre: Consulta/modifica los datos.

Responsabilidades: Presentar/Capturar todos los datos correspondientes a un grupo.

Precondiciones:

Selección del grupo correspondiente.

El campo id\_grupo no es modificable

Poscondiciones:

Los campos obligatorios deben poseer la información correspondiente.

$box\_contrasenia.value \neq vacio$

$box\_laboratorio.value \neq vacio$

Se modificó el registro en la tabla grupos con los datos correspondientes.

Presenta en pantalla la notificación del hecho.

Si se modificó el laboratorio al cual está relacionado el grupo, se debe hacer el ajuste correspondiente del archivo de notas  $notas\_grp[id\_grupo]$ , según la cantidad de notas del laboratorio.

Caso de uso: Eliminar Grupo.

Actores: Administrador del sistema /Tutor

Tipo: Secundario.

Descripción: Si un Grupo ha quedado mal registrado es posible eliminar su registro.

Curso normal de eventos. Idem a Eliminar Laboratorio, pero en el paso 6 se elimina el registro de la tabla Grupos y en los pasos donde se menciona Laboratorio se asume Grupo.

Diagrama de interacción Eliminar grupo. posee la misma secuencia de eliminar laboratorio.

### Contratos:

Nombre: Selecciona opción para eliminar grupo.

Responsabilidades: Presentación de menú u opciones para escoger la opción eliminar grupo.

Precondiciones:

El usuario autenticado debe ser el administrador del sistema o un tutor  
 $sesion\_id \in id\_administrador \vee sesion\_id \in tabla\_tutores$

Poscondiciones:

El sistema presenta la interfase para seleccionar el tutor.

Nombre: Selección de grupo.

Responsabilidades: permitir seleccionar un grupo y presentar sus datos.

Precondiciones:

Selección de la opción editar grupo.

Poscondiciones:

El sistema presenta la interfase para visualizar los datos del grupo.

Nombre: Selección Confirmar eliminar.

Responsabilidades: Eliminar el registro de la base de datos.

Excepciones:

Si el grupo seleccionado posee estudiantes relacionados no se puede eliminar y se notifica el hecho.

Precondiciones:

Selección del grupo correspondiente.

Poscondiciones:

Se ha borrado el registro en la tabla grupo  
 $id\_grupo \notin tabla\_grupo$

Caso de uso: Listar grupos.

Actores: Administrador del sistema / Tutor / Estudiante

Tipo: Secundario/

Descripción: Se desea consultar un listado de los tutores registrados.

Curso normal de eventos. Idem a listar laboratorios

Diagrama de interacción: posee la misma secuencia de Listar laboratorios.

### Contratos:

Nombre: Selecciona opción para listar grupos.

Responsabilidades: Presentación de menú u opciones para escoger la opción listar grupos.

Excepciones:

Si el usuario desea descargar el archivo de documentación del grupo puede dar clic sobre el nombre del archivo y este se descarga en la máquina localmente.

Precondiciones:

El usuario debe haber sido autenticado

$sesion\_id \in id\_administrador \vee sesion\_id \in tabla\_tutores \vee sesion\_id \in tabla\_estudiantes$

Poscondiciones:

El sistema presenta la interfase con la información de los tutores.

$\forall$  grupo se creó un enlace para descargar su archivo de documentación.

Caso de uso: Registrar Tutor a un Grupo.

Actores: Administrador del sistema / Tutor

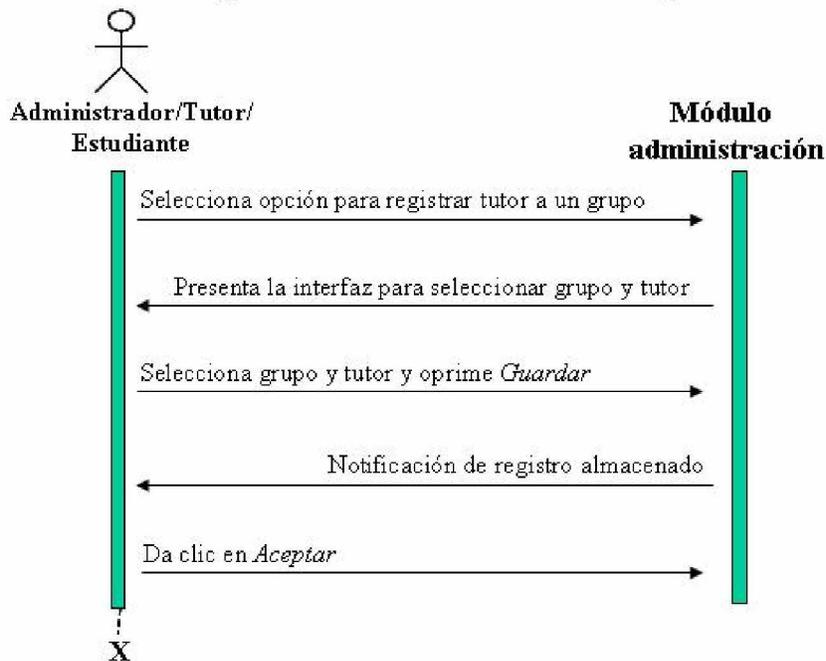
Tipo: Secundario/

Descripción: Se desea registrar tutor a un grupo para que haga seguimiento de las actividades y notas de los estudiantes.

Curso normal de eventos.

Administrador	Sistema
0. Selecciona la opción Grupo / Registrar tutor a un grupo.	1. Presenta la interfaz para seleccionar el grupo y el tutor a relacionar.
2. Selecciona el grupo y el tutor y oprime en Guardar.	3. Valida que ya no esté registrado el tutor en el grupo y guarda la información en la tabla de índices rel_grupos_tutores.
	4. Notifica que el registro ha sido guardado.
5. Da clic en Aceptar.	

## Diagrama de interacción: Registrar Tutor a un Grupo



Contratos:

Nombre: Selecciona opción para registrar tutor a un grupo.

Responsabilidades: Presentación de menú u opciones para escoger la opción registrar tutor a un grupo.

Precondiciones:

El usuario autenticado debe ser el administrador del sistema.  
 $sesion\_id \in id\_administrador$

Poscondiciones:

El sistema presenta la interfase para seleccionar el grupo y el tutor.

Nombre: Selecciona grupo y tutor.

Responsabilidades: relacionar un tutor a un grupo.

Excepciones:

Si el tutor ya estaba registrado en el grupo se presenta un mensaje notificando el hecho.

Precondiciones:

Haber seleccionado la opción registrar tutor a un grupo.

Poscondiciones:

La relación entre grupo y tutor queda en un registro de la tabla `rel_grupos_tutores`.

Se notifica que la relación/registro quedó grabada satisfactoriamente.

Caso de uso: Registrar estudiante en un grupo.

Actores: Administrador del sistema / Tutor

Tipo: Secundario/

Descripción: Se desea registrar un estudiante en un grupo para que pueda realizar prácticas en el LIR relacionado para el grupo.

Curso normal de eventos. Idem a Registrar tutor a un grupo, pero en el paso 3 la información se graba en la tabla rel\_grupos\_est.

Diagrama de interacción: Posee la misma secuencia de Registrar tutor a un grupo.

Contratos:

Nombre: Selecciona opción para registrar estudiante en un grupo.

Responsabilidades: Presentación de menú u opciones para escoger la opción registrar estudiante en un grupo.

Precondiciones:

El usuario autenticado debe ser el administrador del sistema o un tutor.  
 $sesion\_id \in id\_administrador \vee sesion\_id \in tabla\_tutores$

Poscondiciones:

El sistema presenta la interfase para seleccionar el grupo y el estudiante.

Nombre: Selecciona grupo y estudiante.

Responsabilidades: relacionar un estudiante en un grupo.

Excepciones:

Si el estudiante ya estaba registrado en el grupo se presenta un mensaje notificando el hecho.

Precondiciones:

Haber seleccionado la opción registrar estudiante en un grupo.

Poscondiciones:

La relación entre grupo y estudiante queda en un registro de la tabla rel\_grupos\_est.  
Se notifica que la relación/registro quedó grabada satisfactoriamente.

Caso de uso: Registrar estudiantes en un grupo (por archivo).

Actores: Administrador del sistema / Tutor

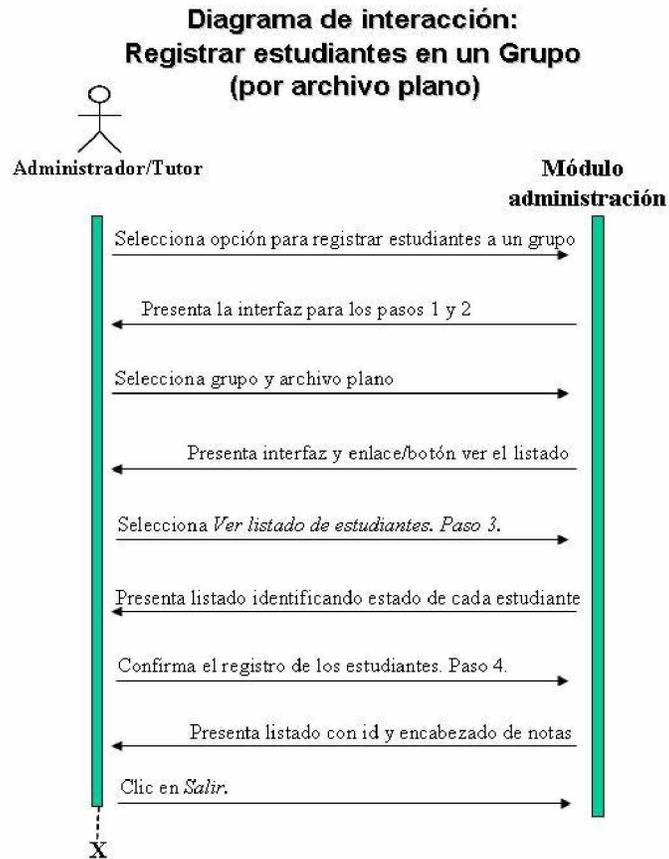
Tipo: Secundario/

Descripción: Se desea registrar un grupo de estudiantes en un grupo empleando un archivo plano, con el fin de que puedan realizar prácticas en el LIR relacionado para el grupo.

Curso normal de eventos.

Administrador	Sistema
0. Selecciona la opción Grupo / Registrar	1. Presenta la interfaz para realizar los dos

estudiantes en un grupo (archivo).	primeros pasos: 1. Seleccionar el grupo y 2 Seleccionar el archivo plano.
2. Selecciona el grupo y el archivo y oprime en Subir archivo ó Continuar.	3. Presenta el resultado de la transferencia del archivo y presenta un enlace ó botón para ver el listado de estudiantes (paso 3).
4. Selecciona Ver listado de estudiantes.	5. Presenta el listado de estudiantes notificando por cada uno su estado en el sistema a través de una convención de números y colores: 0 (Rojo). Registrado en el sistema y en el grupo. 1. (Amarillo). Registrado en el sistema pero NO en el grupo 2. (Verde). Sin registrar en el sistema ni en el grupo.  Se presenta un enlace/botón para el 4 paso: Confirmar registro de estudiantes.
6. Da clic en Confirmar.	7. Presenta un listado con el id de los estudiantes y un encabezado identificando el número de notas.
8. Da clic en Salir.	



Contratos:

Nombre: Selecciona opción para registrar estudiantes a un grupo.

Responsabilidades: Presentar interfase para seleccionar grupo archivo plano. Paos 1 y 2.

Excepciones:

Si el estudiante ya estaba registrado en el grupo se presenta un mensaje notificando el hecho.

Precondiciones:

$sesion\_id \in id\_administrador \vee sesion\_id \in tabla\_tutores$

Poscondiciones:

Presenta la interfase para seleccionar grupo y archivo plano.

Nombre: Selección de grupo y archivo. Pasos 1 y 2.

Responsabilidades: Referenciar el grupo y subir al servidor el archivo plano.

Excepciones:

Si se oprime en Salir se pasa a la página de portada.

Si la estructura del archivo plano es incorrecta se notifica el hecho y se vuelve al paso 1 y 2.

Precondiciones:

Haber seleccionado la opción Registrar estudiantes a un grupo.

Poscondiciones:

Se verifica la estructura del archivo plano: (el separador de campo es la coma , ).

$Id\_estudiante, nombres, apellidos, e\_mail \wedge$

$Longitud(id\_estudiante) \leq 12 \wedge$

$Longitud(nombres) \leq 20 \wedge$

$Longitud(apellidos) \leq 20 \wedge$

$Longitud(e\_mail) \leq 40$

Si la estructura del archivo plano es correcta, se eliminan todos los registros de la tabla tmp\_estudiantes y cada registro del archivo plano se adiciona a dicho archivo incluyendo el valor del campo estado para cada uno, donde:

Si (estudiante registrado en el sistema y en el grupo) => estado=0.

Si (estudiante registrado en el sistema pero NO en el grupo) => estado=1.

Si (estudiante sin registrar en el sistema ni en el grupo) => estado=2.

Presentación de interfaz notificando el resultado de la verificación del archivo y con un enlace/botón para ver el listado de los estudiantes.

Nombre: Ver listado de estudiantes. Paso 3.

Responsabilidades: Mostrar el listado de estudiantes del archivo plano.

Excepciones:

Si se oprime en Salir se pasa a la página de portada.

Si la estructura del archivo plano es incorrecta se notifica el hecho y se vuelve al paso 1 y 2.

Si se oprime en Regresar al paso 1 y 2, se retorna a dicha página.

Precondiciones:

Pasos 1 y 2.  
Estructura correcta del archivo plano.

Poscondiciones:

Se presenta el listado con la información en el archivo tmp\_estudiantes, identificando con colores cada uno de los registros de acuerdo al estado, así:

- Si (estado=0) => Rojo
- Si (estado=1) => Amarillo.
- Si (estado=2) => Verde.

Se crea un enlace/botón para confirmar el registro de los estudiantes.

Nombre: Confirmar registro de estudiantes. Paso 4.

Responsabilidades: Registrar la información de cada estudiante de acuerdo al estado de su registro.

Excepciones:

Si la estructura del archivo plano es incorrecta se notifica el hecho y se vuelve al paso 1 y 2.

Si se oprime en Regresar al paso 1 y 2, se retorna a dicha página.

Precondiciones:

Paso 3.

Poscondiciones:

Cada registro del archivo tmp\_estudiantes, se graba así:

- ∀ registro.estado=0 => No hay necesidad de grabarlo en ninguna parte
- ∀ registro.estado=1 => add(id\_estudiante) en notas\_grp[id\_grupo].
- ∀ registro.estado=2 => add(id\_estudiante, nombres, apellidos, e\_mail) en tabla\_estudiantes ^ add(id\_estudiante) en notas\_grp[id\_grupo].

Se presenta un listado con el id de todos los estudiantes del grupo y sus correspondientes notas.

Se crea un enlace/botón ir al paso 1 y 2.

