

**SISTEMA PARA EL CONTROL DE ACCESO A ESPACIOS RESTRINGIDOS**

**JULIO ALBERTO GÓMEZ GUEVARA**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE FISICOMECÁNICAS  
ESCUELA INGENIERÍA DE SISTEMAS E INFORMÁTICA  
BUCARAMANGA**

**2007**

**SISTEMA PARA EL CONTROL DE ACCESO A ESPACIOS RESTRINGIDOS**

**JULIO ALBERTO GÓMEZ GUEVARA**

**Proyecto de grado para optar al título de  
Ingeniero de Sistemas**

**Director**

**Ing. LUIS IGNACIO GONZÁLEZ RAMÍREZ**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE FISICOMECÁNICAS  
ESCUELA INGENIERÍA DE SISTEMAS E INFORMÁTICA  
BUCARAMANGA**

**2007**

## DEDICATORIA

A mi padre Álvaro Gómez Mancilla y a mi madre Eva Lucía Guevara Acevedo por haber sido la razón fundamental de haber terminado mis estudios en ingeniería de sistemas, por haber siempre estado ahí apoyándome y motivándome a salir adelante, por ser un ejemplo como personas de bien y por darme el amor para poder enfrentar las dificultades que se me presentan diariamente en mi vida. Por ser mi luz y mi guía y por tenerlos siempre en mi corazón deseando sean muy felices y orgullosos por sus hijos, Julio, Carlos, Álvaro y la patuca Adriana.

A Silvia Cristina Carreño Moreno por ser mi balance emocional a lo largo de mi carrera, que a pesar de tantas peleas pero también alegrías, me mantuvo contento y enfocado en lo que estaba y debía hacer. A ella, por ser la niña de mi corazón espero este trabajo sea de su grado.

A mis hermanos que cuando me veían triste me apoyaron y me impulsaron a seguir adelante. A kike por hacerme siempre reír y por sus sabios consejos, a Carlos, por ser tan buena persona, inteligente y noble y a la mona por ser mi guía espiritual.

A Yolanda Mejía Moreno por haber compartido su tiempo y sus ideas en la realización de este proyecto y por haber sido ayuda en la realización de este libro.

## TABLA DE CONTENIDO

DEDICATORIA.....	4
RESUMEN .....	7
SUMMARY .....	9
INTRODUCCIÓN.....	11
1. PRESENTACIÓN DEL PROYECTO .....	13
1.1 OBJETIVOS.....	13
1.1.1 OBJETIVO GENERAL .....	13
1.1.2 OBJETIVOS ESPECÍFICOS .....	13
1.2 JUSTIFICACIÓN.....	14
1.2.1 Definición Del Problema.....	14
1.3 Impacto.....	16
1.3.1 Técnico .....	16
1.3.2 Económico.....	16
1.3.3 Social .....	16
1.4 Viabilidad .....	16
1.4.1 Técnica.....	16
1.4.2 Económica .....	17
1.4.3 Social .....	17
2. MARCO TEÓRICO .....	17
2.1 SISTEMAS DE CONTROL DE ACCESO .....	17
2.2 HARDWARE .....	18
2.2.1 Microcontrolador.....	19
2.2.1.1 Componentes .....	19
2.2.1.2 Ventajas del uso de microcontroladores.....	19
2.2.2 Microcontrolador 16F877.....	20
2.2.2.1 Características Principales.....	20
2.2.2.2 Diagrama de pines .....	21
2.2.2.3 Descripción de pines.....	22
2.2.3 Teclado .....	24
2.2.4 Sensor Magnético .....	26
2.2.5 INTERFASE DE COMUNICACIONES (INTERFASE SERIAL RS-232).....	26
2.2.5.1 Tipos de Comunicaciones Seriales.....	27
2.2.5.1.1 Comunicación serial síncrona.....	27
2.2.5.1.2 Comunicación serial asíncrona.....	27
2.2.5.2 Norma RS-232.....	29
2.2.5.3 Conexión del microcontrolador al puerto serie de la computadora.....	30
2.2.5.4 Cable de interfase serial RS-232 .....	31
2.2.5.5 El chip MAX 232 .....	32
2.3 FUNDAMENTOS SOBRE BASE DE DATOS .....	33
2.3.1 Definición .....	33
2.3.2 Ventajas de las bases de datos .....	33
2.3.3 Conceptos Generales .....	34
2.4 MYSQL .....	38
2.4.1 Características de MySql.....	38
2.5 JAVA .....	39
2.5.1 Historia .....	39

2.5.2 Definición .....	40
2.5.3 Conceptos Básicos de Java .....	40
2.5.4 Características de Java.....	43
2.6 MPLAB .....	45
3. MARCO METODOLÓGICO .....	47
3.1 MODELO DE DESARROLLO .....	48
3.2 LENGUAJE DE MODELADO .....	49
4. ANALISIS DE REQUERIMIENTOS Y DISEÑO GLOBAL DEL SISTEMA.....	52
4.1 RECOLECCIÓN DE REQUERIMIENTOS.....	53
4.1.1 REQUERIMIENTOS DE LA DIRECCIÓN DE ESCUELA .....	53
4.1.2 REQUERIMIENTOS DEL ADMINISTRADOR DE LA SALA.....	53
4.1.3 REQUERIMIENTOS DE LOS USUARIOS .....	54
4.2 DISEÑO GLOBAL DEL SISTEMA .....	54
4.2.1 Diagramas de Casos de Uso .....	55
4.3 DISEÑO DETALLADO .....	71
4.3.1 Prototipo Básico Módulo de Administración .....	71
4.3.2 Prototipo Básico Modulo del Microcontrolador.....	73
4.3.3 Prototipo de Interfaces.....	73
4.3.4 Prototipo de Control .....	73
5. IMPLEMENTACIÓN .....	75
5.1 MODELO DE DATOS .....	75
5.2 DESCRIPCIÓN GENERAL DE SCAAR.....	76
5.3 PRESENTACIÓN DEL SOFTWARE.....	81
5.4 ESPECIFICACIÓN DE LA INTERFASE.....	83
BIBLIOGRAFÍA .....	106
CONCLUSIONES.....	108
RECOMENDACIONES .....	109

## RESUMEN

### TITULO:

SISTEMA PARA EL CONTROL DE ACCESO A ESPACIOS RESTRINGIDOS\*

### AUTORES:

GÓMEZ GUEVARA, Julio Alberto \*\*

### PALABRAS CLAVES:

Software, hardware, control de acceso, Java, microcontroladores, PIC16F877, generación de informes estadísticos.

### DESCRIPCIÓN:

Sistema Control de Acceso Espacios Restringidos (SCAAR) permite controlar las entradas y salidas de personas en áreas restringidas; controla los tiempos en que la persona está en el área y, verifica que no existan dos eventos iguales (que alguien que ya entró no vuelva a entrar sin antes salir del sistema o viceversa).

SCAAR cuenta con un sensor magnético, chapa eléctrica, alarma, teclados para acceso, que se sitúan en la entrada de la sala de desarrollo de software

---

\* Trabajo de grado. Modalidad: Investigación.

\*\* Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingeniería de Sistemas e Informática.  
Director: Luís Ignacio González Ramírez.

de la Escuela de Ingeniería de Sistemas. Este dispositivo se comunica con la computadora central para recibir ordenes y/o enviar datos que solicite el software, validar el par código/clave suministrado a través de los teclados, registrar cualquier tipo de incidencia con el fin de realizar consultas necesarias para la dirección de escuela como: informes en tiempo real sobre las salas que estén en funcionamiento y las personas que ingresaron y se encuentran actualmente en el área, reportes estadísticos sobre el tiempo de uso de las salas, informe sobre ingresos de más de una persona por código, es decir, llevar el registro de posibles accesos de personal no autorizado, e informe del historial de las alarmas que hayan ocurrido.

El desarrollo del software estuvo guiado por el modelo evolutivo, el cual consiste en la elaboración de un prototipo sistema que se construye para evaluar mejor los requisitos que desea que cumpla. Para el análisis de las necesidades del sistema se usó los diagramas de casos de uso, diagramas de actividad, observaciones e innovaciones resultantes de las pruebas realizadas que permitieron evidenciar los principales módulos que se implementaron en el software.

## SUMMARY

### TITLE:

SYSTEM FOR CONTROLLING ACCESS TO RESTRICTED AREAS SCAAR\*

### AUTORS:

GÓMEZ GUEVARA, Julio Alberto\*\*

### KEY WORDS:

Software, Hardware, Access Control, Java, Microcontrollers, PIC16F877, Statistics' reports generation.

### DESCRIPTION:

Sistema de Control de Acceso a Areas Restringidas (SCAAR) allows to manage the in comings and out comings of people into restricted areas; It controls the times when a person is in the room and it verifies that who is in doesn't try to get in again, because it would be a bad flow of the system (the same goes around).

SCAAR has a magnetic sensor, an electronic lock, an alarm, two keyboards put one in the inside and one on the outside of the room. This dispositive gets communicated with the main computer to receive orders and/or to send the data that the software requires, validate an id and a password given by the user trough the keyboard, to register any kind of incidence looking forward to success on the necessaries consults for the manager of the school as: statistics reports about the time of usage of the rooms, reports about the incomings of

---

\* Degree project. Modality: Investigation.

\*\* Physical-mechanical Engineerings Faculty. System Engineering School. Project Director: Luis Ignacio Gonzalez Ramirez

more than one person using one single id/password, it is, recording the registers of the possible not allowed staff trying to enter to the rooms, and the reports of the historical of alarms that have been occurred.

The development of the Software was guided by the evolutive prototype, which is a prototype that consists on the creation of a prototype that is made to evaluate the requirements of the system. For the analysis of the necessities of the system it was used the diagrams of use cases, activities diagrams, observations and innovations that were given by the proves made that allow to show the principles modules that were implemented in the software.

## INTRODUCCIÓN

Históricamente, las sociedades y el ser humano han necesitado controlar el acceso a ciertas áreas y lugares. Esta necesidad es motivada por el temor a que personas inescrupulosas substraigan material valioso de acuerdo a criterios personales, sociales, comerciales, etc.

Actualmente, las organizaciones requieren controlar el acceso de su personal en áreas sensibles de la empresa, con el objetivo de llevar un registro de ingresos y minimizar los accesos no autorizados. Aunque la inversión en estos sistemas no debe ser onerosa como: la implantación de vidrios a pruebas de balas, guardias armados las 24 horas del día o cámaras de video; las organizaciones si deben contemplar controles razonables para evitar el acceso de individuos e incluso de personal "no autorizado" a los espacios restringidos. Hoy en día la tecnología ha cambiado ciertas cosas, pero en el fondo persisten razones y motivos para mantener mecanismos de control de acceso sobre áreas que se desean proteger. Los mecanismos de validación han sufrido modificaciones, actualmente, existen controles biométricos, magnéticos, visuales, etc.

El presente proyecto tiene por objeto presentar un mecanismo que permita identificar y autenticar a los usuarios, que garantice una relación costo/beneficio y que requiera la menor cantidad de esfuerzo en su implantación y uso.

El Sistema de Control de Acceso a Áreas Restringidas (SCAAR) se diseñó para supervisar el acceso de los integrantes que pertenecen a los diferentes grupos de investigación y desarrollo de software a las salas de la Escuela de Ingeniería de Sistemas (EISI) de la Universidad Industrial de Santander (UIS). Para la escuela estas áreas restringidas, ofrecen recursos informáticos, son de gran responsabilidad, y, por esta razón, se desea un uso eficiente, un nivel óptimo

de seguridad, un control de acceso seguro y un cuidado estricto de los equipos, de las áreas y de los muebles, entre otros. Para llevar a cabo se hace uso de sistemas caracterizados por ser autónomos y por cumplir su función a cabalidad brindando comodidad a quienes interactúan con ellos.

# 1. PRESENTACIÓN DEL PROYECTO

## 1.1 OBJETIVOS

### 1.1.1 OBJETIVO GENERAL

Desarrollo del hardware y del software para un sistema de control de acceso a espacios restringidos aplicado al laboratorio José Alberto Villabona.

### 1.1.2 OBJETIVOS ESPECÍFICOS

1) Diseñar e implementar un Software que permita:

1. Crear, eliminar, actualizar y consultar la información relacionada con las zonas, los grupos y los usuarios autorizados para utilizar los recursos del laboratorio José Alberto Villabona de la Escuela de Ingeniería de Sistemas de la Universidad Industrial de Santander.
2. Obtener información en tiempo real de:
  - a) Zonas Activas (en funcionamiento).
  - b) Zonas No-Activas.
  - c) Usuarios que ingresaron y que se encuentran actualmente en el área.
3. Presentar informes estadísticos sobre el uso del laboratorio por zonas, grupos y usuarios por:
  - a) Última semana (desde la fecha actual).
  - b) Último mes (desde la fecha actual).
  - c) Semanas.
  - d) Meses.
  - e) Semestres.
  - f) Año.

- 2) Diseñar e implementar un dispositivo Hardware que este en comunicación con el Software del computador para:
  1. Enviar datos que requiera el Software del *computador* (identificación de acceso, clave de seguridad, entre otros).
  2. Recibir datos que requiera el Dispositivo Hardware, como órdenes (enviadas por el Software) de pedir clave de seguridad, activar/desactivar seguro de la puerta del área restringida, activar/desactivar alarma, activar/desactivar zumbador, entre otros.
  3. Efectuar acciones físicas, como activar-desactivar seguro de la puerta del área restringida, activar-desactivar un zumbador y activar-desactivar alarma.

## **1.2 JUSTIFICACIÓN**

### **1.2.1 Definición Del Problema**

La EISI de la UIS, cuenta, actualmente, con el laboratorio José Alberto Villabona, compuesto por cuatro salas de informática para uso de los grupos de investigación y desarrollo de software, así como para soporte de las asignaturas. Dos de estas salas son asignadas a grupos de desarrollo de software e investigación y sólo pueden ser accedidas por los miembros.

Para llevar a cabo esta política, la escuela cuenta con personas encargadas de aprobar los grupos, asignarles equipos y áreas, y, llevar a cabo un control de acceso. Para lo último, el estudiante deja su carnet con el encargado (auxiliar) de tomar el registro de hora de entrada y hora de salida en el puesto de control. El acceso a las salas mencionadas depende de si existe o no personal que

controle dicho acceso, lo cual imposibilita la entrada de los usuarios los fines de semana y, en general, en las horas en que estos (los auxiliares) no se encuentren. Esto último genera inconvenientes a los usuarios debido a que, en muchos casos, son alterados sus cronogramas de trabajo al no permitírseles su ingreso libremente, sin problemas de acceso y sin depender de entes externos.

Se tiene como objetivo general del proyecto llegar a la autonomía total del sistema con un control de las áreas restringidas, brindando seguridad a los recursos allí ubicados, permitiendo a los usuarios ingresar en cualquier momento sin depender de terceros. Las zonas deben ser monitoreadas desde un punto central (PC del administrador), y que el grupo administrativo pueda confiar en la labor del sistema, en los informes que brinda y demás.

Para esto, las bases para que se alcance dando los primeros pasos en el análisis, diseño, documentación y puesta en marcha de la primera versión del Sistema que controlará el acceso a un área restringida. Este sistema contará con un Hardware y un Software que estarán en constante intercambio de datos sobre sus estados y novedades. El Hardware constará de un dispositivo ubicado en la puerta de ingreso al área restringida y llevará a cabo funciones como activar/desactivar el seguro de la puerta, el zumbador para llamar la atención del usuario y la alarma de seguridad. El Software se encontrará tanto en el *computador* como en el dispositivo Hardware antes mencionado. El Software que se encuentra en el computador tendrá funciones como: gestionar la base de datos, presentar informes, interactuar con el dispositivo de Hardware, y obtener información en tiempo real de zonas en funcionamiento, de los usuarios que ingresan y que se encuentran en la sala.

El Software ubicado en el dispositivo Hardware permitirá el control del seguro de la puerta y de la alarma, a su vez realizará la comunicación con el computador para transmitir su estado y recibir el del computador.

## **1.3 Impacto**

### **1.3.1 Técnico**

El desarrollo del proyecto propone una nueva perspectiva respecto a la forma como se accede a un recurso restringido para cierta comunidad, se logra el uso de los recursos informáticos por medio de la automatización de los procesos involucrados, además se logra un mejoramiento u optimización de los mismos.

### **1.3.2 Económico**

Este sistema proporciona una mejora en la relación costo-beneficio debido al uso eficiente de las salas, además, se protegen los recursos informáticos de la Escuela de Ingeniería de Sistemas con el uso de este dispositivo se disminuye la probabilidad de robo o fraude.

### **1.3.3 Social**

El uso de este sistema de acceso facilitará el ingreso a la sala de trabajo de los grupos de desarrollo de software haciendo más flexible su uso por parte de sus usuarios y permitiendo optimizar el uso de los recursos de la sala. Esto brinda una herramienta útil para cumplir de manera eficaz y eficiente las tareas involucradas con el trabajo de grado. El sistema permitirá mejorar las condiciones de trabajo de las personas encargadas del control y administración de la sala.

## **1.4 Viabilidad**

### **1.4.1 Técnica**

La implantación de este sistema es viable siendo un proyecto útil para la EISI, con los equipos y las herramientas técnicas requeridas para cumplir a

cabalidad el proyecto; además, se con los recursos humanos que poseen la capacidad de asimilar y compartir los conocimientos de nuevas tecnologías necesarias para el desarrollo del proyecto.

#### **1.4.2 Económica**

Cabe notar que para desarrollar este proyecto se dispone de los recursos e infraestructura necesaria, además, el beneficio económico no sólo será observado durante el proceso de evolución del proyecto, sino también en su posterior uso, exhibiendo una disminución de costos por la efectividad de la aplicación respecto a la inversión realizada.

#### **1.4.3 Social**

Los efectos colaterales derivados de la presente aplicación serán útiles, pues además de beneficiar a sus usuarios directos también afectará de manera positiva a todos los entes que de una u otra forma están relacionados con esta instancia; como lo es el caso del personal encargado del control y la administración de las salas de computadores, facilitando su labor gracias al uso de herramientas software en este proceso sistematizado.

## **2. MARCO TEÓRICO**

En este capítulo se describe la información recopilada en la fase de investigación y desarrollo, que sirve de base para la realización del presente proyecto. Se presentan los conceptos fundamentales que debe tener claros el lector para la comprensión del desarrollo general de este trabajo de grado.

### **2.1 SISTEMAS DE CONTROL DE ACCESO**

Los Sistemas de Control de Acceso permiten controlar el ingreso y la salida de las personas en un área de acceso restringido. En el caso de las Salas de Desarrollo de Software de la EISI, el acceso esta permitido, únicamente, para los integrantes de los Grupos de Investigación de dicha escuela. Este control puede hacerse tan estricto como se desee mediante la incorporación de dispositivos como teclados, que permiten ingresar una clave personal, o lectores biométricos que utilizan alguna característica física a fin de determinar si la persona que intenta ingresar al área restringida es quien dice ser.

En cuanto al control de acceso, se considera, tradicionalmente, que comprende los siguientes “mecanismos de seguridad”:

- *Identificación de usuarios*; proceso por el cual se identifica a una persona, ejemplo: la cédula de identidad para el caso de un país, o el código de identificación (User ID) para el caso de un computador.
- *Autenticación de usuarios*; proceso que tiene por objeto la confirmación que la persona que se identificó es quien dice ser; ejemplo de este proceso es el uso de palabras claves.
- *Autorización de usuarios*; proceso que determina quién tiene acceso a qué objetos, y qué tipo de acceso tiene.

Cada uno de estos mecanismos esta apoyado por una “tecnología de seguridad”, por ejemplo: palabras claves (passwords), los tokens inteligentes o de seguridad, certificados digitales, los dispositivos de reconocimiento de huellas digitales, palma de la mano, iris del ojo, etc. Estas tecnologías están presentes en una gran variedad de “productos de seguridad”<sup>1</sup>.

## **2.2 HARDWARE**

---

<sup>1</sup> Tomado de <http://www.pc-news.com/detalle.asp?sid=&id=11&Ida=1336>

## 2.2.1 Microcontrolador

Se denomina microcontrolador a un dispositivo programable capaz de realizar diferentes actividades que requieran del procesamiento de datos digitales y del control y comunicación digital de diferentes dispositivos<sup>2</sup>. Realmente consiste en un sencillo pero completo computador contenido en el corazón (chip) de un circuito integrado<sup>3</sup>.

### 2.2.1.1 Componentes

Un microcontrolador dispone normalmente de los siguientes componentes<sup>4</sup>:

- Procesador o UCP (Unidad Central de Proceso).
- Memoria RAM para Contener los datos.
- Memoria para el programa tipo ROM/PROM/EPROM.
- Líneas de E/S para comunicarse con el exterior.
- Diversos módulos para el control de periféricos (temporizadores, Puertas Serie y Paralelo, CAD: Conversores Analógico/Digital, CDA: Conversores Digital/Analógico, etc.).
- Generador de impulsos de reloj que sincronizan el funcionamiento de todo el sistema.