Se crea un enlace/botón para Salir.

Caso de uso: Hacer Práctica.

Actores: Estudiante.

Tipo: Primario.

Descripción: Permitir a un estudiante realizar una práctica en un LIR específico.

Curso normal de eventos.

Administrador	Sistema
0. Selecciona la opción Hacer práctica.	1. Presenta la interfaz ingresar a realizar la práctica.
2. Selecciona el grupo y el archivo y oprime en Subir archivo ó Continuar.	3. Presenta el resultado de la transferencia del archivo y presenta un enlace ó botón para ver el listado de estudiantes (paso 3).

## ANEXO C. DUMP DE LA BASE DE DATOS SALIR.

---

```
# MySQL-Front Dump 2.5
#
# Host: localhost  Database: salir
# -----
# Server version 4.0.15-nt

#
# Table structure for table 'autoreslaboratorios'
#

CREATE TABLE autoreslaboratorios (
  id_autor char(9) NOT NULL default '0',
  nombre char(20) default '0',
  apellido char(20) default '0',
  sitio_trabajo char(30) default '0',
  e_mail char(40) default '0',
  telefono char(20) default '0',
  PRIMARY KEY (id_autor),
  KEY id_autor (id_autor)
) TYPE=MyISAM COMMENT='Autores de Laboratorios';

#
# Table structure for table 'estudiantes'
#

CREATE TABLE estudiantes (
  id_estudiante char(12) NOT NULL default '0',
  id_grupo int(3) unsigned default NULL,
  nombre char(20) default '0',
  apellido char(20) default '0',
  e_mail char(40) default '0',
  PRIMARY KEY (id_estudiante),
  UNIQUE KEY id_estudiante (id_estudiante),
  KEY id_estudiante_2 (id_estudiante)
) TYPE=MyISAM COMMENT='estudiantes inscritos en los grupos de trabajo';

#
# Table structure for table 'grupos'
#

CREATE TABLE grupos (
  id_grupo int(3) unsigned NOT NULL default '0',
  descripcion char(40) default '0',
  contrasenia char(6) NOT NULL default '0',
```

```

documentacion char(20) default NULL,
laboratorio int(3) unsigned NOT NULL default '0',
activo int(10) unsigned default NULL,
PRIMARY KEY (id_grupo),
KEY id_grupo (id_grupo)
) TYPE=MyISAM COMMENT='grupos de trabajo en los laboratorios';

```

```

#
# Table structure for table 'laboratorio'
#

```

```

CREATE TABLE laboratorio (
  id_laboratorio int(3) unsigned NOT NULL default '0',
  ip char(15) default '____.____.____.____',
  puerto int(4) unsigned NOT NULL default '80',
  path char(60) NOT NULL default '',
  nombre char(60) NOT NULL default '0',
  ubicacion char(30) default '0',
  area_aplicacion char(30) default '0',
  duracion_practica int(3) unsigned NOT NULL default '60',
  numero_notas int(2) unsigned NOT NULL default '0',
  documentacion char(20) default '0',
  autoevaluacion int(1) unsigned default NULL,
  PRIMARY KEY (id_laboratorio),
  UNIQUE KEY id_laboratorio (id_laboratorio),
  KEY id_laboratorio_2 (id_laboratorio)
) TYPE=MyISAM COMMENT='registro de todos los laboratorios';

```

```

#
# Table structure for table 'notas_grp301'
#

```

```

CREATE TABLE notas_grp301 (
  id_estudiante char(12) NOT NULL default '',
  nota1 float default NULL,
  PRIMARY KEY (id_estudiante)
) TYPE=MyISAM;

```

```

#
# Table structure for table 'practicas'
#

```

```

CREATE TABLE practicas (
  id_practica int(5) unsigned NOT NULL auto_increment,
  estudiante char(12) NOT NULL default '',
  laboratorio int(3) unsigned NOT NULL default '0',
  fecha date NOT NULL default '0000-00-00',
  hora int(2) unsigned NOT NULL default '0',

```

```

    minuto int(2) unsigned NOT NULL default '0',
    estado int(1) unsigned NOT NULL default '0',
    PRIMARY KEY (id_practica),
    UNIQUE KEY id_practica (id_practica)
) TYPE=MyISAM;

#
# Table structure for table 'rel_autores_laboratorios'
#

CREATE TABLE rel_autores_laboratorios (
  laboratorio int(3) unsigned NOT NULL default '0',
  autor char(9) NOT NULL default ''
) TYPE=MyISAM;

#
# Table structure for table 'rel_grupos_est'
#

CREATE TABLE rel_grupos_est (
  grupo int(3) unsigned NOT NULL default '0',
  estudiante char(12) NOT NULL default '0'
) TYPE=MyISAM;

#
# Table structure for table 'rel_grupos_tutores'
#

CREATE TABLE rel_grupos_tutores (
  grupo int(3) unsigned NOT NULL default '0',
  tutor char(9) NOT NULL default '0'
) TYPE=MyISAM;

#
# Table structure for table 'tmp_estudiantes'
#

CREATE TABLE tmp_estudiantes (
  tmp_id_estudiante char(12) NOT NULL default '0',
  tmp_nombre char(20) NOT NULL default '0',
  tmp_apellido char(20) NOT NULL default '0',
  tmp_e_mail char(40) NOT NULL default '0',
  tmp_estado int(1) unsigned NOT NULL default '0',
  PRIMARY KEY (tmp_id_estudiante),
  UNIQUE KEY tmp_id_estudiante (tmp_id_estudiante),
  KEY tmp_id_estudiante_2 (tmp_id_estudiante)
) TYPE=MyISAM COMMENT='Temporal de estudiantes';

```

```
#  
# Table structure for table 'tutores'  
#  
  
CREATE TABLE tutores (  
  id_tutor char(9) NOT NULL default '0',  
  nombre char(35) default '0',  
  contrasenia char(6) default '0',  
  e_mail char(40) default '0',  
  PRIMARY KEY (id_tutor),  
  KEY id_tutor (id_tutor)  
) TYPE=MyISAM COMMENT='tutores de laboratorios';
```

```
#  
# Table structure for table 'usuarios_admin'  
#  
  
CREATE TABLE usuarios_admin (  
  id int(3) unsigned NOT NULL default '0',  
  nombre_usuario char(12) NOT NULL default '0',  
  password char(10) NOT NULL default 'password',  
  PRIMARY KEY (id)  
) TYPE=MyISAM;
```

## ANEXO D. CODIGO ARCHIVO: SESION\_INICIO\_EST.PHP

---

```
<?php require_once('Connections/mysql.php'); ?>
<?php
// *** Validate request to login to this site.
session_start();

$loginFormAction = $_SERVER['PHP_SELF'];
if (isset($accesscheck)) {
    $GLOBALS['PrevUrl'] = $accesscheck;
    session_register('PrevUrl');
}

if (isset($_POST['box_usuario'])) {
    $loginUsername=$_POST['box_usuario'];
    $password=$_POST['box_clave'];
    $MM_fldUserAuthorization = "";
    $MM_redirectLoginSuccess = "menuppal_est.php";
    $MM_redirectLoginFailed = "error_passwd.php";
    $MM_redirecttoReferrer = false;
    mysql_select_db($database_mysql, $mysql);

    $LoginRS__query=sprintf("SELECT estudiante, contrasenia, id_grupo FROM rel_grupos_est,
grupos WHERE rel_grupos_est.estudiante = '%s' AND rel_grupos_est.grupo = grupos.id_grupo
AND grupos.contrasenia='%s'",
    get_magic_quotes_gpc() ? $loginUsername : addslashes($loginUsername),
    get_magic_quotes_gpc() ? $password : addslashes($password));

    $LoginRS = mysql_query($LoginRS__query, $mysql) or die(mysql_error());
    $loginFoundUser = mysql_num_rows($LoginRS);
    if ($loginFoundUser) {
        $loginStrGroup = "";
        $loginStrGroup = $LoginRS['id_grupo'];

        //declare two session variables and assign them
        $GLOBALS['sesion_id'] = $loginUsername;
        $GLOBALS['sesion_contrasenia'] = $password;
        $GLOBALS['sesion_id_grupo'] = $loginStrGroup;

        //register the session variables
        session_register("sesion_id");
        session_register("sesion_contrasenia");
        session_register("sesion_id_grupo");

        if (isset($_SESSION['PrevUrl']) && false) {
            $MM_redirectLoginSuccess = $_SESSION['PrevUrl'];
        }
        header("Location: " . $MM_redirectLoginSuccess );
    }
    else {
        header("Location: " . $MM_redirectLoginFailed );
    }
}
```

```

}
}
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>Inicio de Sesion de Estudiante</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<style type="text/css">
<!--
body {
    background-image: url(imagenes/Fondo.jpg);
}
.style1 {
    color: #0000CC;
    font-weight: bold;
}
-->
</style></head>

<body>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<div align="center">
    <p><?php echo $_POST['box_est'] ?> </p>
</div>
<form name="inicio_est" method="POST" action="<?php echo $loginFormAction; ?>">
<table width="40%" border="0" align="center">
    <tr>
        <td bgcolor="#99CCFF"><span class="style1">Usuario</span></td>
        <td bgcolor="#99CCFF"><input name="box_usuario" type="text" id="box_usuario"></td>
    </tr>
    <tr>
        <td bgcolor="#99CCFF"><span class="style1">Clave</span></td>
        <td bgcolor="#99CCFF"><input name="box_clave" type="password" id="box_clave"></td>
    </tr>
    <tr>
        <td bgcolor="#99CCFF">&nbsp;</td>
        <td bgcolor="#99CCFF"><input type="submit" name="Submit" value="Ingresar"></td>
    </tr>
</table>
</form>
</body>
</html>

```

## ANEXO E. CODIGO ARCHIVO: HACER\_PRACTICA.PHP (SEGMENTO)

---

A continuación se presenta un segmento del contenido del archivo hacer\_practica.php

```
<?php mysql_select_db($database_mysql, $mysql);
    $usuario = $_SESSION['sesion_id'];

    $query_Recordset1 = "SELECT * FROM practicas, laboratorio WHERE estudiante =
        'usuario' and laboratorio = id_laboratorio";
    $Recordset1 = mysql_query($query_Recordset1, $mysql) or die(mysql_error());
    $row_Recordset1 = mysql_fetch_assoc($Recordset1);
    $totalRows_Recordset1 = mysql_num_rows($Recordset1);

    if ($row_Recordset1['estado'] == 0)
    {
        $estado = "Practica sin comenzar";
    }
    if ($row_Recordset1['estado'] == 1)
    {
        $estado = "Practica inicia y sin terminar";
    }
    if ($row_Recordset1['estado'] == 2)
    {
        $estado = "Practica terminada";
    }
    $dir_ip= "http://".$row_Recordset1['ip'].":".$row_Recordset1['puerto'].
        $row_Recordset1['path']."?usuario=".$usuario."&";

    $id_practica      = $row_Recordset1['id_practica'];
    $estado_codigo    = $row_Recordset1['estado'];
    $practica_archivo = "docs_practica/lab".$row_Recordset1['id_laboratorio']."_".
        $_SESSION['sesion_grupo']."_".$row_Recordset1['estudiante'].
        "_".$row_Recordset1['id_practica'];
    $notas            = "";

    // Calculo del tiempo restante de práctica.
    $formato_fecha = '%d/%m/%Y';
    $formato_hora = '%H:%M:%S';
    $solo_hora = '%H';
    $solo_minuto = '%M';
    $sfecha = strftime($formato_fecha);
    $shorat = strftime($formato_hora);
    $shora = strftime($solo_hora);
    $sminuto= strftime($solo_minu);

    $min_turno = 60 * $row_Recordset1['hora'] + $row_Recordset1['minuto'];
    $min_actual = 60 * strval($shora) + strval($sminuto);
    $tiempo_transcurrido= $min_actual - $min_turno;
    $tiempo = 30 - $tiempo_transcurrido - 3;

    //declare session variables and assign them
    $GLOBALS['sesion_id_practica'] = $id_practica;
    $GLOBALS['sesion_estado'] = $estado_codigo;
```

```

$GLOBALS['sesion_practica_archivo'] = $practica_archivo;
$GLOBALS['sesion_notas'] = $notas;
$GLOBALS['sesion_tiempo'] = $tiempo;

//register the session variables
session_register("sesion_id_practica");
session_register("sesion_estado");
session_register("sesion_practica_archivo");
session_register("sesion_notas");
session_register("sesion_tiempo"); */
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>Pr&aacute;ctica</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<style type="text/css">
<!--
body {
    background-image: url(imagenes/numeros.gif);
}
-->
</style></head>
<body>
<form name="practica" method="post">
    <p><?php echo $_SESSION['session_id']; ?><br>
    Usted tiene turno reservado en este momento para el laboratorio :
    <input name="box_laboratorio" type="text" readonly="true"
        value="<?php echo $row_Recordset1['nombre']; ?>" size="60" maxlength="60"
        width="60" ><br>
    Desde las:
        <?php echo $row_Recordset1['hora'], " : ", $row_Recordset1['minuto']?></p>
    <hr>
    Hora actual : <?php echo $shorat; ?><br>
    Tiempo de pr&aacute;ctica transcurrido : <?php echo $tiempo_transcurrido; ?><br>
    Tiempo de pr&aacute;ctica restante : <?php echo $tiempo; ?>
    <p>Estado de la pr&aacute;ctica:
    <input name="box_estado" type="text" readonly="true" value="
        <?php echo $estado; ?>" ></p>
    <p> <a href="<?php echo $dir_ip;?>" target="_blank">Para ingresar haga click aqui
    </a></p>
    <p>&nbsp;</p>
    <p>&nbsp;</p>
</form>
</body>
</html>
<?php
mysql_free_result($Recordset1);
?>

```

## ANEXO F. CODIGO ARCHIVO: INICIO.PHP

---

```
<?php
session_start();
$usuario = $_SESSION['sesion_id'];
$id_practica = $_SESSION['sesion_id_practica'];
$estado_codigo = $_SESSION['sesion_estado'];
$practica_archivo = $_SESSION['sesion_practica_archivo'];
$notas = $_SESSION['sesion_notas'];
$tiempo = $_SESSION['sesion_tiempo'];

if ($estado_codigo == "2") {
?>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN"
"http://www.w3.org/TR/html4/frameset.dtd">
<html>
<head>
<title>LABORATORIO INTERACTIVO REMOTO</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
<frameset rows="120,*" cols="*" framespacing="0" frameborder="NO" border="0">
  <frame src="inicio2.php" name="titulo" scrolling="NO">
  <frame src="marcoPpal.php" name="principal" id="principal" scrolling="AUTO">
</frameset>
<noframes><body>
</body></noframes>
</html>
<?php
}
else { ?>
<html>
<body>
ACCESO NO PERMITIDO
</body>
</html>

<?
}
?>
```

## ANEXO G. CODIGO ARCHIVO: REGISTRO\_NOTAS.PHP

---

```
<?php
session_start();

$usuario = $_SESSION['sesion_id'];
$id_practica = $_SESSION['sesion_id_practica'];
$estado_codigo = $_SESSION['sesion_estado'];
$practica_archivo = $_SESSION['sesion_practica_archivo'];
$notas = $_SESSION['sesion_notas'];
$tiempo = $_SESSION['sesion_tiempo'];

function registro_notas($nt) {
    $_SESSION['sesion_notas'] = $nt;
}

if (isset($_POST['regNotas'])) {
    registro_notas($_POST['regNotas']);
    ?>
    <html>
    <body>
        LAS NOTAS HAN SIDO REGISTRADAS...
    <script language="javascript">
        Location.href = "inicio.php";
    </script>
    </body>
    </html>
    <?
}
```

## ANEXO H. PRUEBAS DE RENDIMIENTO

---

### Conexión cliente – sistema SALIR

Protocolo http

Escenario 1. Red Lan de la Universidad Autónoma de Manizales

Servidor: APACHE 2.0.32 con soporte PHP

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	x		http	1	45
8	x		http	1	45
9	x		http	1	46
10	x		http	1	46
11	x		http	1	46
12	x		http	1	45
1		x	http	1	45
2		x	http	1	45
3		x	http	1	47
4		x	http	1	47
5		x	http	1	46
6		x	http	1	47
7		x	http	1	47
8		x	http	1	47
9		x	http	1	46
10		x	http	1	45

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	5	45
8	X		http	5	45
9	X		http	5	46
10	X		http	5	46
11	X		http	5	46
12	X		http	5	45
1		x	http	5	45
2		x	http	5	45
3		x	http	5	47
4		x	http	5	47
5		x	http	5	46
6		x	http	5	47

7		x	http	5	47
8		x	http	5	47
9		x	http	5	46
10		x	http	5	45

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	10	45
8	X		http	10	45
9	X		http	10	46
10	X		http	10	46
11	X		http	10	46
12	X		http	10	45
1		x	http	10	46
2		x	http	10	46
3		x	http	10	47
4		x	http	10	47
5		x	http	10	47
6		x	http	10	47
7		x	http	10	47
8		x	http	10	47
9		x	http	10	46
10		x	http	10	45

Servidor: JAKARTA TOMCAT 5.0

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	1	55
8	X		http	1	56
9	X		http	1	56
10	X		http	1	57
11	X		http	1	57
12	X		http	1	58
1		X	http	1	58
2		X	http	1	58
3		X	http	1	60
4		X	http	1	60
5		X	http	1	60
6		X	http	1	59
7		X	http	1	58
8		X	http	1	58
9		X	http	1	57
10		X	http	1	56

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	5	55
8	X		http	5	57
9	X		http	5	57
10	X		http	5	57
11	X		http	5	57
12	X		http	5	58
1		x	http	5	58
2		x	http	5	59
3		x	http	5	60
4		x	http	5	61
5		x	http	5	61
6		x	http	5	60
7		x	http	5	59
8		x	http	5	58
9		x	http	5	58
10		x	http	5	56

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	10	56
8	X		http	10	56
9	X		http	10	56
10	X		http	10	58
11	X		http	10	58
12	X		http	10	57
1		x	http	10	58
2		x	http	10	58
3		x	http	10	61
4		x	http	10	61
5		x	http	10	61
6		x	http	10	62
7		x	http	10	61
8		x	http	10	59
9		x	http	10	57
10		x	http	10	57

Escenario 2. Conexión desde España. Banda ancha 1Mbps.

Servidor Apache 2.0.32 con PHP

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	1	820
8	X		http	1	850
9	X		http	1	900
10	X		http	1	933
11	X		http	1	920
12	X		http	1	933
1		x	http	1	910
2		x	http	1	940
3		x	http	1	942
4		x	http	1	933
5		x	http	1	916
6		x	http	1	850
7		x	http	1	826
8		x	http	1	823
9		x	http	1	825
10		x	http	1	830

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	3	821
8	X		http	3	850
9	X		http	3	910
10	X		http	3	940
11	X		http	3	936
12	X		http	3	933
1		x	http	3	920
2		x	http	3	950
3		x	http	3	962
4		x	http	3	951
5		x	http	3	928
6		x	http	3	868
7		x	http	3	832
8		x	http	3	816
9		x	http	3	810
10		x	http	3	830

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	5	810
8	X		http	5	880
9	X		http	5	828
10	X		http	5	930
11	X		http	5	958
12	X		http	5	963
1		x	http	5	975
2		x	http	5	972
3		x	http	5	977
4		x	http	5	963
5		x	http	5	940
6		x	http	5	916
7		x	http	5	905
8		x	http	5	894
9		x	http	5	848
10		x	http	5	820

Servidor JAKARTA TOMCAT 5.0

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	1	854
8	X		http	1	852
9	X		http	1	926
10	X		http	1	945
11	X		http	1	928
12	X		http	1	959
1		x	http	1	936
2		x	http	1	982
3		x	http	1	975
4		x	http	1	973
5		x	http	1	930
6		x	http	1	888
7		x	http	1	854
8		x	http	1	846
9		x	http	1	842
10		x	http	1	840

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	3	861

8	X		http	3	867
9	X		http	3	959
10	X		http	3	969
11	X		http	3	969
12	X		http	3	956
1		x	http	3	938
2		x	http	3	971
3		x	http	3	1000
4		x	http	3	953
5		x	http	3	939
6		x	http	3	916
7		x	http	3	860
8		x	http	3	846
9		x	http	3	824
10		x	http	3	879

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	5	822
8	X		http	5	925
9	X		http	5	857
10	X		http	5	954
11	X		http	5	958
12	X		http	5	989
1		x	http	5	1006
2		x	http	5	1016
3		x	http	5	1014
4		x	http	5	988
5		x	http	5	975
6		x	http	5	941
7		x	http	5	924
8		x	http	5	912
9		x	http	5	866
10		x	http	5	834

Escenario 3. Conexión telefónica. MODEM 33.6

Servidor Apache 2.0.32 con PHP

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	1	940
8	X		http	1	945
9	X		http	1	1001
10	X		http	1	1023

11	X		http	1	1016
12	X		http	1	1025
1		x	http	1	1020
2		x	http	1	980
3		x	http	1	962
4		x	http	1	954
5		x	http	1	958
6		x	http	1	990
7		x	http	1	962
8		x	http	1	945
9		x	http	1	916
10		x	http	1	920

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	3	936
8	X		http	3	950
9	X		http	3	990
10	X		http	3	1010
11	X		http	3	1025
12	X		http	3	1025
1		x	http	3	990
2		x	http	3	1005
3		x	http	3	1015
4		x	http	3	1000
5		x	http	3	980
6		x	http	3	984
7		x	http	3	982
8		x	http	3	965
9		x	http	3	950
10		x	http	3	948

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	5	956
8	X		http	5	962
9	X		http	5	970
10	X		http	5	981
11	X		http	5	1020
12	X		http	5	1035
1		x	http	5	1025
2		x	http	5	1023
3		x	http	5	1012
4		x	http	5	1015
5		x	http	5	1012

6		x	http	5	990
7		x	http	5	985
8		x	http	5	975
9		x	http	5	970
10		x	http	5	965

Servidor JAKARTA TOMCAT 5.0

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	1	959
8	X		http	1	977
9	X		http	1	1027
10	X		http	1	1062
11	X		http	1	1050
12	X		http	1	1064
1		x	http	1	1037
2		x	http	1	1016
3		x	http	1	978
4		x	http	1	966
5		x	http	1	969
6		x	http	1	1007
7		x	http	1	991
8		x	http	1	993
9		x	http	1	931
10		x	http	1	966

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	3	968
8	X		http	3	975
9	X		http	3	1017
10	X		http	3	1054
11	X		http	3	1073
12	X		http	3	1050
1		x	http	3	995
2		x	http	3	1053
3		x	http	3	1053
4		x	http	3	1016
5		x	http	3	1004
6		x	http	3	1001
7		x	http	3	1009
8		x	http	3	998
9		x	http	3	982
10		x	http	3	970

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	5	993
8	X		http	5	986
9	X		http	5	1010
10	X		http	5	984
11	X		http	5	1026
12	X		http	5	1071
1		x	http	5	1050
2		x	http	5	1055
3		x	http	5	1029
4		x	http	5	1045
5		x	http	5	1048
6		x	http	5	1025
7		x	http	5	1031
8		x	http	5	990
9		x	http	5	982
10		x	http	5	972

Comunicación – Cliente – Laboratorio LIR

Protocolo http.

Escenario 1. Red Lan de la Universidad Autónoma de Manizales

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	1	45
8	X		http	1	45
9	X		http	1	46
10	X		http	1	45
11	X		http	1	45
12	X		http	1	45
1		x	http	1	45
2		x	http	1	45
3		x	http	1	46
4		x	http	1	46
5		x	http	1	46
6		x	http	1	47
7		x	http	1	47
8		x	http	1	47
9		x	http	1	46
10		x	http	1	45

Escenario 2. Conexión desde España. Banda Ancha 1Mbps.

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	1	2680
8	X		http	1	2685
9	X		http	1	2690
10	X		http	1	2710
11	X		http	1	2720
12	X		http	1	2710
1		x	http	1	2715
2		x	http	1	2719
3		x	http	1	2730
4		x	http	1	2725
5		x	http	1	2740
6		x	http	1	2735
7		x	http	1	2728
8		x	http	1	2690
9		x	http	1	2685
10		x	http	1	2675

Escenario 3. Conexión telefónica. MODEM 33.6

HORA DIA			PROTOCOLO	Conexiones Simultáneas	Tiempo respuesta (mSeg)
Hora	AM	PM			
7	X		http	1	3560
8	X		http	1	3550
9	X		http	1	3580
10	X		http	1	3590
11	X		http	1	3585
12	X		http	1	3610
1		x	http	1	3680
2		x	http	1	3670
3		x	http	1	3680
4		x	http	1	3690
5		x	http	1	3700
6		x	http	1	3670
7		x	http	1	3660
8		x	http	1	3580
9		x	http	1	3550
10		x	http	1	3550

Transmisión de broadcast de video (wmv)

Escenario 1. Universidad Autónoma de Manizales

HORA DIA			PROTOCOLO	Conexiones simultáneas	Tiempo respuesta
Hora	AM	PM			
7	X		video broadcast tcp	1	12339
8	X		video broadcast tcp	1	12960
9	X		video broadcast tcp	1	13991
10	X		video broadcast tcp	1	13625
11	X		video broadcast tcp	1	15100
12	X		video broadcast tcp	1	13618
1		x	video broadcast tcp	1	12763
2		x	video broadcast tcp	1	14418
3		x	video broadcast tcp	1	12780
4		x	video broadcast tcp	1	12975
5		x	video broadcast tcp	1	13521
6		x	video broadcast tcp	1	12822
7		x	video broadcast tcp	1	14047
8		x	video broadcast tcp	1	16227
9		x	video broadcast tcp	1	15051
10		x	video broadcast tcp	1	16735

HORA DIA			PROTOCOLO	Conexiones simultáneas	Tiempo respuesta
Hora	AM	PM			
7	X		video broadcast tcp	5	14645
8	X		video broadcast tcp	5	13843
9	X		video broadcast tcp	5	16122
10	X		video broadcast tcp	5	14331
11	X		video broadcast tcp	5	14401
12	X		video broadcast tcp	5	16563
1		x	video broadcast tcp	5	12485
2		x	video broadcast tcp	5	13908
3		x	video broadcast tcp	5	12864
4		x	video broadcast tcp	5	16187
5		x	video broadcast tcp	5	13688
6		x	video broadcast tcp	5	16537
7		x	video broadcast tcp	5	16894
8		x	video broadcast tcp	5	15148
9		x	video broadcast tcp	5	14272
10		x	video broadcast tcp	5	16747

HORA DIA			PROTOCOLO	Conexiones simultáneas	Tiempo respuesta
Hora	AM	PM			
7	X		video broadcast tcp	10	15037
8	X		video broadcast tcp	10	16019
9	X		video broadcast tcp	10	13158
10	X		video broadcast tcp	10	15767
11	X		video broadcast tcp	10	14290
12	X		video broadcast tcp	10	14396

1	x	video broadcast tcp	10	14330
2	x	video broadcast tcp	10	12159
3	x	video broadcast tcp	10	15558
4	x	video broadcast tcp	10	15594
5	x	video broadcast tcp	10	16024
6	x	video broadcast tcp	10	12817
7	x	video broadcast tcp	10	16399
8	x	video broadcast tcp	10	14384
9	x	video broadcast tcp	10	16668
10	x	video broadcast tcp	10	15568

Escenario 2. Conexión desde España. Banda Ancha 1Mbps.