### 2.2.1.2 Ventajas del uso de microcontroladores

Los productos que para su regulación incorporan un microcontrolador disponen de las siguientes ventajas:

- *Aumento de prestaciones:* un mayor control sobre un determinado elemento representa una mejora considerable en el mismo.

---

<sup>2</sup> Tomado de <http://atenea.unicauca.edu.co/~sisempotrados/Documentos/pic%2016f877.pdf>

<sup>3</sup> Tomado de <http://www.monografias.com/trabajos12/microco/microco.shtml>

<sup>4</sup> Tomado de <http://www.monografias.com/trabajos12/microco/microco.shtml>

- *Aumento de la fiabilidad:* al reemplazar el microcontrolador por un elevado número de elementos disminuye el riesgo de averías y se precisan menos ajustes.
- *Reducción del tamaño en el producto acabado:* La integración del microcontrolador en un chip disminuye el volumen, la mano de obra y los stocks.
- *Mayor flexibilidad:* las características de control están programadas por lo que su modificación sólo necesita cambios en el programa de instrucciones.

## **2.2.2 Microcontrolador 16F877**

El modelo 16F877 posee características que hacen a este microcontrolador un dispositivo versátil, eficiente y práctico para ser empleado en la aplicación que, posteriormente, será detallada<sup>5</sup>.

### **2.2.2.1 Características Principales**

Este microcontrolador es fabricado por MicroChip familia a la cual se le denomina PIC.

Algunas de sus características se muestran a continuación:

- Soporta modo de comunicación serial, posee dos pines para ello.
- Amplia memoria para datos y programa.
- Memoria reprogramable: La memoria en este PIC es la que se denomina FLASH; este tipo de memoria se puede borrar electrónicamente.
- Set de instrucciones reducido (tipo RISC), pero con las instrucciones necesarias para facilitar su manejo.

---

<sup>5</sup> Tomado de <http://www.monografias.com/trabajos18/descripcion-pic/descripcion-pic.shtml>

- Frecuencia máxima: DX-20MHz
- Memoria de programa flash palabra de 14 bits: 8KB
- Posiciones RAM de datos: 368
- Posiciones EEPROM de datos: 256
- Puertos E/S: A,B,C,D,E
- Número de pines: 40
- Interrupciones: 14
- Timers: 3
- Módulos CCP: 2
- Comunicaciones Serie: MSSP, USART
- Comunicaciones paralelo: PSP
- Líneas de entrada de CAD de 10 bits: 8
- Juego de instrucciones: 35 Instrucciones
- Longitud de la instrucción: 14 bits
- Arquitectura: Harvard
- CPU: Risc
- Canales Pwm: 2

A lo largo de los capítulos de este libro, se explicará la manera como el microcontrolador realiza su labor de control en SCAAR, profundizando así en algunas de las características antes mencionadas.

#### **2.2.2.2 Diagrama de pines**

A continuación se presenta en la Figura 1 la distribución de los pines del PIC16F877.

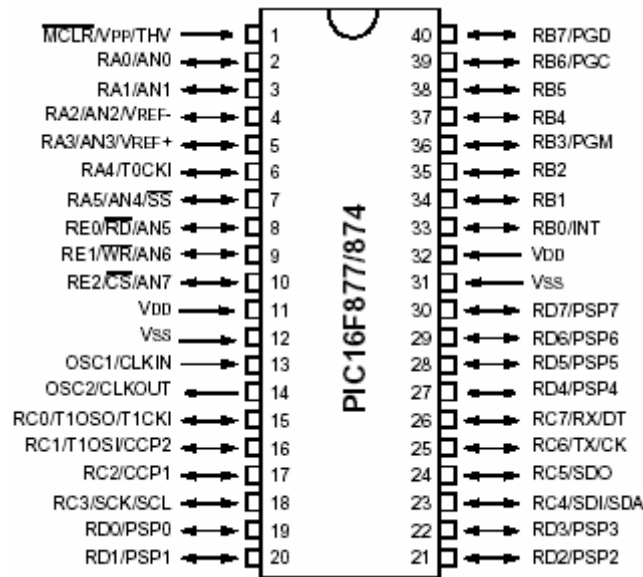


Figura 1

### 2.2.2.3 Descripción de pines

En la siguiente tabla 1 se observa la descripción de cada uno de los pines del PIC16F877<sup>6</sup>.

NOMBRE DEL PIN	PIN	TIPO	TIPO DE BUFFER	DESCRIPCIÓN
OSC1/CLKIN	13	I	ST/MOS	Entrada del oscilador de cristal / Entrada de señal de reloj externa
OSC2/CLKOUT	14	O	-	Salida del oscilador de cristal
MCLR/Vpp/THV	1	I/P	ST	Entrada del Master clear (Reset) o entrada de voltaje de programación o modo de control high voltaje test
RA0/AN0	2	I/O	TTL	PORTA es un puerto I/O bidireccional
RA1/AN1	3	I/O	TTL	RA0: puede ser salida analógica 0
RA2/AN2/ Vref-	4	I/O	TTL	RA1: puede ser salida analógica 1
RA3/AN3/Vref+	5	I/O	TTL	RA2: puede ser salida analógica 2 o referencia negativa de voltaje
RA4/T0CKI	6	I/O	ST	RA3: puede ser salida analógica 3 o referencia positiva de voltaje
RA5/SS/AN4	7	I/O	TTL	RA4: puede ser entrada de reloj el timer0.
				RA5: puede ser salida analógica 4 o el

<sup>6</sup> Tomado de <http://ww1.microchip.com/downloads/en/DeviceDoc/39582b.pdf>

				esclavo seleccionado por el puerto serial síncrono.
<b>RBO/INT</b>	33	I/O	TTL/ST	PORTB es un puerto I/O bidireccional. Puede ser programado todo como entradas
<b>RB1</b>	34	I/O	TTL	RB0 puede ser pin de interrupción externo.
<b>RB2</b>	35	I/O	TTL	RB3: puede ser la entrada de programación de bajo voltaje
<b>RB3/PGM</b>	36	I/O	TTL	Pin de interrupción
<b>RB4</b>	37	I/O	TTL	Pin de interrupción
<b>RB5</b>	38	I/O	TTL	Pin de interrupción. Reloj de programación serial
<b>RB6/PGC</b>	39	I/O	TTL/ST	
<b>RB7/PGD</b>	40	I/O	TTL/ST	
<b>RCO/T1OSO/T1CKI</b>	15	I/O	ST	PORTC es un puerto I/O bidireccional
<b>RC1/T1OS1/CCP2</b>	16	I/O	ST	RCO puede ser la salida del oscilador timer1 o la entrada de reloj del timer1
<b>RC2/CCP1</b>	17	I/O	ST	RC1 puede ser la entrada del oscilador timer1 o salida PMW 2
<b>RC3/SCK/SCL</b>	18	I/O	ST	RC2 puede ser una entrada de captura y comparación o salida PWN
<b>RC4/SD1/SDA</b>	23	I/O	ST	RC3 puede ser la entrada o salida serial de reloj síncrono para modos SPI e I2C
<b>RC5/SD0</b>	24	I/O	ST	RC4 puede ser la entrada de datos SPI y modo I2C
<b>RC6/Tx/CK</b>	25	I/O	ST	RC5 puede ser la salida de datos SPI
<b>RC7/RX/DT</b>	26	I/O	ST	RC6 puede ser el transmisor asíncrono USART o el reloj síncrono.
				RC7 puede ser el receptor asíncrono USART o datos síncronos

<b>RD0/PSP0</b>	19	I/O	ST/TTL	PORTD es un puerto bidireccional paralelo
<b>RD1/PSP1</b>	20	I/O	ST/TTL	
<b>RD2/PSP2</b>	21	I/O	ST/TTL	
<b>RD3/PSP3</b>	22	I/O	ST/TTL	
<b>RD4/PSP4</b>	27	I/O	ST/TTL	
<b>RD5/PSP5</b>	28	I/O	ST/TTL	
<b>RD6/PSP6</b>	29	I/O	ST/TTL	
<b>RD7/PSP7</b>	30	I/O	ST/TTL	
<b>RE0/RD/AN5</b>	8	I/O	ST/TTL	PORTE es un puerto I/O bidireccional  RE0: puede ser control de lectura para el puerto esclavo paralelo o entrada analógica 5  RE1: puede ser escritura de control para el puerto paralelo esclavo o entrada analógica 6  RE2: puede ser el selector de control para el puerto paralelo esclavo o la entrada analógica 7.
<b>RE1/WR/AN</b>	9	I/O	ST/TTL	
<b>RE2/CS/AN7</b>	10	I/O	ST/TTL	
<b>Vss</b>	12.3 1	P	-	Referencia de tierra para los pines lógicos y de I/O
<b>Vdd</b>	11.3 2	P	-	Fuente positiva para los pines lógicos y de I/O
<b>NC</b>	-	-	-	No está conectado internamente

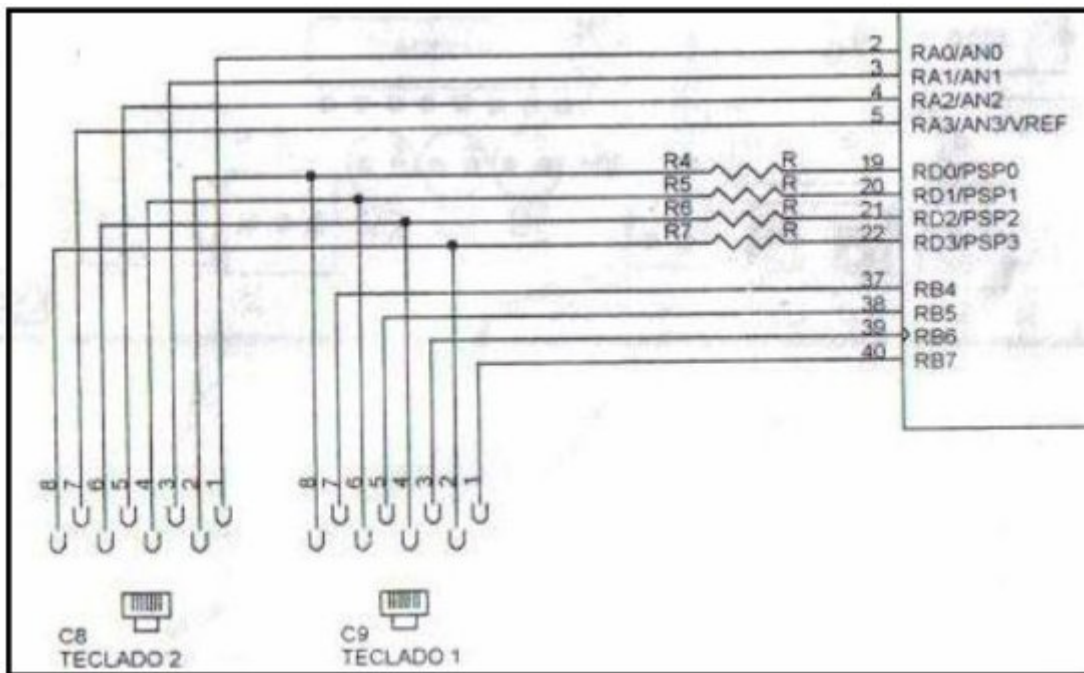
**Tabla 1**

### 2.2.3 Teclado

Desde el punto de vista eléctrico, cada tecla de un teclado es un mecanismo idéntico a un pulsador. La aportación del teclado consiste en la configuración de las teclas para que necesiten pocas líneas de entrada en la detección de la que se ha presionado.

Para disminuir las líneas necesarias para detectar la tecla pulsada, éstas se agrupan de forma matricial en filas y columnas. Con esta configuración un

teclado matricial de 16 teclas sólo precisa 8 líneas del PIC para su gestión. Si cada tecla actuase como un pulsador individual se precisarían 16 líneas de E/S del microcontrolador para gestionarlas. En pocas palabras, el teclado es un dispositivo de entrada de datos que consta de 16 teclas o pulsadores, dispuestos e interconectados en filas y columnas. En la figura 2 se muestra el esquema de un teclado matricial.



**Figura 2**

Esta ventaja de pocas líneas requeridas en el microcontrolador tiene su costo el cual se ve reflejado en la programación, al leer un puerto directamente no es posible determinar qué pulsador se está presionando sino que debemos analizar adicionalmente una combinación de ceros y unos en los puertos que intervienen en las filas o columnas y leer simultáneamente que se recibe en los puertos que intervienen en las columnas o filas. De esta manera no muy sencilla se puede determinar el pulsador presionado.

## 2.2.4 Sensor Magnético

Es un tipo de interruptor el cual es activado/desactivado por la intervención de un imán. Su instalación es muy sencilla la cual consiste en colocar el interruptor en una parte fija (como la pared en el caso del sistema SCAAR) y el imán en la parte móvil (parte que por intervención externa se desplaza alejándose o acercándose al interruptor).

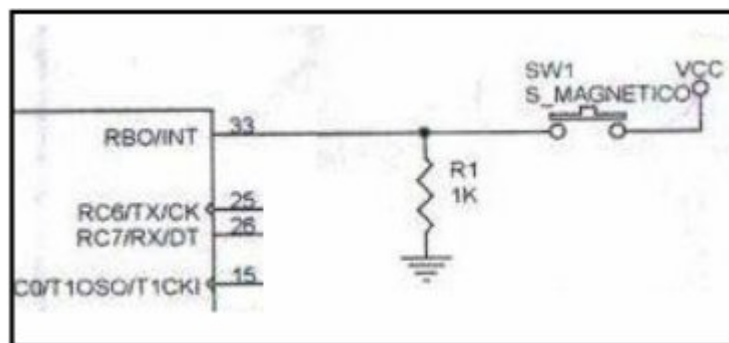


Figura 3

Cuando el imán esta cerca del interruptor éste se activa transmitiendo de uno de sus extremos al otro un impulso eléctrico pre-configurado (un uno o un cero establecido a nivel de programación en el microcontrolador); de lo contrario, el interruptor permanece desactivado sin transmitir el pulso. Se debe tener en cuenta, como se muestra en la figura 3, que al no estar activado el sensor la resistencia pull-up envía a un nivel lógico (para nuestro caso a un cero) el valor transmitido al puerto (RB0) para que éste pueda detectar si ocurre cualquier otro cambio futuro (descargar el puerto).

## 2.2.5 INTERFASE DE COMUNICACIONES (INTERFASE SERIAL RS-232)

El puerto serial de las computadoras es conocido como puerto RS-232 el cual permite la comunicación entre dos dispositivos como por ejemplo, entre una computadora y otra, una computadora y un mouse, una computadora y una impresora y para nuestro caso, una computadora y un microcontrolador. El tipo de comunicación entre estos dispositivos se conoce como comunicación serial el cual fue la que se utilizó para el desarrollo del sistema SCAAR. Otro tipo de intercambio de información binaria se conoce como comunicación paralela.

Esta tiene la característica de transmitir todos los bits de un dato de manera simultánea lo cual hace que su velocidad sea mayor comparada con la comunicación serial. Sin embargo, tiene la desventaja de utilizar una gran cantidad de líneas y esto hace que sea más costosa su implementación y además de que la señal transmitida se atenúa a grandes distancias.

### **2.2.5.1 Tipos de Comunicaciones Seriales**

Existen dos tipos de comunicaciones seriales: la síncrona y asíncrona<sup>7</sup>.

#### **2.2.5.1.1 Comunicación serial síncrona**

En la comunicación Serial síncrona, se necesitan 2 líneas, una línea sobre la cual se transmitirán los datos y otra la cual contendrá los pulsos de reloj que indicaran cuando un dato es valido. Ejemplos de este tipo de comunicación son los protocolos:

- I2C (Inter Integrated Circuit)
- SPI (Serial Peripheral Interface)

#### **2.2.5.1.2 Comunicación serial asíncrona**

En la comunicación Serial asíncrona, no son necesarios los pulsos de reloj. La duración de cada bit esta determinada por la velocidad con la cual se realiza la transferencia de datos.

Los datos serie se encuentran encapsulados en tramas de la forma:



**Figura 4. Encapsulado de un dato que se transmite serialmente**

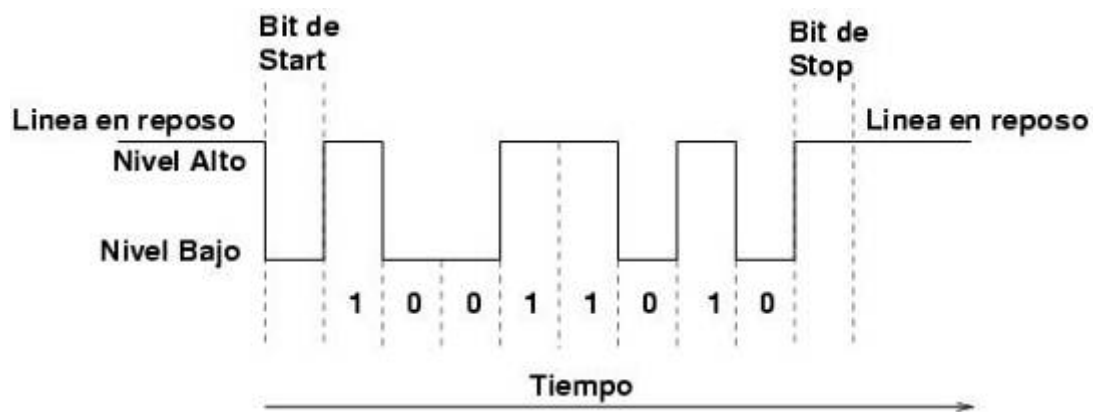
---

<sup>7</sup> Tomado de <http://www.i-micro.com/pdf/articulos/rs-232.pdf>

Primero se envía un bit de start, a continuación los bits de datos (primero el bit de mayor peso) y finalmente los bits de STOP.

El número de bits de datos y de bits de Stop es uno de los parámetros configurables, así como el criterio de paridad par o impar para la detección de errores. Normalmente, las comunicaciones serie tienen los siguientes parámetros: 1 bit de Start, 8 bits de Datos, 1 bit de Stop y sin paridad.

En la figura 5 se puede ver un ejemplo de la transmisión del dato binario 10011010. La línea en reposo está a nivel alto:



**Figura 5. Forma de transmisión serial de un dato binario**

Normalmente, cuando no se realiza ninguna transferencia de datos, la línea del transmisor se encuentra en estado de Idle (en espera), esto quiere decir en un estado alto.

Para iniciar la transmisión de datos, el transmisor coloca esta línea en bajo durante un determinado tiempo, a lo cual se le conoce como bit de arranque (Start bit) y a continuación empieza a transmitir en un intervalo de tiempo fijo, los bits correspondientes al dato, empezando siempre por el BIT menos significativo (LSB), y terminando con el BIT mas significativo. Si el receptor no está sincronizado con el transmisor, este desconoce cuando se van a recibir los datos. Por lo tanto el transmisor y el receptor deberán tener los mismos

parámetros de velocidad, paridad, número de bits del dato transmitido y de BIT de parada.

En circuitos digitales, cuyas distancias son cortas, se puede manejar transmisiones en niveles lógicos TTL (0-5V), pero cuando las distancias aumentan, estas señales tienden a distorsionarse debido al efecto capacitivo de los conductores y su resistencia eléctrica. El efecto se incrementa a medida que se incrementa la velocidad de la transmisión.

Todo esto origina que los datos recibidos no sean igual a los datos transmitidos, por lo que no se puede permitir la transferencia de datos. Una de las soluciones más lógicas, es aumentar los márgenes de voltaje con que se transmiten los datos, de tal manera que las perturbaciones por causa de la línea se puedan corregir.

#### **2.2.5.2 Norma RS-232**

Ante la gran variedad de equipos, sistemas y protocolos que existen surgió la necesidad de un acuerdo que permitiera a los equipos de varios fabricantes comunicarse entre si. La EIA (Electronics Industry Association) elaboró la norma RS-232, la cual define la interfase mecánica, los pines, las señales y los protocolos que debe cumplir la comunicación serial.

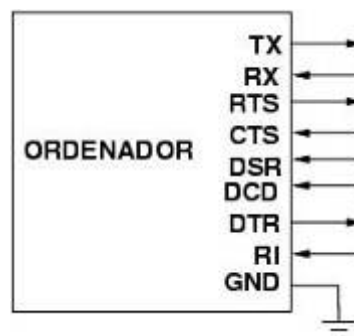
Todas las normas RS-232 cumplen con los siguientes niveles de voltaje:

- Un "1" lógico es un voltaje comprendido entre  $-5v$  y  $-15v$  en el transmisor y entre  $-3v$  y  $-25v$  en el receptor.
- Un "0" lógico es un voltaje comprendido entre  $+5v$  y  $+15 v$  en el trasmisor y entre  $+3v$  y  $+25 v$  en el receptor.