Rata de transmisión. 300 Kbps

HORA DIA			PROTOCOLO	Conexiones simultáneas	Tiempo respuesta
Hora	AM	PM			
7	X		video broadcast tcp	1	13340
8	X		video broadcast tcp	1	13050
9	X		video broadcast tcp	1	14953
10	X		video broadcast tcp	1	13041
11	X		video broadcast tcp	1	13964
12	X		video broadcast tcp	1	13715
1		x	video broadcast tcp	1	14202
2		x	video broadcast tcp	1	16911
3		x	video broadcast tcp	1	12312
4		x	video broadcast tcp	1	16787
5		x	video broadcast tcp	1	14272
6		x	video broadcast tcp	1	14561
7		x	video broadcast tcp	1	13939
8		x	video broadcast tcp	1	16587
9		x	video broadcast tcp	1	15344
10		x	video broadcast tcp	1	12134

Rata de transmisión. 128 Kbps

HORA DIA			PROTOCOLO	Conexiones simultáneas	Tiempo respuesta
Hora	AM	PM			
7	X		video broadcast tcp	5	12608
8	X		video broadcast tcp	5	14400
9	X		video broadcast tcp	5	13030
10	X		video broadcast tcp	5	16639
11	X		video broadcast tcp	5	15061
12	X		video broadcast tcp	5	16051
1		x	video broadcast tcp	5	15988
2		x	video broadcast tcp	5	13714
3		x	video broadcast tcp	5	13414

4		x	video broadcast tcp	5	12455
5		x	video broadcast tcp	5	12020
6		x	video broadcast tcp	5	12834
7		x	video broadcast tcp	5	12803
8		x	video broadcast tcp	5	15482
9		x	video broadcast tcp	5	13923
10		x	video broadcast tcp	5	12623

Rata de transmisión. 32 Kbps

HORA DIA			PROTOCOLO	Conexiones simultáneas	Tiempo respuesta
Hora	AM	PM			
7	X		video broadcast tcp	10	16765
8	X		video broadcast tcp	10	15000
9	X		video broadcast tcp	10	15056
10	X		video broadcast tcp	10	13588
11	X		video broadcast tcp	10	16081
12	X		video broadcast tcp	10	16259
1		x	video broadcast tcp	10	13646
2		x	video broadcast tcp	10	14026
3		x	video broadcast tcp	10	12562
4		x	video broadcast tcp	10	12605
5		x	video broadcast tcp	10	12147
6		x	video broadcast tcp	10	16245
7		x	video broadcast tcp	10	16264
8		x	video broadcast tcp	10	16175
9		x	video broadcast tcp	10	12506
10		x	video broadcast tcp	10	16765

Escenario 3. Conexión telefónica. MODEM a 33.6 Kbps

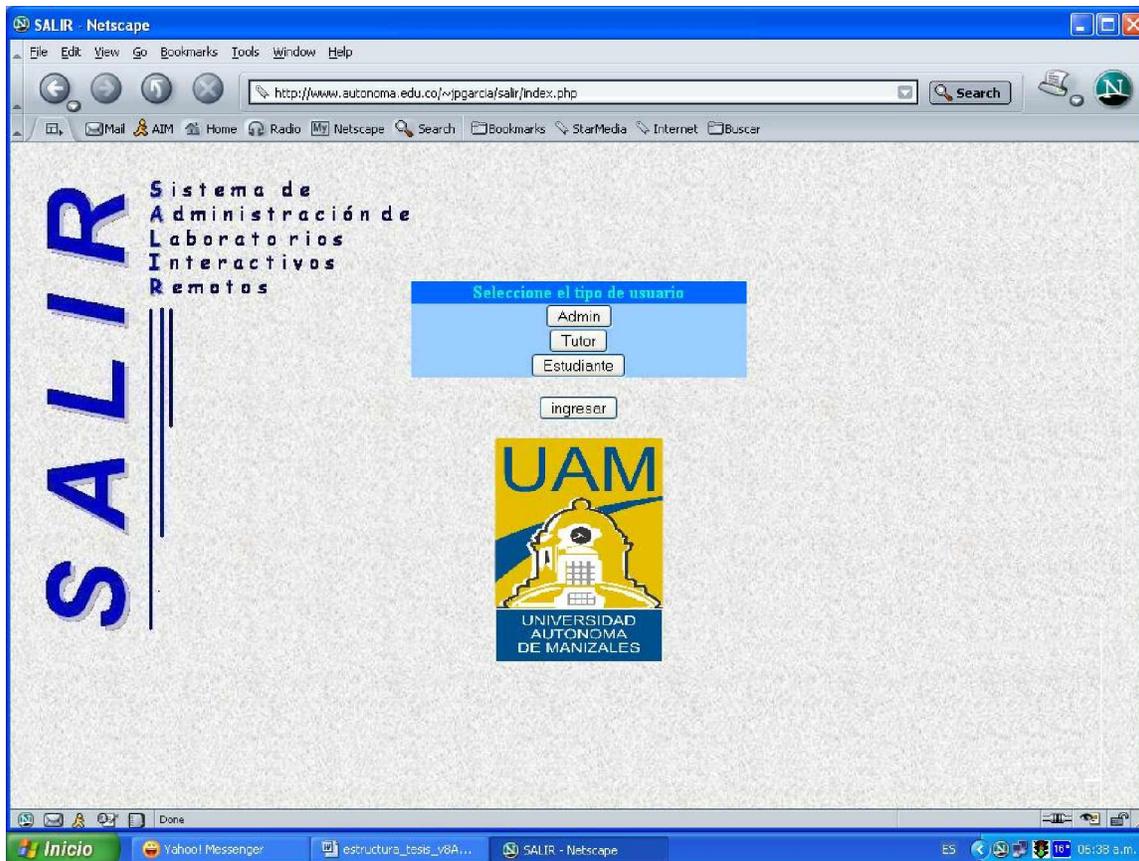
Rata de transmisión. 32 Kbps

HORA DIA			PROTOCOLO	Conexiones simultáneas	Tiempo respuesta
Hora	AM	PM			
7	X		video broadcast tcp	1	15352
8	X		video broadcast tcp	1	15117
9	X		video broadcast tcp	1	16449
10	X		video broadcast tcp	1	16467
11	X		video broadcast tcp	1	15389
12	X		video broadcast tcp	1	12472
1		x	video broadcast tcp	1	15707
2		x	video broadcast tcp	1	16460
3		x	video broadcast tcp	1	12158
4		x	video broadcast tcp	1	15438
5		x	video broadcast tcp	1	16517
6		x	video broadcast tcp	1	15274

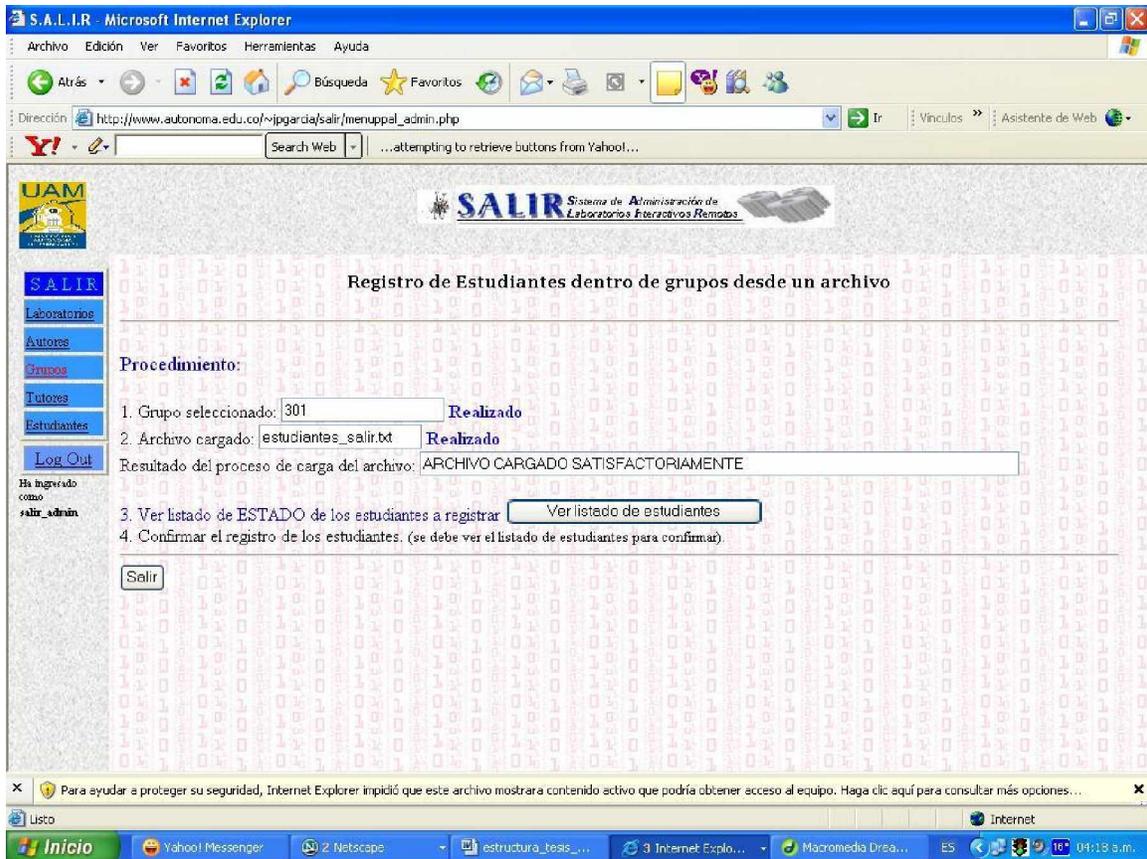
7		x	video broadcast tcp	1	12843
8		x	video broadcast tcp	1	14258
9		x	video broadcast tcp	1	12765
10		x	video broadcast tcp	1	16226

## ANEXO I. INTERFASES DEL SISTEMA DE ADMINISTRACION SALIR

---



Ingreso al sistema SALIR.



Registro de estudiantes mediante un archivo plano 1. (Usuario administrador)

S.A.L.I.R - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Atrás Búsqueda Favoritos

Dirección http://www.autonoma.edu.co/~vipgarcia/salir/menuppal\_admin.php

UAM SALIR Sistema de Administración de Laboratorios Interactivos Remotos

0 Registrado en el sistema y en el grupo  
 1 Registrado en el sistema pero NO en el grupo  
 2 Sin registrar en el sistema ni en el grupo

Total de Estudiantes en el archivo : 7

id_estudiante	Nombre	Apellido	e-mail	Estado
20041001	JuanFelipe	García	jfelipe@uam.edu.co	0
20041002	JuanDavid	García	jdavid@uam.edu.co	0
20041003	Sara	García	saga@uam.edu.co	1
20051001	Ale	Rojas	dale@uam.edu.co	0
20051004	Jose	Jaramillo	jejara@uam.edu.co	2
20051005	Pedro	Gonzales	pegonz@uam.edu.co	2
20051006	Ruben	Alvarez	rubenal@uam.edu.co	1

Paso 4 y último. Confirmar el registro de los estudiantes.

Regresar al Paso 1 y 2. Definir grupo y archivo

Salir

Para ayudar a proteger su seguridad, Internet Explorer impidió que este archivo mostrara contenido activo que podría obtener acceso al equipo. Haga clic aquí para consultar más opciones...

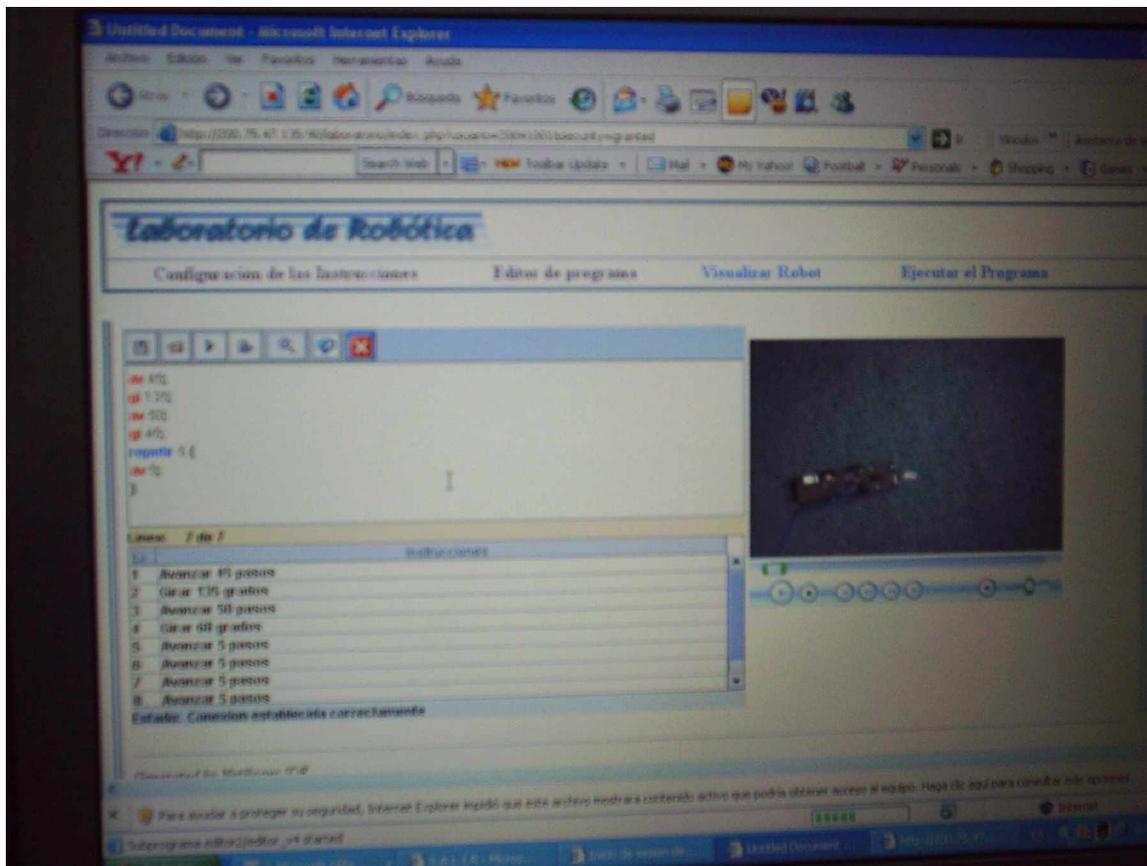
Inicio Yahoo! Mess... 2 Netscape estructura\_1... 3 Internet... Macromedia... estudiantes... E5 04:31 a.m.

Registro de estudiantes mediante un archivo plano 2. (Usuario administrador)

Registro de estudiantes mediante un archivo plano 3. (Usuario administrador)



Enlace para realizar práctica. (Usuario estudiante)



Ejecución de una práctica en el LIR Control de Dispositivos Robóticas. (Usuario estudiante)



Fin de práctica. (Usuario estudiante)

## ANEXO J. INTERFASES Y DESCRIPCIÓN DEL FUNCIONAMIENTO DEL LABORATORIO INTERACTIVO REMOTO

---

En el presente anexo, se muestran los componentes del Laboratorio Interactivo Remoto y su funcionamiento.

Como vimos anteriormente, en la Figura 19. Diagrama de un LIR, los componentes se pueden describir de acuerdo al elemento en cuestión, ya sea del cliente, del servidor LIR o del dispositivo robótico.

En resumen, los tres grandes pilares para este laboratorio comprenden, el Cliente Web, el Servidor del LIR y el Dispositivo Remoto.

Para cada uno de estos pilares, es necesario utilizar aplicativos específicos para desarrollar las tareas indicadas para lograr los procesos sugeridos, es decir, del lado del cliente, es necesario cumplir ciertas condiciones de diseño para lograr el objetivo y del lado del servidor, es necesario contar con los requerimientos de hardware apropiados.

Cliente Web.

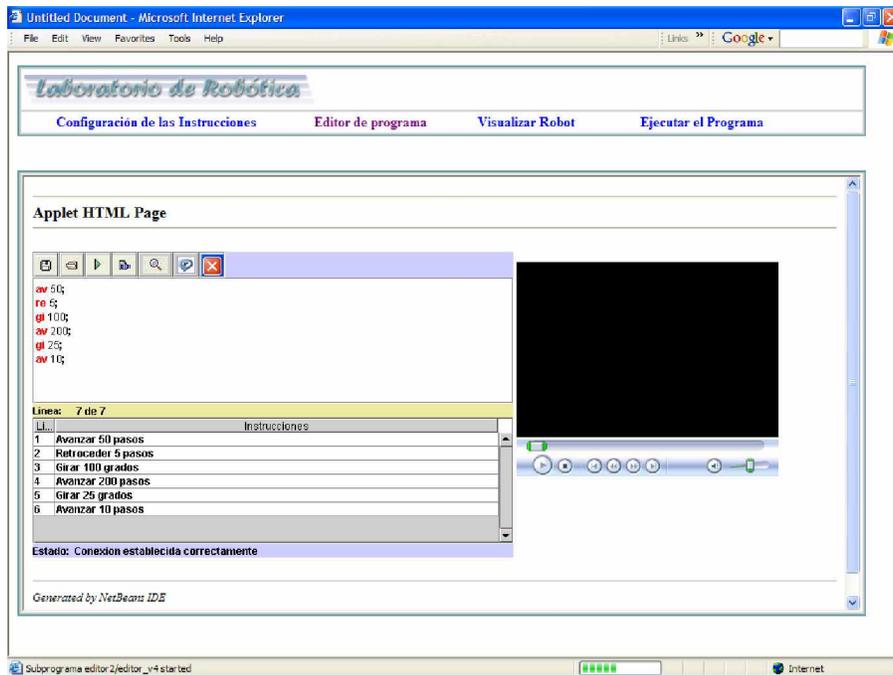
### Requerimientos

Nuestro cliente Web, es una estación portátil con un procesador AMD a 1.8 Ghz, 256 Mb de ram y un disco duro de 40 Gb, tarjeta de Red y Módem integrados. El sistema operativo es Windows XP Home edition y su navegador web principal es el Internet Explorer 6. Es requisito fundamental, que el cliente cuente con el plug-in de java con el runtime (JRE – Java Runtime Environment) más reciente que sea posible.