El envío de niveles lógicos (bits) a través de cables o líneas de transmisión necesita la conversión a voltajes apropiados. En los microcontroladores para representar un '0 lógico' se trabaja con voltajes inferiores a 0.8V y para un '1 lógico' con voltajes mayores a 2V. En general cuando se trabaja con familias TTL y CMOS se asume que un "0" lógico es igual a cero Volts y un "1" lógico es igual a 5 Volts.

La importancia de conocer esta norma, radica en los niveles de voltaje que maneja el puerto serial del ordenador, ya que son diferentes a los que utilizan los microcontroladores y los demás circuitos integrados. Por lo tanto se necesita de una interfase que haga posible la conversión de niveles de voltaje a los estándares manejados por los CI TTL.

Además de transmitirse los datos de una forma serie asíncrona son necesarias una serie de señales adicionales, que se definen en la norma. Las tensiones empleadas están comprendidas entre +15/-15 voltios

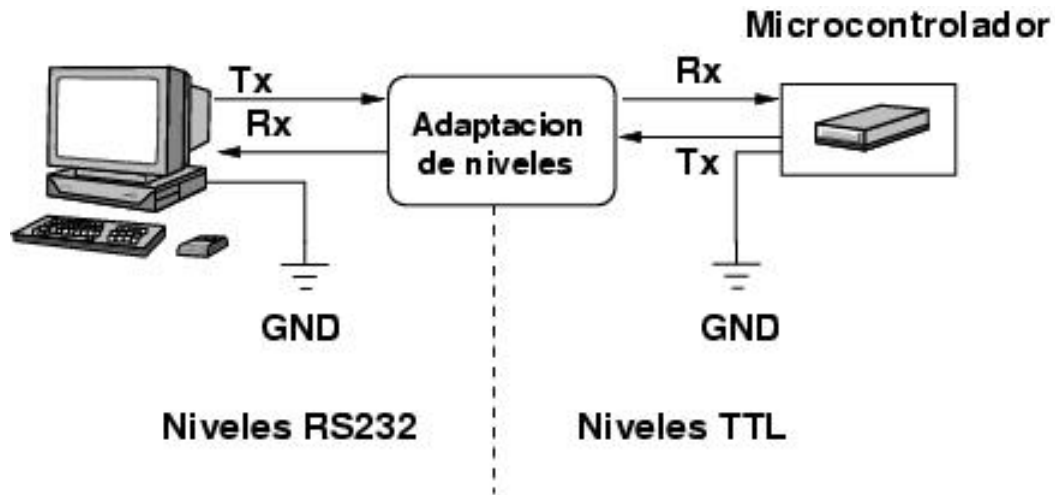


**Figura 6. Norma RS-232**

### **2.2.5.3 Conexión del microcontrolador al puerto serie de la computadora**

Para conectar el PC a un microcontrolador por el puerto serie se utilizan las señales Tx, Rx y GND. El PC utiliza la norma RS232, por lo que los niveles de tensión de los pines están comprendidos entre +15 y -15 voltios. Los

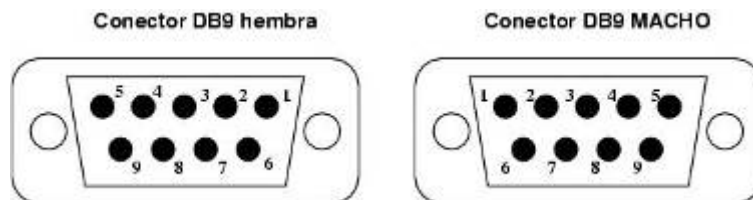
microcontroladores normalmente trabajan con niveles TTL (0-5v). Es necesario por tanto intercalar un circuito que adapte los niveles<sup>8</sup>:



**Figura 7. Conexión del Microcontrolador al Puerto Serial de la Computadora**

#### 2.2.5.4 Cable de interfase serial RS-232

Es un cable con dos conectores DB9 en sus extremos, uno hembra y otro macho, por medio del cual se establece la comunicación entre la computadora que tiene instalado el software y el dispositivo hardware. El extremo conector hembra se conecta al puerto serial de la computadora y el extremo macho al conector DB9 hembra de la tarjeta programadora.



**Figura 8. Conector DB9**

<sup>8</sup> Tomado de <http://www.learobotics.com/proyectos/cuadernos/ct1/ct1.html>

La información asociada a cada uno de los pines es la siguiente:

<b>Número de pin</b>	<b>Señal</b>
1	<b>DCD</b> (Data Carrier Detect)
2	<b>RX</b>
3	<b>TX</b>
4	<b>DTR</b> (Data Terminal Ready)
5	<b>GND</b>
6	<b>DSR</b> (Data Sheet Ready)
7	<b>RTS</b> (Request To Send)
8	<b>CTS</b> (Clear To Send)
9	<b>RI</b> (Ring Indicator)

**Tabla 2**

#### **2.2.5.5 El chip MAX 232**

Este circuito soluciona los problemas de niveles de voltaje cuando se requiere enviar unas señales digitales sobre una línea RS-232. Este chip se utiliza en aquellas aplicaciones donde no se dispone de fuentes dobles de +12 y -12 Volts. El MAX 232 necesita solamente una fuente de +5V para su operación, internamente tiene un elevador de voltaje que convierte el voltaje de +5V al de doble polaridad de +12V y -12V. Cabe mencionar que existen una gran variedad de CI que cumplen con la norma RS-232 como lo son: MAX220, DS14C232, MAX233, LT1180A. Sin embargo por el precio y su facilidad de obtención, se optó por el CI MAX232 de la marca MAXIN<sup>9</sup>.

---

<sup>9</sup> Tomado de [http://www.i-micro.com/pdf/articulos/Serial\\_RS232.pdf](http://www.i-micro.com/pdf/articulos/Serial_RS232.pdf)

El esquema es el siguiente:

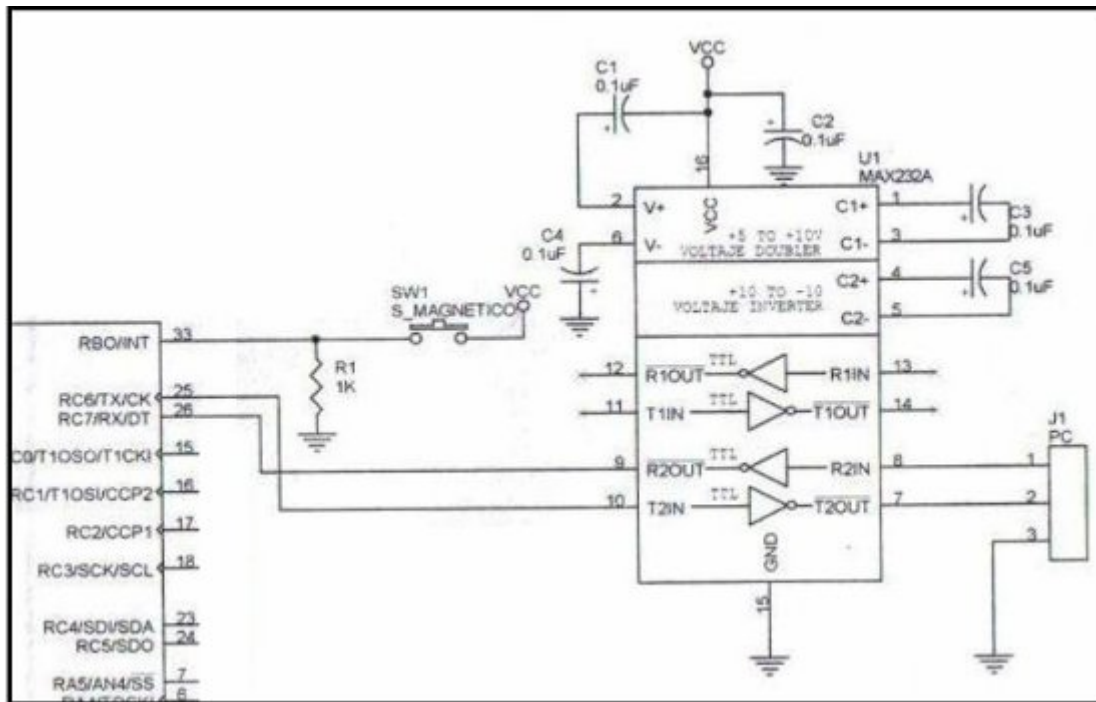


Figura 9. MAX-232

## 2.3 FUNDAMENTOS SOBRE BASE DE DATOS

### 2.3.1 Definición

Es un conjunto exhaustivo no redundante de datos estructurados organizados independiente de su uso y su implementación en máquina accesibles en tiempo real y compatibles con usuarios concurrentes con necesidad de información diferente y no predicable en tiempo<sup>10</sup>.

### 2.3.2 Ventajas de las bases de datos

Entre las ventajas más importantes de las bases de datos se encuentra:

- Independencia de datos y tratamiento.

<sup>10</sup> Tomado de <http://www.monografias.com/trabajos11/basda/basda.shtml>

- Cambio en datos no implica cambio en programas y viceversa (Menor coste de mantenimiento).
- Coherencia de resultados.
- Reduce redundancia.
- Acciones lógicamente únicas.
- Se evita inconsistencia.
- Mejora en la disponibilidad de datos.
- Restricciones de seguridad.
- Accesos (Usuarios a datos).
- Operaciones (Operaciones sobre datos).
- Más eficiente la gestión de almacenamiento.

### **2.3.3 Conceptos Generales**

Campo. Un determinado elemento de información contenido en un registro<sup>11</sup>.

Índice. Se trata de un tipo de tabla particular que contiene los valores de uno o varios campos clave y punteros hacia la posición del registro actual. Estos valores y punteros se encuentran almacenados en un determinado orden y pueden emplearse para presentar la información en ese mismo orden.

Consulta. Una orden de lenguaje de consulta estructurado (SQL) diseñado para recuperar un cierto grupo de registros de uno o varias tablas o realizar una operación en una tabla.

Recordset. Un conjunto de registros creado a partir de una o varias tablas en una base de Datos.

Clave. Una tarea importante en la modelación de bases de datos es especificar cómo se van a distinguir las entidades y las relaciones. La clave primaria es un conjunto de uno o más atributos que, tomados colectivamente, nos permite

---

<sup>11</sup> Tomado de <http://www.javeriana.edu.co/puj/viceadm/drh/capacitacion.old/nmaterial/access/c2/c2.html>

identificar de forma única una entidad en el conjunto de entidades y una relación en el conjunto de relaciones.

**Integridad Referencial.** Reglas que se siguen para preservar las relaciones definidas entre las tablas cuando se introducen o eliminan registros. Si se exige la Integridad Referencial se impide agregar registros a una tabla relacionada cuando no existe un registro asociado en la tabla principal, realizar cambios en los valores de la tabla principal que resultarían en registros huérfanos en una tabla relacionada y eliminar registros de la tabla principal cuando existen registros relacionados coincidentes en una tabla relacionada<sup>12</sup>.

**SQL.** Lenguaje de consulta estructurado. El estándar aceptado internacionalmente para sistemas relacionales, que incluye no sólo consulta sino también la definición de base de datos, manipulación, seguridad y algunos aspectos de integridad referencial.

**Sistema de Gestión de Base de Datos.** Un sistema de gestión de base de datos (DBMS) consiste en una colección de datos interrelacionados y una colección de programas para acceder a esos datos. El objetivo primordial de un DMBS es proporcionar un entorno que sea a la vez conveniente y eficiente para ser utilizado al almacenar y extraer información de la base de datos.

Los sistemas de bases de datos están diseñados para gestionar grandes bloques de información. La gestión de estos datos implica tanto la definición de estructuras para el almacenamiento de la información como la provisión de mecanismos para la manipulación de la información. Además, los sistemas de bases de datos deben mantener la seguridad de la información almacenada, pese a caídas del sistema o intentos de acceso no autorizados. Si los datos van a ser compartidos por varios usuarios, el sistema debe evitar posibles resultados anómalos.

---

<sup>12</sup> Tomado de <http://www.lcc.uma.es/~canal/iat/Tema4.pdf>

Gestor de la base de datos. Un gestor de base de datos es un módulo de programa que proporciona la interfaz entre los datos de bajo nivel almacenados en la base de datos y los programas de aplicación y consultas hechos al sistema. El gestor de base de datos es responsable de las siguientes tareas<sup>13</sup>:

- Interacción con el gestor de archivos.
- Implantación de la integridad.
- Implantación de la seguridad.
- Copia de seguridad y recuperación.
- Control de concurrencia.

Administrador de base de datos. Una de las razones principales para tener sistemas de gestión de bases de datos es tener control central de los datos y de los programas que acceden a esos datos. La persona que tiene dicho control central sobre el sistema se llama administrador de la base de datos (Database administrator (DBA) ).

Las funciones del administrador de base de datos incluyen<sup>14</sup>:

- Definición de esquema.
- Definición de la estructura de almacenamiento y del método de acceso.
- Modificación del esquema y de la organización física.
- Concesión de autorización para el acceso a los datos.
- Especificación de las restricciones de integridad.

Modelo de datos. Conjunto de herramientas conceptuales para describir los datos y de ellos, sus relaciones, semántica asociada y restricciones de consistencia. Existen tres grupos de modelos de datos<sup>15</sup>:

---

<sup>13</sup> Tomado de [http://html.rincondelvago.com/bases-de-datos\\_9.html](http://html.rincondelvago.com/bases-de-datos_9.html)

<sup>14</sup> Tomado de <http://www.monografias.com/trabajos34/base-de-datos/base-de-datos.shtml>

<sup>15</sup> Tomado de <http://www.ica.org.ve/guidox/bd/gubdpres02er.pdf>

- Lógicos basados en objetos: Describen los datos a nivel conceptual y de visión. Poseen la característica de proporcionar capacidad de estructuración bastante flexible y permite la definición de restricciones de datos explícitamente.
- Lógicos basados en registros: Describen los datos a nivel conceptual y físico. La base de datos está estructurada en registros de formato fijo de varios tipos. Donde cada registro define un número fijo de campos de longitud fija. Tales modelos son: relacional, de red y jerárquicos.

Modelo Entidad Relación. Es uno de los modelos de datos más utilizados actualmente debido a que facilita el diseño de la base de datos al permitir la especificación de un esquema que representa la estructura lógica global de la base de datos.

Entidad. Una entidad es una cosa u objeto que posee un significado de interés para el sistema en cuestión sobre las necesidades de información que se van a conocer o a mantener. Sobre la entidad se almacenan, capturan o procesan datos. Cada entidad es de existencia única dentro de un mismo diagrama Entidad. Se representa mediante un rectángulo con el nombre de la entidad en primer lugar y una lista de atributos o campos que describen dicha entidad. El nombre de la entidad se da en mayúscula y en singular, debe ser lo más representativo posible del tipo o clase de elemento, no de las instancias. Cada entidad posee un atributo llave o campo clave, el cual por convención es el primer mencionado.

Relación. Es una asociación entre dos entidades, cada terminal está caracterizado por un nombre, una cardinalidad y una opcionabilidad. La relación es representada por una línea que conecta las entidades que se relacionan. Existen tres tipos de relaciones: de uno a uno, de uno a muchos y de muchos a muchos.

## 2.4 MYSQL

Programa cliente para crear y usar una base de datos, confiable, rápido, compacto, poderoso y multiplataforma podemos hacer las bases de datos a código abierto.

Esta base de datos la desarrolló la empresa Mysql AB, una gran ventaja es que se puede utilizar gratis y su código fuente siempre esta disponible<sup>16</sup>.

### 2.4.1 Características de MySql

Entre las características disponibles en las últimas versiones se puede destacar<sup>17</sup>:

- Amplio subconjunto del lenguaje SQL. Algunas extensiones son incluidas igualmente.
- Disponibilidad en gran cantidad de plataformas y sistemas.
- Diferentes opciones de almacenamiento según si se desea velocidad en las operaciones o el mayor número de operaciones disponibles.
- Transacciones y claves foráneas.
- Conectividad segura.
- Replicación.
- Búsqueda e indexación de campos de texto.

---

<sup>16</sup> Tomado de <http://www.monografias.com/trabajos24/oracle/oracle.shtml>

<sup>17</sup> Tomado de <http://es.wikipedia.org/wiki/MySQL>

## 2.5 JAVA

### 2.5.1 Historia

Tal vez la contribución más importante a la fecha, por parte de la revolución del microprocesador, es que hizo posible el desarrollo de las computadoras personales, que ahora suman cientos de millones a nivel mundial. Las computadoras personales han tenido un profundo impacto en la vida de las personas, y en manera en que las empresas realizan y administran su negocio.

Muchas personas creen que la siguiente área importante en la que los microprocesadores tendrán un profundo impacto es en los dispositivos electrónicos para uso doméstico. Al reconocer esto, Sun Microsystems patrocinó en 1991 un proyecto interno de investigación denominado Green. El proyecto desembocó en el desarrollo de un lenguaje basado en C++ al que su creador, James Gosling, llamó Oak debido a un roble que tenía a la vista desde su ventana en las oficinas de Sun. Posteriormente se descubrió que ya existía un lenguaje de programación con el mismo nombre. Cuando un grupo de gente de Sun visitó una cafetería local, sugirieron el nombre Java (una variedad de café) y así se quedó.

Pero el proyecto Green tuvo algunas dificultades. El mercado para los dispositivos electrónicos inteligentes de uso doméstico no se desarrollaba tan rápido como Sun había anticipado. Pero aún, un contrato importante por el que Sun había competido se le otorgó a otra empresa. De manera que el proyecto corría el riesgo de cancelarse. Pero para su buena fortuna, la popularidad de World Wide Web explotó en 1993 y la gente de Sun se dio cuenta inmediatamente del potencial de Java para agregar contenido dinámico y animaciones a las páginas web. Esto trajo nueva vida al proyecto.

Sun anunció formalmente a Java en una conferencia importante que tuvo lugar en mayo de 1995. Por lo general, un evento como éste no habría generado

mucha atención. Sin embargo, Java generó un interés inmediato en la comunidad de negocios, debido al fenomenal interés en World Wide Web. En la actualidad, Java se utiliza para desarrollar aplicaciones empresariales a gran escala, para mejorar la funcionalidad de los servidores de World Wide Web (las computadoras que proporcionan el contenido que vemos en nuestros navegadores Web), para proporcionar aplicaciones para los dispositivos domésticos (como teléfonos celulares, radio localizadores y asistentes digitales personales) y para muchos otros propósitos<sup>18</sup>.

### **2.5.2 Definición**

Java es una plataforma virtual de software desarrollada por Sun Microsystems, de tal manera que los programas creados en ella puedan ejecutarse sin cambios en diferentes tipos de arquitecturas y dispositivos computacionales ("Diferentes plataformas").

### **2.5.3 Conceptos Básicos de Java**

Programación Orientada a Objetos (POO)

Junto con el paradigma de la orientación a procedimientos, son las dos filosofías generales de diseño más importantes. A diferencia de la orientación a procedimientos (OP), la orientación a objetos (OO) no concibe los procesos como una secuencia de procedimientos con su entrada y salida sino que se basa en un conjunto de objetos interactuando.

Veamos a continuación los aspectos más destacados de esta filosofía general de diseño.

---

<sup>18</sup> Tomado de <http://www.mygnet.com/articulos/java/769/>

1. Clases y objetos. Es importante distinguir entre los conceptos de clase y objeto:

Clase: Es un modelo abstracto de un tipo de objeto. Define sus métodos y atributos.

Objeto: Es una instancia de una clase, es decir, la implementación con valores de un modelo abstracto.

Las clases no son entidades independientes sino que se agrupan jerárquicamente heredando características y atributos. Cada instancia o implementación real de una clase constituirá un nuevo objeto por lo que se pueden crear infinitos objetos distintos a partir de una sola clase.

2. Encapsulación. Se define como el proceso de empaquetar juntos los métodos y los datos en un objeto. El objeto se encarga de ocultar sus datos al resto de objetos. La encapsulación permite una seguridad mayor en el acceso a los datos ya que este acceso depende directamente de cada objeto. Asimismo, permite abstraer los detalles internos de funcionamiento del objeto.

3. Intercambio de mensajes. Los objetos se comunican entre sí mediante mensajes de invocación a métodos.

4. Herencia. Es el concepto que define la adopción de todas las características de una clase por parte de otra clase que es definida como descendiente o heredera de la primera.

La principal consecuencia de la herencia es la posibilidad de reutilizar clases ya que se pueden crear nuevas a partir de las ya creadas.

La herencia puede ser de dos tipos, simple si sólo es posible heredar características de una sola clase, o múltiple si se pueden heredar características de varias clases<sup>19</sup>.

5. Polimorfismo. Genéricamente, el polimorfismo es la capacidad de tomar varias formas.

Aplicado a los lenguajes de programación, debemos considerar dos tipos de polimorfismo: polimorfismo funcional y polimorfismo de datos.