El JRE, es utilizado para ejecutar el Applet albergado en el servidor, encargado de realizar la compilación del lenguaje que el estudiante debe crear para poder controlar el dispositivo robótico remoto.

Además de las condiciones anteriores, es necesario modificar las políticas de seguridad administradas para garantizar una ejecución del applet transparente o si es posible, firmar el applet y que la confianza en el mismo de acceso a las zonas prohibidas.

La siguiente figura, muestra la interfase Web vista por el cliente del laboratorio.



Interfase del Editor del Laboratorio Interactivo Remoto

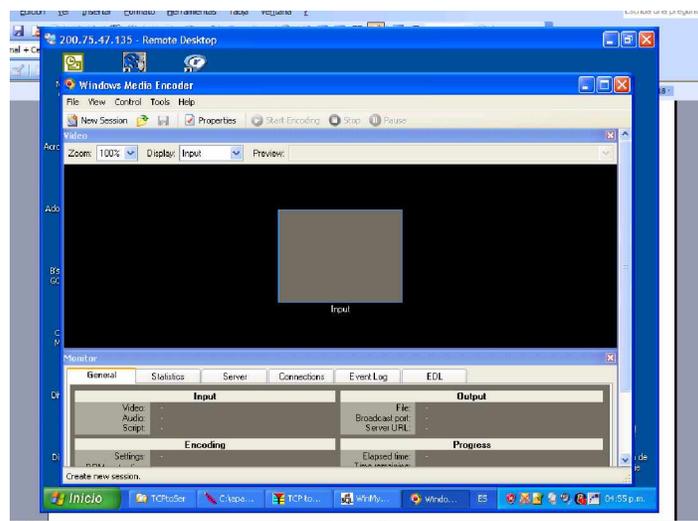
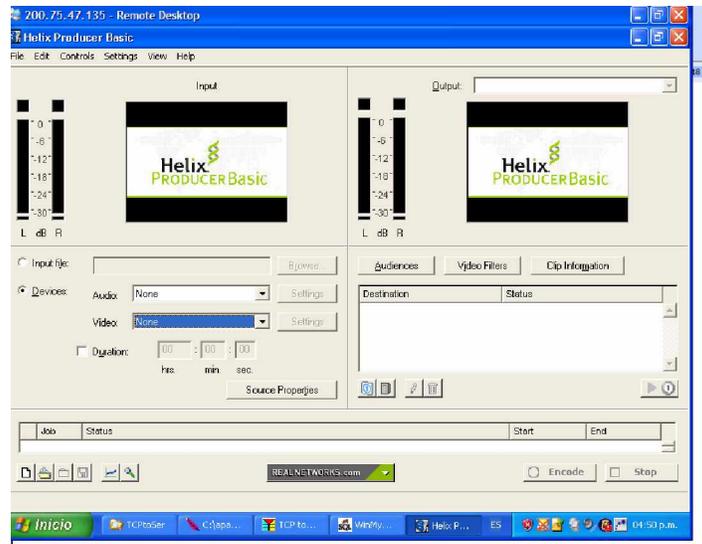
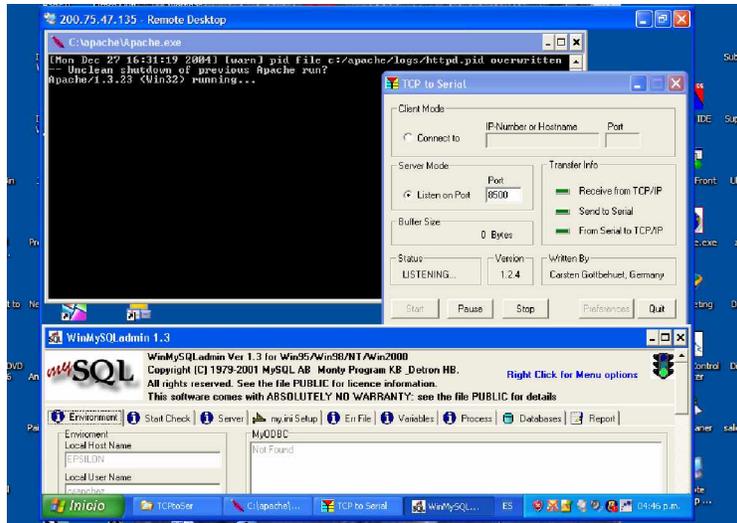
Servidor del LIR

### Requerimientos

El servidor que contiene los aplicativos del LIR y las interfases de hardware, tiene las siguientes características: Procesador Pentium 3 a 1.0 Ghz, 256 Mb de Ram, disco duro de 40 Gb y sistema operativo Windows XP Pro.

Las herramientas de software necesarias para el funcionamiento del LIR son: Servidor Web Apache (Soporte PHP), Interfase TCP / RS232, MySQL y Windows Media Encoder o Helix Producer, en este caso se tienen ambos.

Las siguientes pantallas muestran dichos programas en ejecución.

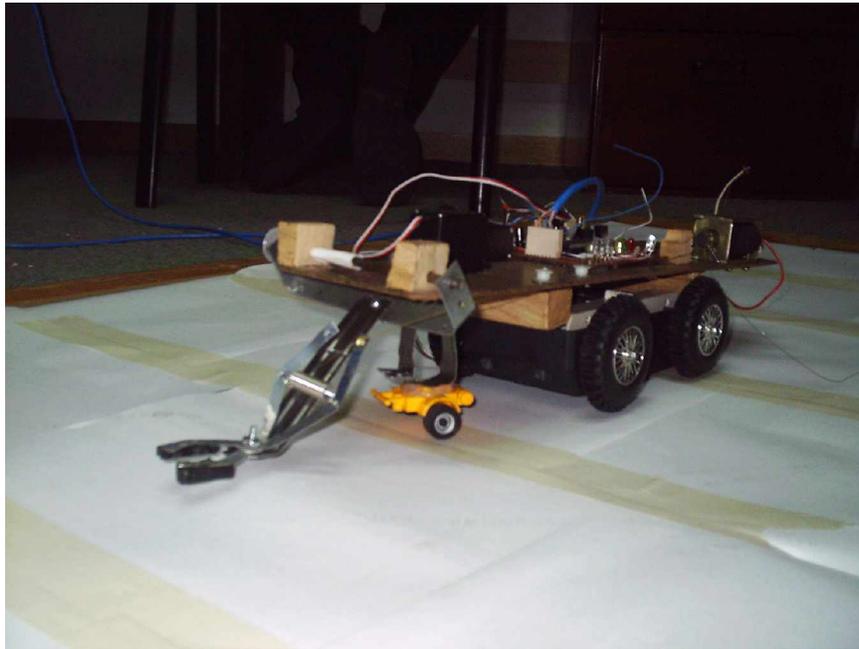


Pantallas de los programas en ejecución en el servidor del LIR

## Dispositivo Robótico

El dispositivo robótico es un móvil que está fabricado con materiales comunes, como piezas de madera y metal moldeadas para formar los soportes. El dispositivo de tracción es un motor paso a paso de 4 devanados. El sistema de dirección y de elevación del dispositivo recolector está conformado por servomotores de uso en aeromodelismo y el sistema de activación de la pinza está elaborado con un solenoide de gran poder.

La siguiente figura muestra una vista del móvil.



La lista de componentes del móvil para su construcción es:

Componente	Cantidad	Descripción
Motor Paso a Paso	1	Motor de 4 devanados
Servo Motor Aeromodelismo	2	Servos marca Futaba
Solenoide	1	Dispositivo electromecánico
Microcontrolador PIC 16F873	1	Micro con capacidad de comunicación RS232 y 20 puertos

En los Anexos L y M, se muestran el análisis y el diseño del dispositivo robótico.

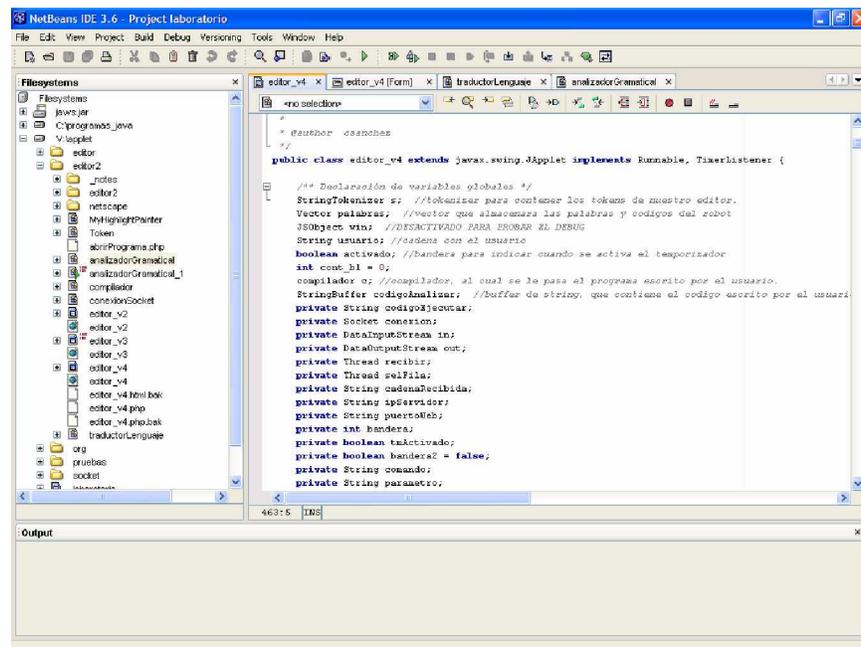


## ANEXO K. APLICATIVO JAVA (APPLET) DEL LIR (COMPILADOR E INTERPRETE DE COMANDOS)

El applet Java encargado de ejecutar en la máquina cliente la interfase que permite la compilación de las instrucciones de telecontrol, está compuesto de los siguientes elementos:

- Clase Editor
- Clase Analizador Gramatical
- Clase Traductor Lenguaje

La clase editor es la encargada de la parte gráfica y de reunir los demás componentes en el aplicativo final.



La clase Analizador Gramatical, es la encargada de hacer un análisis sintáctico de las instrucciones de acuerdo al lenguaje definido por el usuario. Finalmente, la clase Traductor Lenguaje es quien se encarga de pasar los componentes del lenguaje definido al lenguaje tipo SCPI propietario que reconoce el dispositivo robótico para ejecutar las instrucciones según los parámetros especificados.

## Listado del código fuente de las clases del Applet.

### CLASE EDITOR

```
* editor_v4.java
*
* Created on 4 de agosto de 2004, 02:18 PM
*/

package editor2;

import javax.swing.text.*;
import java.awt.Color;
import java.awt.Font;
import javax.swing.table.*;
import javax.swing.ListSelectionModel;
import java.awt.Point;
import java.util.*;
import netscape.javascript.*;
import org.netbeans.examples.lib.timerbean.*;
import java.net.Socket;
import java.io.*;
import java.util.Vector;

/**
 *
 * @author csanchez
 */
public class editor_v4 extends javax.swing.JApplet implements Runnable, TimerListener {

    /** Declaración de variables globales */
    StringTokenizer s; //tokenizer para contener los tokens de nuestro editor.
    Vector palabras; //vector que almacenara las palabras y codigos del robot
    JObject win; //DESACTIVADO PARA PROBAR EL DEBUG
    String usuario; //cadena con el usuario
    boolean activado; //bandera para indicar cuando se activa el temporizador
    int cont_b1 = 0;
    compilador c; //compilador, al cual se le pasa el programa escrito por el usuario.
    StringBuffer codigoAnalizar; //buffer de string, que contiene el código escrito por el usuario, después de darle un formato apropiado.
    private String codigoEjecutar;
    private Socket conexion;
    private DataInputStream in;
    private DataOutputStream out;
    private Thread recibir;
    private Thread selfila;
    private String cadenaRecibida;
    private String ipServidor;
    private String puertoWeb;
    private int bandera;
    private boolean tmActivado;
    private boolean bandera2 = false;
    private String comando;
    private String parametro;
    private boolean ack;
    private String mensaje;
    private DefaultTableModel model;
    private int filaSeleccionada;
    private ListSelectionModel selectionModel;
    Vector posEnter;
    Integer posCursor;
    int filaActual;
    int totalFilas;
    /** Este método inicializa el applet... */
```

```

public void init() {
    filaActual = totalFilas = 1;
    posEnter = new Vector();
    initComponents();
    cargarParametros();
    cargarEstilos();
    configurarTabla();
    c = new compilador(palabras);
    codigoAnalizar = new StringBuffer();
    //TIMERS
    timer_recibir.addTimerListener(this);
    timer_recibir.start();
    recibir = new Thread(this,"recibir");
    selfila = new Thread(this,"selfila");
    tmActivado = false;
    try {
        conexion = new Socket("200.75.47.135",8500);
        in = new DataInputStream(conexion.getInputStream());
        out = new DataOutputStream(conexion.getOutputStream());
        recibir.start();
        if (conexion.isConnected())
            StatusBar.setText("Estado: Conexion establecida correctamente");
        else {
            StatusBar.setText("Ocurrió un error en la conexión con el servidor");
            //botonCorrerPrograma.setEnabled(false);
            //botonCorrerPrograma.setDisabledIcon();
        }
    }

    catch (IOException io) {
        StatusBar.setText("Estado: Error al intentarse conectar con el servidor");
        botonCorrerPrograma.setEnabled(false);
    }
}

/** Método cargarEstilos()
 * Este método es utilizado para cargar los estilos gráficos del editor
 **/

private void configurarTabla() {
    model = new DefaultTableModel();
    model.addColumn("Linea");
    model.addColumn("Instrucciones");
    tabla1.setModel(model);
    TableColumn column = tabla1.getColumnModel().getColumn(0);
    column.setMaxWidth(25);
    column.setPreferredWidth(25);
    column.setWidth(25);
    model.setNumRows(4);
    tabla1.setFont(new Font("arial",Font.BOLD,12));
    selectionModel = tabla1.getSelectionModel();
}

public void cargarEstilos() {
    StyledDocument doc = editor.getStyledDocument(); //objeto de tipo documento para aplicar el formato de color
    win = (JObject) JObject.getWindow(this);

    // Texto rojo en negrilla
    Style style = editor.addStyle("rojo_n", null);
    StyleConstants.setForeground(style, Color.red);
    StyleConstants.setBold(style, true);

    // Texto rojo
    Style style2 = editor.addStyle("rojo", null);
    StyleConstants.setForeground(style2, Color.red);

    // Texto azul en negrilla
    Style style3 = editor.addStyle("azul_n", null);