Polimorfismo funcional o sobrecarga: El polimorfismo funcional es la posibilidad de referenciar, mediante el mismo identificador, diferentes funciones. La mayoría de los lenguajes de programación presentan sobrecarga de operadores. En algunos, también las funciones de entrada/salida están sobrecargadas, pero sólo unos pocos admiten el polimorfismo en funciones definidas por el programador.

En los Lenguajes Orientados a Objetos, las operaciones se aplican siempre a un objeto. El método que se ejecuta viene determinado por la clase a la que pertenece el objeto que recibe el mensaje. Por este motivo no hay limitaciones a la sobrecarga de funciones definidas en clases distintas.

Polimorfismo de datos: El polimorfismo de datos es la capacidad de un identificador para hacer referencia a instancias de distintas clases. El polimorfismo de datos es una característica de los lenguajes orientados a objetos.

El polimorfismo de datos aumenta las posibilidades de reutilización de código, favoreciendo la construcción de módulos como extensión de otros ya existentes. En los lenguajes con polimorfismo de datos es necesario distinguir entre:

---

<sup>19</sup> Tomado de <http://www.unav.es/cti/manuales/Java/indice.html>

- Tipo estático: tipo con el que se declara una referencia o variable.
- Tipo dinámico: tipo de objeto asociado a una referencia en un momento dado.

En muchos lenguajes, el polimorfismo de datos ésta limitado por las relaciones de herencia: el tipo dinámico ha de ser un descendiente del tipo estático. Las razones son dos:

- Cualquier instancia de una clase es instancia de las clases antecesoras.
- Cualquier instancia de una clase posee los atributos declarados por las clases antecesoras y entiende los métodos definidos por ellas<sup>20</sup>.

#### **2.5.4 Características de Java**

Las características principales que nos ofrece Java respecto a cualquier otro lenguaje de programación, son:

**Simplicidad.** El código Java se expresa en archivos de texto comunes, y tiene una apariencia muy familiar para los programadores de C/C++ y para los programadores en general. Los fragmentos del lenguaje que se han visto, lo ejemplifican.

**Orientación a objetos.** A primera vista parece que todos los productos nuevos de software tienen algo "orientado a objetos". En el caso de Java, la aplicación de esa calificación es totalmente acertada. De hecho, en Java, todo es un objeto. Se verá que éstos se definen en clases, con atributos y métodos, agrupándose en paquetes, y exponiendo interfaces, permitiendo herencia y

---

<sup>20</sup> Tomado de [http://www.bice.com.ar/uploaded/PDF/teoria\\_polimorfismo.pdf](http://www.bice.com.ar/uploaded/PDF/teoria_polimorfismo.pdf)

polimorfismo, implementando así la tecnología de objetos de una manera muy completa, y a la vez sencilla y agradable.

Soporte de múltiples plataformas. El proceso de compilación no produce código para un procesador en particular, como en los compiladores C/C++, sino que genera código para un procesador ideal, denominado máquina virtual Java (Java Virtual Machine, o Java VM). La razón de esta conducta es simple: la necesidad de poder ejecutarse en cualquier plataforma, sin necesidad de cambiar el código fuente, ni aun de recompilar. Todo programa Java se traduce en una serie de códigos (llamados bytecodes) para un procesador imaginario, al Java VM, que puede ser implementado fácilmente en cualquier plataforma poderosa. Es así como el lenguaje y sus aplicaciones se independizan de la máquina destino, a costa de algo de velocidad, provocada por la necesidad de "simular" la Java VM, mediante software. Aún los tipos primitivos de datos quedan definidos de un solo golpe, para todas las plataformas. Nos evitamos así las pesadillas de portabilidad, conocidas por los programadores de C/C++, al cambiar, por ejemplo, de ambiente a ambiente, el tamaño de los enteros, o el conjunto de caracteres soportados. En Java, esos problemas no existen: sólo existe una máquina virtual.

Manejo automático de memoria. Ya se ha mencionado, que el lenguaje Java se compone de objetos. Lo que no se ha descrito, es que esos objetos pueden y deben crearse, y tienen una vida que dura hasta su destrucción. Mientras que la creación de los objetos se deja bajo la voluntad del programador, la destrucción definitiva de un objeto ocurre cuando no es más referenciado por otros objetos del programa. De esta forma, se elimina una de las causas más comunes de error en otros lenguajes, como la destrucción por el programador de objetos aún en uso en el programa, o la falta de destrucción de objetos que ya son inútiles, pues no se usan en el resto de la ejecución, pero que molestan con empleo de recursos. Esta técnica de manejo automático de memoria ocupada por los objetos se denominan garbage collection, algo así como recolección de basura. En una aplicación Java hay siempre un proceso,

ejecutado como un hilo de ejecución separado, que se ocupa de recorrer la memoria donde se encuentran los objetos, y determinan cuáles son pueden liberarse y destruirse.

Seguridad. Java ha sido adaptado a Internet, pensando mucho en el tema seguridad. Una "applet" java se ejecuta en un entorno que impide su intromisión en la máquina cliente. No hay, entonces, posibilidad de un virus Java, viajando por toda la red<sup>21</sup>.

## **2.6 MPLAB**

El MPLAB es un programa que permite la simulación en pantalla del funcionamiento de los microcontroladores PIC, es un programa de acceso gratuito y que puede bajarse desde el apartado de herramientas (Tool) de la página de Microchip<sup>22</sup>.

El MPLAB®IDE V7.00 integra en un único entorno de desarrollo soporte para todas las herramientas, con las últimas versiones del editor MPEditor v7.00, del simulador MPLAB SIM v8.70, del ensamblador MPASM v3.90, del compilador MPLAB C18, del linkador MPLINK v3.90, del generador de librerías MPLIB v3.90 para microcontroladores PICs, ensamblador/linkador/librerías para dsPIC ASM30, LINK30, LIB30 v1.20.03, compilador MPLAB C30, simulador MPLAB SIM30 v 3.00, y el control de herramientas hardware como son los emuladores en circuito MPLAB ICE 2000 v4.00 y la versión v2.00 del MPLAB ICE 4000, depurador en circuito MPLAB ICD2 v1.70, programador de desarrollo PICSTART Plus v4.30.04, programadores de producción PRO MATE II v7.00 y PM3 v2.00, la PicKit1 vía USB v2.00 También se pueden incluir la integración de herramientas de terceras compañías como los compiladores C, de HITECH y el de CCS.

---

<sup>21</sup> Tomado de <http://www.monografias.com/trabajos16/java/java.shtml>

<sup>22</sup> Tomado de <http://www.terra.es/personal/fremiro/mplab.htm>.

Las mejoras fundamentales que se han conseguido con esta versión de 32 bits son: optimización del tiempo de programación con el PICSTART® Plus mediante la selección del rango de programa de forma automática, incluye también una característica para suspender la programación de una forma rápida y segura. El trazado con filtrado, incorporado en versiones anteriores incluye en la memoria de trazado solo el código seleccionado en el tiempo de ejecución, ahora el MPLAB®IDE V7.00 incorpora el filtrado de exclusión para el MPLAB ICE 2000, MPLAB ICE 4000 y el simulador y que asegura que las secciones marcadas de código serán excluidas de la memoria de trazado, esta última característica es útil para quitar del trazado los bucles de temporización que a menudo llenan y pierden espacio en la memoria de trazado, también se ha modificado el cuadro de dialogo de los puntos de ruptura para el ICD2 mostrando los puntos de ruptura permitidos y cuantos se están usando. Esta característica nos ayuda particularmente para saber el número de puntos de ruptura, usando el ICD2, que tiene cada dispositivo. Con el MPLAB®IDE V7.00 se introduce la simulación para nuevos microcontroladores de la gama base y la gama media. No solo desaparecen completamente cientos de dlls sino que se han implementado muchas de las características para la simulación de los nuevos PIC18X y dsPIC® incluyendo UART1 y soporte de nuevos estímulos síncronos de reloj.

Finalmente con la versión de MPLAB®IDE V7.00 ha evolucionado para incorporar nuevos procedimientos de instalación. En vez de descargar un fichero cercano a 40 MB para cada versión, el usuario descargará una utilidad de instalación y seleccionará la herramienta que desea instalar. La utilidad descargará automáticamente solo los componentes necesarios. Así el usuario tardará un 50% menos de tiempo en la descarga<sup>23</sup>.

---

<sup>23</sup> Tomado de [http://www.sagitron.es/notas\\_prensa/microchip/mcv700.htm](http://www.sagitron.es/notas_prensa/microchip/mcv700.htm)

### 3. MARCO METODOLÓGICO

En la realización de todo proyecto es fundamental seguir una metodología que se encuentre de acuerdo con las características del mismo, para obtener un sistema que cumpla con los objetivos propuestos. En un sentido amplio se puede entender un proceso como aquel conjunto de actividades, que tomadas en su totalidad, producen un resultado de valor para un cliente.

Las metodologías descomponen el proceso hasta el nivel de tareas o actividades elementales. Para cada tarea se identifica un procedimiento que define la forma de ejecutarla y es el vehículo de comunicación entre desarrolladores y usuarios. El modelo de desarrollo permite realizar la especificación de requerimientos, desarrollo de los módulos, implementación, revisión y pruebas, entre otras actividades necesarias para obtener el producto deseado.

Dentro del modelo, el análisis de requisitos constituye la fase de comunicación más intensa del proceso de ingeniería del software. Esta fase es fundamental para obtener el éxito del producto final que requiere un perfecto entendimiento y comprensión del mismo. En este proceso de comprensión de necesidades y aplicación de principios operativos se obtienen uno o más productos, estos son productos intermedios, si sirven de base para realizar nuevos productos durante el desarrollo de productos finales, o puede desecharse después de la aprobación del cliente o emplearlo como primera parte de una actividad de análisis a la que seguirá el diseño y la construcción es decir el prototipo se convierte en la primera versión del producto terminado.

Los estilos de desarrollo del producto varían entre las diferentes clases de proyectos, necesitando diferentes clases de tareas y distintos órdenes de las mismas. La selección equivocada de un modelo de ciclo de vida puede ocasionar omisión de tareas o una secuenciación inapropiada de las mismas, lo cual va en contra de la planificación y eficiencia del proyecto. Una buena

elección por el contrario, es una garantía de que el esfuerzo se utiliza eficientemente.

Es así como se decidió adoptar en el desarrollo del Sistema de Control de Acceso a Áreas Restringidas el modelo evolutivo, el cual se basa en la idea de mantener un continuo contacto con el usuario en la etapa de análisis. Esto se hace en un momento en la que el usuario evalúa un prototipo, que evitando el diseño detallado, se preocupa más en el flujo de información y en la interfaz que este evalúa hasta crear un prototipo final, que será lo más cercano a los requerimientos del mismo. Luego se procede a la programación y puesta en marcha del software.