```

```

StyleConstants.setForeground(style3, Color.blue);
StyleConstants.setBold(style3, true);

// Texto azul
Style style4 = editor.addStyle("azul", null);
StyleConstants.setForeground(style4, Color.blue);
StyleConstants.setBold(style4, true);

// Texto negro en negrilla
Style style5 = editor.addStyle("negro_n", null);
StyleConstants.setForeground(style5, Color.black);
//StyleConstants.setFont()
StyleConstants.setBold(style5, true);
}

/** Método cargarParametros()
 * Utilizado para inicializar los parámetros individuales, de los que depende el editor,
 * como las palabras del lenguaje definido por el usuario y el nombre de usuario
 */

public void cargarParametros() {
    String keywords = getParameter("keywords"); //Parametro que contiene la cadena con palabras y comandos para mover el robot.
    //String keywords = "1&go,2&back,3&turn,4&up,5&close";
    this.ipServidor = getParameter("ipServidor");
    this.puertoWeb = getParameter("puertoWeb");
    s = new StringTokenizer(keywords, ",");
    palabras = new Vector(s.countTokens());
    while (s.hasMoreTokens()) {
        String elementos[] = s.nextToken().split("&"); //separación en un arreglo de la cadena con palabras definidas por el usuario y codigos de comando del robot.
        palabras.addElement(elementos);
    } // Aqui se adicionan todas las palabras y codigos de comando al respectivo vector, para un mejor manejo posterior.
    usuario = getParameter("usuario");
}

/** Método highlight, que recibe el patron para colorear y el estilo respectivo
 * @param String patron Palabra que debe ser coloreada con el estilo respectivo en el editor
 * @param String estilo Estilo utilizado para colorear la palabra
 */
private void highlight(String patron, String estilo) {
    StyledDocument doc = editor.getStyledDocument(); //objeto de tipo documento, que permite aplicar color
    String text = "";
    try {
        text = doc.getText(0, doc.getLength());
    } catch (BadLocationException be) {}
    int pos = 0;
    // Búsqueda del patron en el texto
    while ((pos = text.indexOf(patron, pos)) >= 0) {
        // Crea el coloreador usando un pintor privado y lo aplica alrededor del patron
        doc.setCharacterAttributes(pos, patron.length(), editor.getStyle(estilo), true);
        pos += patron.length();
    }
}

/** Este método es llamado dentro del inicializador init() que
 * inicializa la ventana.
 * Este contenido es generado automáticamente por el editor de NETBEANS.
 */
}

private void initComponents() {
    JFrame1 = new javax.swing.JFrame();
    nomArchivo = new javax.swing.JTextField();
    jLabel1 = new javax.swing.JLabel();
    jButton2 = new javax.swing.JButton();
    timer_recibir = new org.netbeans.examples.lib.timerbean.Timer();
    jPanel1 = new javax.swing.JPanel();
    jButton1 = new javax.swing.JButton();
    jButton3 = new javax.swing.JButton();
}

```

```

botonCompilar = new javax.swing.JButton();
botonCorrerPrograma = new javax.swing.JButton();
botonVerLenguaje = new javax.swing.JButton();
botonAbrirVideo = new javax.swing.JButton();
jButton4 = new javax.swing.JButton();
jPanel4 = new javax.swing.JPanel();
jPanel3 = new javax.swing.JPanel();
jScrollPane1 = new javax.swing.JScrollPane();
editor = new javax.swing.JTextPane();
jPanel5 = new javax.swing.JPanel();
jLabel2 = new javax.swing.JLabel();
lineaEditor = new javax.swing.JLabel();
jPanel2 = new javax.swing.JPanel();
StatusBar = new javax.swing.JLabel();
jScrollPane2 = new javax.swing.JScrollPane();
tabla1 = new javax.swing.JTable();

jFrame1.getContentPane().setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());

jFrame1.getContentPane().add(nomArchivo, new org.netbeans.lib.awtextra.AbsoluteConstraints(120, 10, 160, -1));

jLabel1.setText("Nombre Archivo");
jFrame1.getContentPane().add(jLabel1, new org.netbeans.lib.awtextra.AbsoluteConstraints(10, 10, -1, -1));

jButton2.setText("Guardar");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

jFrame1.getContentPane().add(jButton2, new org.netbeans.lib.awtextra.AbsoluteConstraints(90, 40, 100, -1));

timer_recibir.setDelay(600L);

jPanel1.setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());

jPanel1.setBackground(new java.awt.Color(204, 204, 255));
jButton1.setIcon(new javax.swing.ImageIcon() {
    public javax.swing.Icon getIcon() {
        try {
            return new javax.swing.ImageIcon(
                new java.net.URL("http://epsilon:90/laboratorio/imagenes/ico_guardar.gif")
            );
        } catch (java.net.MalformedURLException e) {
        }
        return null;
    }
}.getIcon());
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

jPanel1.add(jButton1, new org.netbeans.lib.awtextra.AbsoluteConstraints(0, 0, 30, 30));

jButton3.setIcon(new javax.swing.ImageIcon() {
    public javax.swing.Icon getIcon() {
        try {
            return new javax.swing.ImageIcon(
                new java.net.URL("http://epsilon:90/laboratorio/imagenes/ico_abrir.gif")
            );
        } catch (java.net.MalformedURLException e) {
        }
        return null;
    }
}

```

```

}.getIcon());
jButton3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton3ActionPerformed(evt);
    }
});

jPanel1.add(jButton3, new org.netbeans.lib.awtextra.AbsoluteConstraints(30, 0, 30, 30));

botonCompilar.setIcon(new javax.swing.ImageIcon() {
    public javax.swing.Icon getIcon() {
        try {
            return new javax.swing.ImageIcon(
                new java.net.URL("http://epsilon:90/laboratorio/imagenes/ico_ejecutar.gif")
            );
        } catch (java.net.MalformedURLException e) {
        }
        return null;
    }
}.getIcon());
botonCompilar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        botonCompilarActionPerformed(evt);
    }
});

jPanel1.add(botonCompilar, new org.netbeans.lib.awtextra.AbsoluteConstraints(60, 0, 30, 30));

botonCorrerPrograma.setIcon(new javax.swing.ImageIcon() {
    public javax.swing.Icon getIcon() {
        try {
            return new javax.swing.ImageIcon(
                new java.net.URL("http://epsilon.autonoma.edu.co:90/laboratorio/imagenes/ico_compilar.gif")
            );
        } catch (java.net.MalformedURLException e) {
        }
        return null;
    }
}.getIcon());
botonCorrerPrograma.setDisabledIcon(new javax.swing.ImageIcon() {
    public javax.swing.Icon getIcon() {
        try {
            return new javax.swing.ImageIcon(
                new java.net.URL("http://200.75.47.135:90/laboratorio/imagenes/ico_compilarD.gif")
            );
        } catch (java.net.MalformedURLException e) {
        }
        return null;
    }
}.getIcon());
botonCorrerPrograma.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        botonCorrerProgramaActionPerformed(evt);
    }
});

jPanel1.add(botonCorrerPrograma, new org.netbeans.lib.awtextra.AbsoluteConstraints(90, 0, 30, 30));

botonVerLenguaje.setIcon(new javax.swing.ImageIcon() {
    public javax.swing.Icon getIcon() {
        try {
            return new javax.swing.ImageIcon(
                new java.net.URL("http://epsilon:90/laboratorio/imagenes/ico_lenguaje.gif")
            );
        } catch (java.net.MalformedURLException e) {
        }
        return null;
    }
}

```

```

    }
    }.getIcon());
    botonVerLenguaje.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            botonVerLenguajeActionPerformed(evt);
        }
    });

jPanel1.add(botonVerLenguaje, new org.netbeans.lib.awtextra.AbsoluteConstraints(120, 0, 40, 30));

botonAbrirVideo.setIcon(new javax.swing.ImageIcon() {
    public javax.swing.Icon getIcon() {
        try {
            return new javax.swing.ImageIcon(
                new java.net.URL("http://epsilon:90/laboratorio/imagenes/ico_video.gif")
            );
        } catch (java.net.MalformedURLException e) {
        }
        return null;
    }
}).getIcon());
botonAbrirVideo.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        botonAbrirVideoActionPerformed(evt);
    }
});

jPanel1.add(botonAbrirVideo, new org.netbeans.lib.awtextra.AbsoluteConstraints(160, 0, 30, 30));

jButton4.setIcon(new javax.swing.ImageIcon() {
    public javax.swing.Icon getIcon() {
        try {
            return new javax.swing.ImageIcon(
                new java.net.URL("http://200.75.47.135:90/laboratorio/imagenes/ico_borrar.gif")
            );
        } catch (java.net.MalformedURLException e) {
        }
        return null;
    }
}).getIcon());
jButton4.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton4ActionPerformed(evt);
    }
});

jPanel1.add(jButton4, new org.netbeans.lib.awtextra.AbsoluteConstraints(190, 0, 30, 30));

getContentPane().add(jPanel1, java.awt.BorderLayout.NORTH);

jPanel4.setLayout(new java.awt.GridLayout(2, 0));

jPanel3.setLayout(new java.awt.BorderLayout());

jPanel3.setBackground(new java.awt.Color(255, 51, 51));
editor.setName("editor");
editor.addCaretListener(new javax.swing.event.CaretListener() {
    public void caretUpdate(javax.swing.event.CaretEvent evt) {
        editorCaretUpdate(evt);
    }
});
editor.addKeyListener(new java.awt.event.KeyAdapter() {
    public void keyPressed(java.awt.event.KeyEvent evt) {
        editorKeyPressed(evt);
    }
    public void keyTyped(java.awt.event.KeyEvent evt) {
        editorKeyTyped(evt);
    }
});

```

```

    }
});

jScrollPane1.setViewportView(editor);

jPanel3.add(jScrollPane1, java.awt.BorderLayout.CENTER);

jPanel5.setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());

jPanel5.setBackground(new java.awt.Color(236, 235, 162));
jLabel2.setFont(new java.awt.Font("MS Sans Serif", 1, 11));
jLabel2.setText("Linea:");
jPanel5.add(jLabel2, new org.netbeans.lib.awtextra.AbsoluteConstraints(0, 0, -1, -1));

lineaEditor.setText("0");
jPanel5.add(lineaEditor, new org.netbeans.lib.awtextra.AbsoluteConstraints(50, 0, 362, -1));

jPanel3.add(jPanel5, java.awt.BorderLayout.SOUTH);

jPanel4.add(jPanel3);

jPanel2.setLayout(new java.awt.BorderLayout());

jPanel2.setBackground(new java.awt.Color(204, 204, 255));
StatusBar.setText("Estado:");
jPanel2.add(StatusBar, java.awt.BorderLayout.SOUTH);

jScrollPane2.setBackground(new java.awt.Color(255, 255, 255));
jScrollPane2.setVerticalScrollBarPolicy(javax.swing.JScrollPane.VERTICAL_SCROLLBAR_ALWAYS);
jScrollPane2.setAutoscrolls(true);
tabla1.setSelectionEnabled(true);
jScrollPane2.setViewportView(tabla1);

jPanel2.add(jScrollPane2, java.awt.BorderLayout.CENTER);

jPanel4.add(jPanel2);

getContentPane().add(jPanel4, java.awt.BorderLayout.CENTER);
}

private void editorCaretUpdate(javax.swing.event.CaretEvent evt) {
    // TODO add your handling code here:
    //posCursor = evt.getDot();

    //lineaEditor.setText(String.valueOf(getNumLinea(posCursor)));
}

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    editor.setText("");
    limpiarTabla();
}

private void botonAbrirVideoActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    Object args[] = {usuario};
    win.call("mostrarVideo",args);
}

private void botonVerLenguajeActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    Object args[] = {usuario};
    win.call("mostrarLenguaje",args);
}

private void botonCorrerProgramaActionPerformed(java.awt.event.ActionEvent evt) {

```

```

// TODO: add your handling code here:
StringTokenizer filas = new StringTokenizer(codigoEjecutar, "\n");
int j = 0;
while (filas.hasMoreTokens()) {
    StringTokenizer cad = new StringTokenizer(filas.nextToken(), " ");
    filaSeleccionada = j;
    seleccionarFila(filaSeleccionada);
    ejecutarComando(cad.nextToken(), cad.nextToken());
    while(!ack);
    ack = false;
    j++;
}
//filaSeleccionada = j;
//seleccionarFila(filaSeleccionada);
}

private void seleccionarFila(int fila) {
    System.out.println("Fila Seleccionada: " + fila);
    //selectionModel.setSelectionInterval(fila, fila);

    tabla1.repaint();
}

private void ejecutarComando(String c1, String c2) {
    comando = c1;
    parametro = c2;
    tmActivado = true;
    bandera2 = true;
}

/** Este método es llamada una vez se presiona una tecla del equipo, y el foco lo
 * tiene el editor
 * @param java.awt.event.KeyEvent evt Evento que ocurre una vez se presiona una tecla
 */
private void editorKeyPressed(java.awt.event.KeyEvent evt) {

    StringTokenizer strT = new StringTokenizer(editor.getText(), "\n");
    totalFilas = strT.countTokens();
    if (evt.getKeyCode() == evt.VK_ENTER) {
        filaActual++;
    }
    if (evt.getKeyCode() == evt.VK_UP) {
        if (filaActual > 1)
            filaActual--;
    }
    if (evt.getKeyCode() == evt.VK_DOWN) {
        if (filaActual < totalFilas)
            filaActual++;
    }
}
lineaEditor.setText(String.valueOf(filaActual) + " de " + totalFilas);
int i = 0;
while (i < palabras.size()) {
    String pal[] = (String[])palabras.get(i);
    highlight(pal[1], "rojo_n");
    i++;
}
highlight("repetir", "azul_n");
highlight("{", "negro_n");
highlight(")", "negro_n");
highlight(";", "negro_n");
}

/** Este método es llamado al pulsar el respectivo botón.
 * @param java.awt.event.ActionEvent evt Evento que ocurre una vez se presiona el botón
 */
private void botonCompilarActionPerformed(java.awt.event.ActionEvent evt) {
/*
    int w_f = 380; //variable que define el ancho de la nueva ventana a abrir
    int h_f = 280; //variable que define el alto de la nueva ventana a abrir
    ven_compilando.setSize(w_f, h_f); //aquí se fija el tamaño de la ventana
    int posCentral = (this.getWidth() - w_f) / 2; //se calcula la posición en X para que la ventana quede centrada

```

```

int altCentral = (this.getHeight() - h_f) / 2; //se calcula la posición en Y para que la ventana quede centrada
Point p = new Point(); //Se crea un objeto que define un punto X,Y
p.x = this.getLocationOnScreen().x + posCentral; //Se asignan las coordenada a dicho punto
p.y = this.getLocationOnScreen().y + altCentral;
ven_compilando.setLocation(p); //Se fija la posición de la ventana con base en el punto creado.
ven_compilando.setVisible(true); //Se muestra la ventana.
//editor_comp.append("Compilando...\n");
//timer_compilando.start();
//activado = true;
//c.compilar(editor.getText());
*/ //if (!self.isAlive())
// self.start();
limpiarTabla();
int elt = palabras.size(); //esta variable contiene el tamaño del vector de palabras introducido por parámetros
String lenguaje[][] = new String[elt][2]; //se crea una matriz de strings para almacenar las palabras y codigos del robot.
//Aquí se hace un barrido para llenar la matriz con los datos.
for (int i = 0; i < elt; i++) {
    String elTemp[] = (String[])palabras.get(i);
    lenguaje[i][0] = elTemp[0];
    lenguaje[i][1] = elTemp[1];
}

codigoAnalizar = formatearCodigo(editor.getText()); //Se lleva el texto del editor a un método que le da un formato apropiado para el analizador de gramática y lo
regresa en un buffer.
analizadorGramatical an = new analizadorGramatical(lenguaje, codigoAnalizar.toString()); //Se crea una instancia del analizador gramatical pasándole el código escrito por
el usuario y el lenguaje por él definido.
if (an.errorFinal.length() > 0) {
    //editor_comp1.setText(an.obtErrorFinal().toString());
    tabla1.setValueAt(an.obtErrorFinal().toString(),0,1);
    StatusBar.setText("Estado: Error");
}
else {
    //editor_comp.setText(an.obtCodigoFinal().toString());
    traductorLenguaje tr = new traductorLenguaje(lenguaje, codigoAnalizar.toString()); //Se crea una instancia del traductor, para convertir el lenguaje del usuario, en
codigo apropiado para enviar al robot.
    tr.traducir();
    codigoEjecutar = tr.obtCodigoFinal().toString();
    //System.out.println("Codigo Ejecutar: " + codigoEjecutar);
    //editor_comp1.setText(tr.obtCodigoFinal().toString());
    StringTokenizer filas = new StringTokenizer(codigoEjecutar, "\n");
    int j = 0;
    while (filas.hasMoreTokens()) {
        if (filas.countTokens() > 4)
            model.insertRow(model.getRowCount(), new Object[]{String.valueOf(j)});
        StringTokenizer cad = new StringTokenizer(filas.nextToken(), " ");
        String val1 = cad.nextToken();
        String val2 = cad.nextToken();
        switch (Integer.parseInt(val1)) {
            case 1:
                tabla1.setValueAt(String.valueOf(j+1),j,0);
                tabla1.setValueAt("Avanzar " + val2 + " pasos",j,1);
                break;
            case 2:
                tabla1.setValueAt(String.valueOf(j+1),j,0);
                tabla1.setValueAt("Retroceder " + val2 + " pasos",j,1);
                break;
            case 3:
                tabla1.setValueAt(String.valueOf(j+1),j,0);
                tabla1.setValueAt("Girar " + val2 + " grados",j,1);
                break;
            case 4:
                tabla1.setValueAt(String.valueOf(j+1),j,0);
                tabla1.setValueAt("Llevar la pinza arriba o abajo",j,1);
                break;
            case 5:
                tabla1.setValueAt(String.valueOf(j+1),j,0);
                tabla1.setValueAt("Activar el electroiman",j,1);

```

```

                break;
            default :
                break;
        }
        j++;
    }
}

private void limpiarTabla() {
    model.setNumRows(4);
    for (int i = 0; i < 4; i++)
        for (int j = 0; j < 2;j++)
            tabla1.setValueAt("", i, j);
}

/** Método encargado de dar el formato al código escrito por el usuario en el editor
 * evitando pasar errores al analizador de gramática
 * @param String codigoFuente Programa escrito por el usurio en el editor.
 * @return StringBuffer buffer con la cadena formateada.
 */
public StringBuffer formatearCodigo(String codigoFuente) {
    StringBuffer codigoFormateado = new StringBuffer();
    // StringBuffer codigoPreFormateado = new StringBuffer();
    // StringTokenizer stok = new StringTokenizer(codigoFuente, "\n\r");
    /*while(stok.hasMoreTokens()) {
        String nt = stok.nextToken();
        if (nt.length() > 0) {
            codigoPreFormateado.append(nt);
            codigoPreFormateado.append("\n\r");
        }
    }
    */
    //recorrido por toda la cadena, caracter por caracter.
    for (int i = 0; i < codigoFuente.length(); i++) {

        //bloques comparativos para formatear los elementos básicos, fuera de los tokens que son las palabras reservadas del lenguaje
        //el analizador de gramática, depende de una buena estructura de estos tokens.
        if (codigoFuente.charAt(i) == ';') {
            codigoFormateado.append(" ; ");
        }
        else {
            if (codigoFuente.charAt(i) == '\r')
                codigoFormateado.append(" \r");
            else {
                if (codigoFuente.charAt(i) == '\n')
                    codigoFormateado.append("\n ");
                else {
                    if (codigoFuente.charAt(i) == '{')
                        codigoFormateado.append(" { ");
                    else {
                        if (codigoFuente.charAt(i) == '}')
                            codigoFormateado.append(" } ");
                        else
                            codigoFormateado.append(codigoFuente.charAt(i));
                    }
                }
            }
        }
    }
    //System.out.println("Codigo preFormateado: " + codigoPreFormateado);
    return codigoFormateado;
}

/** Accion tomada al presionar el boton 3**/
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
    Object args[] = {usuario};

```

```

        win.call("abrirProgramas",args);
    }
    /** Este método, carga el programa en el editor
     * @param String programa Código fuente del programa guardado por el usuario
     */
    public void cargarPrograma(String programa) {
        editor.setText(programa);
    }
    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        int codVal = validarNomArchivo(nomArchivo.getText());
        if (codVal == 0) {
            Object args[] = {usuario, nomArchivo.getText(),editor.getText()};
            win.call("enviarFormulario",args);
            JFrame1.setVisible(false);
        }
        else {

        }
    }
    /** Este método se encarga de validar el nombre de archivo escrito por el usuario, para su respectivo almacenamiento en la base de datos
     * @param String nom Nombre del archivo
     */
    public int validarNomArchivo(String nom) {
        String delim = "?/[]{}+#!@#%&*'()~.,;";
        StringTokenizer strT = new StringTokenizer(nom, delim);
        return 0;
    }
    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        //win.eval(editor.getText());
        int w_f = 310;
        int h_f = 120;
        JFrame1.setSize(w_f, h_f);
        int posCentral = (this.getWidth() - w_f) / 2;
        int altCentral = (this.getHeight() - h_f) / 2;
        Point p = new Point();
        p.x = this.getLocationOnScreen().x + posCentral;
        p.y = this.getLocationOnScreen().y + altCentral;
        JFrame1.setLocation(p);
        JFrame1.setVisible(true);
    }

    private void editorKeyTyped(java.awt.event.KeyEvent evt) {
        // TODO add your handling code here:
    }

    public void onTime(java.awt.event.ActionEvent event) {
        if (tmActivado) {
            byte b;
            if (bandera == 0) {
                try {
                    out.write(this.comando.getBytes());
                } catch (IOException io) {}
            }
            else {
                if (bandera == 1) {
                    //try {
                    //out.writeUTF("\n");
                    //System.out.println("Escribiendo un enter\n");
                    //} catch (IOException io) {}
                }
                else {
                    if (bandera == 2) {
                        try {

```

```

        String cadena = conversionStoC(this.parametro);
        out.write(cadena.getBytes());
        //System.out.println("Enviando: " + cadena);
    } catch (IOException io) {}
    }
    else {
        if (bandera == 3) {

            //try {
            //out.writeUTF("\n");
            //System.out.println("Escribiendo un enter\n");
            //} catch (IOException io) {}
            bandera = 4;
        }
    }
}
if (bandera != 4)
    bandera++;
else {
    bandera = 0;
    tmActivado = false;
}
}
}

public void run() {

    while(Thread.currentThread() == recibir) {
        System.out.println("Entramos al hilo recibir...");
        byte b[] = new byte[8];
        try {
            in.read(b);
        }
        catch (IOException io) {}
        cadenaRecibida = new String(b);
        cadenaRecibida = cadenaRecibida.trim();
        if (cadenaRecibida.equals("u"))
            ack = true;
        else
            ack = false;
        //System.out.println("Recibiendo:---> " + cadenaRecibida);
        //System.out.println("ACK--->" + ack);
    }

    while (Thread.currentThread() == selFila) {
        // seleccionarFila(filaSeleccionada);
        //System.out.println("Fila " + filaSeleccionada + " seleccionada");
    }

    try {
        Thread.currentThread().sleep(1000);
    }
    catch (InterruptedException ie) {}
}

public String conversionStoC(String s) {
    int num = Integer.parseInt(s);
    char c = (char)num;
    return String.valueOf(c);
}
}

```

```

public String recibirSocket() {
    byte b[] = new byte[8];
    String cadenaRecibida;
    try {
        in.read(b);
    }
    catch (IOException io) {}
    cadenaRecibida = new String(b);
    System.out.println("Recibiendo:---> " + cadenaRecibida);
    return cadenaRecibida;
}

// Variables declaration - do not modify
private javax.swing.JLabel StatusBar;
private javax.swing.JButton botonAbrirVideo;
private javax.swing.JButton botonCompilar;
private javax.swing.JButton botonCorrerPrograma;
private javax.swing.JButton botonVerLnaguaje;
public javax.swing.JTextPane editor;
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JFrame jFrame1;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JPanel jPanel1;
private javax.swing.JPanel jPanel2;
private javax.swing.JPanel jPanel3;
private javax.swing.JPanel jPanel4;
private javax.swing.JPanel jPanel5;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JScrollPane jScrollPane2;
private javax.swing.JLabel lineaEditor;
private javax.swing.JTextField nomArchivo;
private javax.swing.JTable tabla1;
private org.netbeans.examples.lib.timerbean.Timer timer_recibir;
// End of variables declaration
}

```

## CLASE ANALIZADOR GRAMATICAL

```
/*
 * analizadorGramatical.java
 *
 * Created on 26 de agosto de 2004, 10:30 AM
 */

package editor2;

import java.util.StringTokenizer;
import java.lang.StringBuffer;

/**
 *
 * @author csanchez
 */
public class analizadorGramatical {

    public String codigo;
    public StringBuffer codigoFinal;
    public StringBuffer errorFinal;
    public String tokenActual;
    public StringTokenizer s;
    public String[][] lenguaje;
    public int linea;
    /** Creates a new instance of analizadorGramatical */
    public analizadorGramatical(String[][] lenguaje, String codigo) {
        this.codigo = codigo;
        this.lenguaje = lenguaje;
        codigoFinal = new StringBuffer();
        errorFinal = new StringBuffer();

        analizarPrograma();
    }
    public void analizarPrograma() {
        linea = 0;
        s = new StringTokenizer(codigo, " ");
        tokenActual = s.nextToken();
        while(s.hasMoreTokens()) {
            //tokenActual = s.nextToken();
            analizarComandoComplejo();
            //analizarComandoSencillo();
        }
        /**
         * if (errorFinal.length() > 0)
         *     obtErrorFinal();
         * else
         *     obtStringBuffer();
         */
    }
    public void analizarComandoComplejo() {
        analizarRetornoCarro();
        if (tokenActual.equals("repetir")) {
            aceptar(tokenActual);
            analizarNumero();
            analizarLlavesAbrir();
            analizarComandoSencillo();
            analizarLlavesCerrar();
            //analizarPuntoyComa();
        }
        else
            analizarComandoSencillo();
    }
    public void analizarRetornoCarro() {
```

```

        if (tokenActual.equals("\r\n")) {
            linea += 1;
            if (s.hasMoreTokens())
                tokenActual = s.nextToken();
        }
    }
    public void analizarNumero() {
        analizarRetornoCarro();
        if (esEntero(tokenActual))
            aceptar(tokenActual);
        else
            errorFinal.append("Error: Se esperaba un entero en la linea..." + linea + "\n");
        //if (Integer.parseInt(tokenActual)) {}
    }
    public void analizarPuntoyComa() {
        analizarRetornoCarro();
        if (tokenActual.equals(","))
            aceptar(tokenActual);
        else {
            errorFinal.append("Se esperaba punto y coma en la linea..." + linea + "\n");
            while (s.hasMoreTokens())
                tokenActual = s.nextToken();
        }
    }
    public void analizarLlavesAbrir() {
        analizarRetornoCarro();
        if (tokenActual.equals("(")) {
            aceptar(tokenActual);
        }
        else
            errorFinal.append("Error: se esperaba llave de apertura en la linea..." + linea + "\n");
    }
    public void analizarLlavesCerrar() {
        analizarRetornoCarro();
        if (tokenActual.equals(")"))
            aceptar(tokenActual);
        else {
            errorFinal.append("Error: se esperaba llave de cierre en la linea..." + linea + "\n");
            while (s.hasMoreTokens())
                tokenActual = s.nextToken();
        }
    }
    public void analizarComandoSencillo() {
        int cnt = 0;
        analizarRetornoCarro();
        for (int i = 0; i < lenguaje.length; i++) {
            if (tokenActual.equals(lenguaje[i][1])) {
                aceptar(tokenActual);
                analizarNumero();
                analizarPuntoyComa();
                i = lenguaje.length;
            }
            else
                cnt++;
            //else
            //    errorFinal = "Error: Se esperaba comando en la linea..." + linea;
        }
        if (cnt == lenguaje.length) {
            errorFinal.append("Error: Se esperaba comando en la linea..." + linea + "\n");
            while (s.hasMoreTokens())
                tokenActual = s.nextToken();
        }
    }
    /*
    if (tokenActual.equals("avanzar")) {
        aceptar(tokenActual);
        analizarNumero();
        analizarPuntoyComa();
    }
    */

```

```

    }
    else {
        if (tokenActual.equals("retroceder")) {
            aceptar(tokenActual);
            analizarNumero();
            analizarPuntoyComa();
        }
        else
            errorFinal = "Error: Se esperaba comando en la linea..." + linea;
    }
}
*/
}

public void aceptar(String exp) {
    codigoFinal.append(exp);
    codigoFinal.append(" ");
    if (s.hasMoreTokens())
        tokenActual = s.nextToken();
}

public boolean esEntero(String s) {
    boolean resultado = false;
    char[] digitos = {'0','1','2','3','4','5','6','7','8','9'};
    for (int i = 0; i < digitos.length; i++) {
        for (int j = 0; j < s.length(); j++) {
            if (s.charAt(j) == digitos[i])
                resultado = true;
        }
    }
    return resultado;
}

public StringBuffer obtCodigoFinal() {
    return codigoFinal;
}

public StringBuffer obtErrorFinal() {
    return errorFinal;
}

/* public static void main (String[] args) {
    String programa = "avanzar 5 ; \n retroceder 5 ; \n repetir 5 { \n avanzar n ; \n } ";
    String com[][] = new String[5][2];
    com[0][0] = "1";
        com[0][1] = "avanzar";
    com[1][0] = "2";
        com[1][1] = "retroceder";
    com[2][0] = "3";
        com[2][1] = "girar";
    com[3][0] = "4";
        com[3][1] = "bajarPinza";
    com[4][0] = "5";
        com[4][1] = "activarPinza";
    analizadorGramatical a = new analizadorGramatical(com, programa);
}
*/
}

```

## CLASE TRADUCTOR LENGUAJE

```
/*
 * traductorLenguaje.java
 *
 * Created on 30 de agosto de 2004, 10:07 AM
 */

package editor2;

import java.util.StringTokenizer;
import java.lang.StringBuffer;
import java.lang.String;
/**
 *
 * @author csanchez
 */
public class traductorLenguaje {

    private String lenguaje[][];
    private String codigo;
    private StringBuffer codigoFinal;
    private String tokenActual;
    private StringBuffer comandoActual;
    private int posActual;
    /** Crea una nueva instancia de traductorLenguaje
     * @param String lenguaje[][] Matriz con los codigos de movimiento y el lenguaje del usuario
     * @param String codigo Cadena de caracteres que contiene el codigo del usuario.
     */
    public traductorLenguaje(String lenguaje[], String codigo) {
        this.codigo = codigo;
        this.lenguaje = lenguaje;
        codigoFinal = new StringBuffer();
        comandoActual = new StringBuffer();
    }

    public void traducir() {
        codigo = codigo.replaceAll("\r\n", "");
        traducirComandos(codigo);
    }

    public void traducirComandos(String c) {
        StringTokenizer s = new StringTokenizer(c, " ");
        int sSize = c.length();
        int pTemp1 = 0;
        int pTemp2 = 0;
        int posRepetir = 0;
        int NumRep = 0;
        int posLlave = 0;
        int posLlaveC = 0;
        int cnt = 0;
        int repv = 0;
        String tokenActual = s.nextToken();
        String subcadena;
        while (s.hasMoreTokens()) {
            if (tokenActual.equals("repetir")) {
                posRepetir = c.indexOf(tokenActual, repv);
                repv = posRepetir + 7;
                if (s.hasMoreTokens())
                    tokenActual = s.nextToken();
                NumRep = Integer.parseInt(tokenActual);
                if (s.hasMoreTokens())
                    tokenActual = s.nextToken();
                posLlave = c.indexOf("{", posRepetir);
                posLlaveC = c.indexOf("}", posRepetir);
                subcadena = c.substring(posLlave + 1, posLlaveC - 1);
                for (int j = 0; j < NumRep; j++)

```

```

        traducirComandos(subcadena);
    }
    else {
        if (tokenActual.equals("{") {
            while(!tokenActual.equals("}")
                if (s.hasMoreTokens()
                    tokenActual = s.nextToken();
            if (s.hasMoreTokens()
                tokenActual = s.nextToken();
        }
        else {
            cnt = 0;
            for (int i = 0; i < lenguaje.length; i++) {
                if (tokenActual.equals(lenguaje[i][1])) {
                    String com = tokenActual;
                    if (s.hasMoreTokens()
                        tokenActual = s.nextToken();
                    int numEv = Integer.parseInt(tokenActual);
                    if (s.hasMoreTokens()
                        tokenActual = s.nextToken();
                    codigoFinal.append(codificar(com, numEv));
                    if (s.hasMoreTokens()
                        tokenActual = s.nextToken();
                    i = lenguaje.length;
                }
                else {
                    cnt++;
                }
            }
            if (cnt == lenguaje.length) {
                while (s.hasMoreTokens()
                    tokenActual = s.nextToken();
            }
        }
    }
}

public String codificar(String s1, int n1) {
    String instruccion = "";
    //char c = (char)n1;
    for (int i = 0; i < lenguaje.length; i++) {
        if (lenguaje[i][1].equals(s1)) {
            if (lenguaje[i][0].equals("act:")) {
                if (n1 == 0)
                    n1 = 100;
                if (n1 == 1)
                    n1 = 97;
            }
            if (lenguaje[i][0].equals("gir:")) {
                /* if (n1 < 60)
                    n1 = 60;
                if (n1 > 160)
                    n1 = 160;
                */
            }
            instruccion = lenguaje[i][0] + " " + n1 + "\n";
        }
    }
    return instruccion;
}

public StringBuffer obtCodigoFinal() {
    return codigoFinal;
}
}

```

## ANEXO L. ANALISIS DEL LIR: "PRACTICA DE PROGRAMACIÓN Y CONTROL DE DISPOSITIVOS ROBOTICOS".

---

Requerimientos y especificaciones iniciales.

El LIR a desarrollar tiene como misión dentro del ejercicio de una práctica que el usuario pueda controlar un dispositivo robótico de manera que recoja un elemento ubicado en el espacio de trabajo y lo desplace y ubique en otro lugar determinado.

La intencionalidad del LIR es que el usuario se enfrente a un sistema que le va a solicitar que defina su propio lenguaje (instrucciones) y que luego emplee dichas instrucciones para escribir sentencias o un programa de varias sentencias para controlar el dispositivo. Esto es porque en el mundo real no existe un lenguaje universal para la manipulación de dispositivos robóticos, así que cada cual posee su lenguaje y es por ello que para la asimilación de este concepto se deja que el usuario defina el propio.

Es también intención del LIR que el usuario tenga un espacio de entrenamiento y práctica en la manipulación remota de dispositivos, teniendo como medio de observación del sistema el video enviado a su pantalla y de esta manera vaya adquiriendo una mayor agilidad en este tipo de entornos.

Los elementos principales del LIR son:

- Un computador, que es el servidor operativo del LIR y es quien se comunica con el servidor de administración.
- Un dispositivo robótico móvil conectado al computador con una pinza en la parte delantera.
- Un espacio de trabajo en donde se encuentran dos o tres cubos de madera de un tamaño adecuado para que la pinza pueda tomarlos y donde hay unos sitios demarcados para ubicar los cubos.
- Una webcam para transmitir el video del estado de todo el sistema.

El dispositivo robótico móvil posee varias acciones básicas iniciales que son:

- Desplazarse hacia adelante
- Desplazarse hacia atrás
- Girar a la izquierda
- Girar a la derecha
- Levantar la pinza
- Bajar la pinza
- Abrir la pinza
- Cerrar la pinza

El LIR debe poseer su propio entorno de programación que permita al usuario:

- Definir sus propias instrucciones para cada una de las acciones del dispositivo robótico.
- Escribir un programa con las instrucciones definidas por el usuario.
- Compilar el programa escrito con el fin de detectar errores sintácticos.
- Ejecutar el programa escrito.
- Visualizar en tiempo real el video transmitido por la webcam del LIR para tener noción del estado y ubicación del dispositivo y así continuar con su manipulación.

Objetivo general.

Desarrollo de un laboratorio de control remoto para manipular un dispositivo móvil con el fin de poner en práctica la definición de un lenguaje propio de usuario y que sirva de ejercitación para el control y la manipulación a través de la web con la ayuda de video.

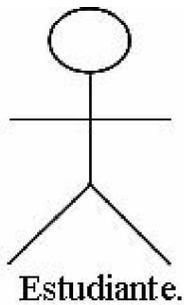
Objetivos específicos.

Construir un dispositivo móvil para ser controlado remotamente por computador, que posea movimientos básicos de desplazamiento y de recolección de elementos mediante una pinza y que una cámara pueda transmitir el video de las acciones realizadas a través de la web.

Desarrollar un ambiente de programación y control, en donde el usuario pueda definir su propio lenguaje, compilar el programa de usuario escrito y ejecutar el mismo sobre el dispositivo; permitiendo visualizar en la pantalla de usuario los movimientos y desplazamientos realizados por el dispositivo como consecuencia de la ejecución de su programa.

Actores del Sistema.

Los actores del sistema son el usuario final ó estudiante, quien es el encargado de hacer las prácticas y el sistema de administración quien es el que hace de enlace entre el estudiante y el sistema de LIR.



Casos de Uso.

El único caso de uso que implementa el LIR es el de realizar una práctica, la cual puede ser de entrenamiento o puede ser una práctica válida para ser evaluada, siendo la única diferencia obvia entre una y otra los resultados obtenidos y entregados al sistema de administración en caso de la evaluación.

Caso de uso: Hacer una práctica

Actores: Estudiante, Sistema de Administración.

Tipo: Primario.

Descripción: Realizar una práctica de control remoto sobre el dispositivo móvil desarrollado.

Curso normal de eventos

Administrador	Sistema
0. El sistema de administración hace una solicitud de práctica sobre el LIR.	1. El LIR habilita la realización de la práctica
2. El usuario debe definir las instrucciones a emplear para las diferentes acciones del dispositivo, luego da clic en Guardar.	3. El sistema guarda las instrucciones.
4. El usuario pasa a escribir sentencias o un programa con las instrucciones definidas por él. Cuando lo crea conveniente oprime en el botón de Guardar.	5. El sistema solicita un nombre para el programa.
6. El usuario ingresa el nombre correspondiente y da clic en Aceptar.	7. El sistema guarda el programa.
8. El usuario pasa a compilar el programa dando clic en el botón de Compilar.	9. El sistema compila todas las instrucciones y notifica si hay errores o si se ha ejecutado el proceso correctamente.
10. El usuario oprime el botón de Ejecutar para correr el programa.	11. El sistema ejecuta paso a paso todas las instrucciones del programa y además por medio de una webcam transmite el video de lo que sucede en tiempo real.
12. El usuario da clic en Salir.	

Cursos alternos.

Si en cualquier momento se da clic en Salir, el sistema termina la práctica y pasa el control al sistema de administración.

El usuario en cualquier momento puede pasar a redefinir instrucciones o a eliminar o adicionar nuevas.

En el paso 4, si el usuario da clic en Guardar, el sistema pide el nombre para dicho programa.

En el paso 4, si el usuario posee un programa escrito con anterioridad y lo desea abrir, da clic en el botón Abrir y el sistema le permite seleccionar el archivo.

El usuario puede modificar el programa escrito y volver a compilarlo y ejecutarlo si es necesario.

El usuario también puede comenzar un nuevo programa en el momento que desee.

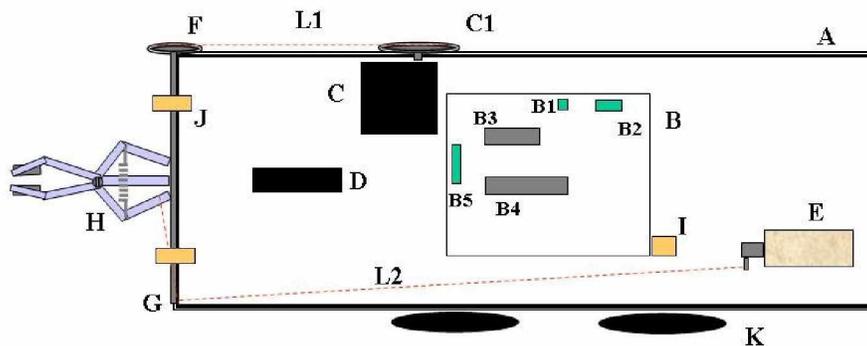
Si se presenta un error de compilación el usuario debe pasar a corregirlo y luego volver a compilar.

Cuando el usuario se sale y se debe hacer evaluación, el sistema informa los resultados al estudiante y además el sistema hace los ajustes para pasar dicha información al sistema de administración.

ANEXO M. DISEÑO DEL DISPOSITIVO MÓVIL PARA EL LIR: "PRACTICA DE PROGRAMACIÓN Y CONTROL DE DISPOSITIVOS ROBOTICOS".

En la siguiente figura se presenta una vista superior del dispositivo.

**Diseño del sistema Robótico**  
**Vista superior**

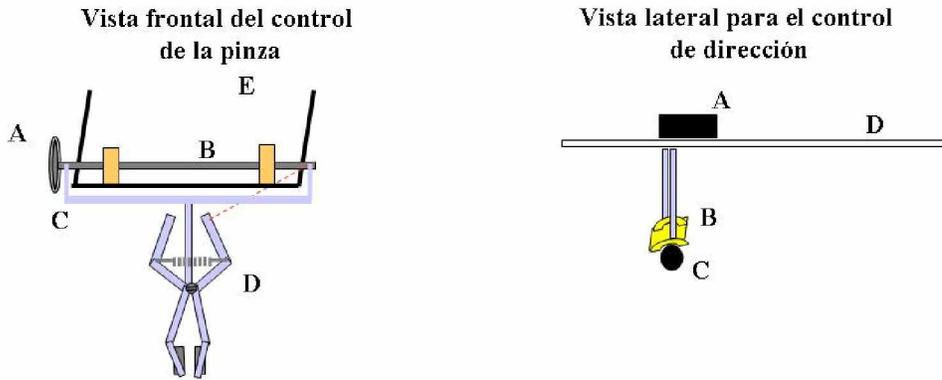


- A. Base de soporte del dispositivo.
- B. Mother board
- B1. Conector hacia el computador
- B2. Conector del motor paso a paso
- B3. MAXIM 232
- B4. Microcontrolador PIC
- B5. Conector de las fuentes de energía
- C. Servo motor. Sube y baja la pinza.  
Hace girar el eje G.
- C1. Terminal en óvalo con canal de C.
- D. Servo motor. Controla la dirección.

- E. Electroimán
- F. Ovalo con canal para hacer girar a G.
- G. Eje para subir y bajar la pinza
- H. Pinza
- I. Taco de madera para detener el eje del electroimán.
- J. Tacos de madera para soportar el eje G.
- K. Llantas.
- L1. Nylon para hacer girar a F.
- L2. Nylon para abrir la pinza (H).

En la siguiente figura se presenta una vista frontal para observar el diseño de la pinza de recolección y una vista lateral para observar el diseño para el control de la dirección del dispositivo.

## Diseño del sistema Robótico



- A. Ovalo que recibe a B.
- B. Eje para soportar y hacer girar a C.
- C. Soporte de la pinza.
- D. Pinza.
- E. Base de soporte del dispositivo.

- A. Servo motor para controlar la dirección
- B. Eje de soporte de la dirección
- C. Llantas de dirección.
- D. Base de soporte del dispositivo.

El diagrama esquemático de la tarjeta electrónica del móvil se muestra en la siguiente figura.