### **3.1 MODELO DE DESARROLLO**

El modelo de Prototipado Evolutivo involucra las fases de investigación preliminar, definición de los requerimientos del sistema, diseño, programación y prueba y evaluación.

Investigación Preliminar: Las metas de esta fase son: determinar el problema y su ámbito, la importancia y sus efectos potenciales sobre la organización, identificar una idea general de la solución para realizar un estudio de que determine la viabilidad de una solución software.

Definición de los requerimientos del sistema: El objetivo de esta etapa es registrar todos los requerimientos y deseos que los usuarios tienen con relación al proyecto. Para determinar así los requisitos mediante la construcción, demostración y retroalimentación del prototipo.

Diseño: Durante la construcción del prototipo, se evitó el diseño detallado. El sistema debe ser entonces rediseñado y documentado según los estándares de la organización para ayudar al mantenimiento. La fase de diseño técnico comprende la documentación que especifica y describe la estructura del

software, el control de flujo, las interfaces de usuario y las funciones, además, la producción de todo lo requerido para promover cualquier mantenimiento futuro del software.

**Programación y Prueba:** Esta fase conlleva los cambios identificados en el diseño, los cuales son implementados y probados para asegurar la corrección y completitud de los mismos con respecto a los requerimientos.

**Evaluación:** Esta etapa consiste en extraer de los usuarios la especificación de los requerimientos adicionales del sistema y verificar si el prototipo desarrollado es acorde con la definición de requerimientos del sistema. Si los usuarios identifican fallas en el prototipo, el desarrollador lo corrige antes de la siguiente evaluación. El prototipo es modificado y evaluado hasta que todos los requerimientos del sistema han sido satisfechos<sup>24</sup>.

### **3.2 LENGUAJE DE MODELADO**

El lenguaje de modelado UML hace posible expresar, de forma gráfica, la interacción de los distintos componentes del Sistema de Información desarrollado.

UML es una especificación de notación orientada a objetos. Se basa en las anteriores especificaciones BOOCH, RUMBAUGH y COAD-YOURDON. Divide cada proyecto en un número de diagramas que representan las posibles vistas del proyecto. Estos diagramas representan la arquitectura del proyecto.

Con UML se debe olvidar el protagonismo excesivo dado al diagrama de clases, este representa una parte importante del sistema, es que el sistema estaba sentado. Se sabe su estructura pero no se sabe que le sucede a sus diferentes partes cuando el sistema empieza a funcionar. UML introduce diagramas que representan una visión dinámica del sistema. Gracias al diseño de la parte dinámica del sistema es posible observar, en la fase de diseño, los

---

<sup>24</sup> Tomado de <http://html.rincondelvago.com/desarrollo-orientado-a-prototipos.html>

problemas de la estructura al propagar errores o de las partes que necesitan ser sincronizadas, así como del estado de cada instancia en todo momento. El diagrama de clases continua siendo importante, más hay que tener presente que su representación es limitada, y que ayuda a diseñar un sistema robusto con partes reutilizables, pero no a solucionar problemas de propagación de mensajes ni de sincronización o recuperación ante estados de error. En resumen, un sistema debe estar bien diseñado, pero también debe funcionar de manera correcta.

UML intenta solucionar el problema de propiedad de código que se da con los desarrolladores, al implementar un lenguaje de modelado común para todos los desarrollos se crea una documentación común, que cualquier desarrollador con conocimientos de UML será capaz de entender, independiente del lenguaje utilizado para el desarrollo.

UML es ahora un estándar, no existe otra especificación de diseño orientado a objetos, es el resultado de las tres opciones existentes en el mercado. Su uso es independiente del lenguaje de programación y de las características de los proyectos, UML ha sido diseñado para modelar todo tipo de proyectos, tanto informáticos como de arquitectura, o de otra rama.

UML permite la modificación de todos sus miembros mediante estereotipos y restricciones. Un estereotipo indica las especificaciones del lenguaje al que se refiere el diagrama de UML. Una restricción identifica un comportamiento forzado de una clase o relación, es decir mediante la restricción se fuerza el comportamiento que tiene el objeto al que se le aplica.

La explicación se basará en los diagramas. Cada diagrama usa la anotación pertinente y la suma de estos diagramas crean las diferentes vistas. Las vistas existentes en UML son:

Vista casos de uso: Se forma con los diagramas de casos de uso, colaboración, estados y actividades.

Vista de diseño: Se forma con los diagramas de clases, objetos, colaboración, estados y actividades.

Vista de procesos: Se forma con los diagramas de la vista de diseño. Recalcando las clases y objetos referentes a procesos.

Vista de implementación: Se forma con los diagramas de componentes, colaboración, estados y actividades.

Vista de despliegue: Se forma con los diagramas de despliegue, interacción, estados y actividades.

Se dispone de dos tipos diferentes de diagramas los que exhiben una vista estática del sistema y los que exhiben una visión dinámica. Aparte los diagramas estáticos son:

Diagrama de clases: muestra las clases, interfaces, colaboraciones y sus relaciones. Son más comunes y proveen una vista estática del proyecto.

Diagrama de objetos: Es un diagrama de instancias de las clases mostradas en el diagrama de clases. Muestra las instancias y como se relacionan entre ellas.

Diagrama de componentes: Muestran la organización de los componentes del sistema. Un componente se corresponde con una o varias clases, interfaces o colaboraciones.

Diagrama de despliegue: Muestra los nodos y sus relaciones. Un nodo es un conjunto de componentes. Se utiliza para reducir la complejidad de los

diagramas de clases y componentes de un gran sistema. Sirve como resumen e índice.

Diagrama de casos de uso: Muestran los casos de uso, actores y sus relaciones. Muestra quien puede hacer que y las relaciones que existen entre acciones (casos de uso). Son muy importantes para modelar y organizar el comportamiento del sistema.

Lo diagramas dinámicos son:

Diagrama de secuencia, Diagrama de colaboración: Muestran a los diferentes objetos, así como las relaciones presentes entre ellos, los mensajes que se envían. Son dos diagramas diferentes, que se pasan de uno a otro sin perdida de información, pero que proveen puntos de vista diferentes del sistema.

Diagrama de estados: muestra los estados, eventos, transiciones y actividades de los diferentes objetos. Son útiles en sistemas que reaccionen a eventos.

Diagrama de actividades: Es un caso especial del diagrama de estados. Muestra el flujo entre los objetos. Se utilizan para modelar el funcionamiento del sistema y el flujo de control entre objetos.

El número de diagramas es muy alto, en la mayoría de los casos excesivos, UML permite definir sólo los necesarios, no todos son necesarios en los proyectos<sup>25</sup>. El capítulo 4, hace referencia al diseño del sistema empleando éste sencillo lenguaje de modelado.

#### **4. ANALISIS DE REQUERIMIENTOS Y DISEÑO GLOBAL DEL SISTEMA**

---

<sup>25</sup> Tomado de <http://www.programacion.net/tutorial/uml/2/>

## **4.1 RECOLECCIÓN DE REQUERIMIENTOS**

Para la recopilación de requerimientos se definen las entidades y personas relacionadas con la información que debe manejar el sistema, para ello se realizaron entrevistas con los encargados de la administración y funcionamiento de las salas de la escuela. Además, se observó la manera como se lleva a cabo el proceso de ingreso y control de uso de dichas salas, para conocer a fondo los requerimientos que el sistema requiere.

A partir de la información recopilada y las entrevistas realizadas se definen los siguientes requerimientos clasificados así:

### **4.1.1 REQUERIMIENTOS DE LA DIRECCIÓN DE ESCUELA**

- Planificar, controlar y evaluar el uso de las salas de desarrollo de software.
- Disminuir costos y/o tiempos en el desarrollo de las diferentes actividades inherentes a la sala de desarrollo de software empleando dispositivos hardware.
- Brindar fácil acceso a las salas de desarrollo de software
- Obtener información en tiempo real sobre las zonas en funcionamiento y sobre los usuarios que ingresaron y se encuentran actualmente en el área.
- Presentar informes estadísticos sobre el tiempo de uso de las salas, en determinados periodos de tiempo.

### **4.1.2 REQUERIMIENTOS DEL ADMINISTRADOR DE LA SALA**

- Mediante una señal de alerta ser informado del estado de la puerta de acceso a la sala. Si se encuentra abierta, sin llevar a cabo el debido proceso, se emitirá una señal sonora.
- Manejar los procesos de administración de la información relacionada con las zonas, los grupos y los usuarios autorizados para utilizar los recursos de la sala de desarrollo de software.
- Agilizar el trabajo de las personas encargadas de las salas

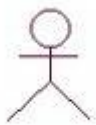
#### 4.1.3 REQUERIMIENTOS DE LOS USUARIOS

- Obtener fácil acceso a las salas de desarrollo de software
- Permitir el ingreso a las salas de desarrollo de software en cualquier momento (siempre y cuando el usuario respete su horario) y sin depender de terceros (auxiliares).

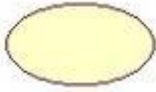



#### 4.2 DISEÑO GLOBAL DEL SISTEMA

A continuación, se describe la manera como se construyó cada uno de los módulos, para cumplir a cabalidad la labor de SCAAR. Para diseñar el Sistema se empleó el Lenguaje Unificado de Modelado UML, enfocado en los diagramas de casos de uso y diagramas de actividades.

Para la realización de los diagramas se utilizaron los siguientes elementos, los cuales se muestran en la tabla<sup>26</sup>

NOMBRE	SIGNIFICADO	SIMBOLO
Actor	Es un rol que un usuario juega con respecto al sistema. Un Actor no necesariamente representa a una persona en particular, sino la labor que realiza	

<sup>26</sup> Tomado de <http://softtlan.blogspot.com/>

	frente al sistema.	
Caso de Uso	Es una operación/tarea específica que se realiza tras una orden de algún agente externo, sea desde una petición de un actor o bien desde la invocación desde otro caso de uso.	
Asociación	Es el tipo de relación más básica que indica la invocación desde un actor o caso de uso a otra operación (caso de uso).	
Dependencia o Instanciación	Es una forma muy particular de relación entre clases, en la cual una clase depende de otra, es decir, se instancia (se crea). Dicha relación se denota con una flecha punteada.	
Generalización	Este tipo de relación es uno de los más utilizados, cumple una doble función dependiendo de su estereotipo, que puede ser de Uso (<<uses>>) o de Herencia (<<extends>>). Este tipo de relación esta orientado exclusivamente para casos de uso (y no para actores).	

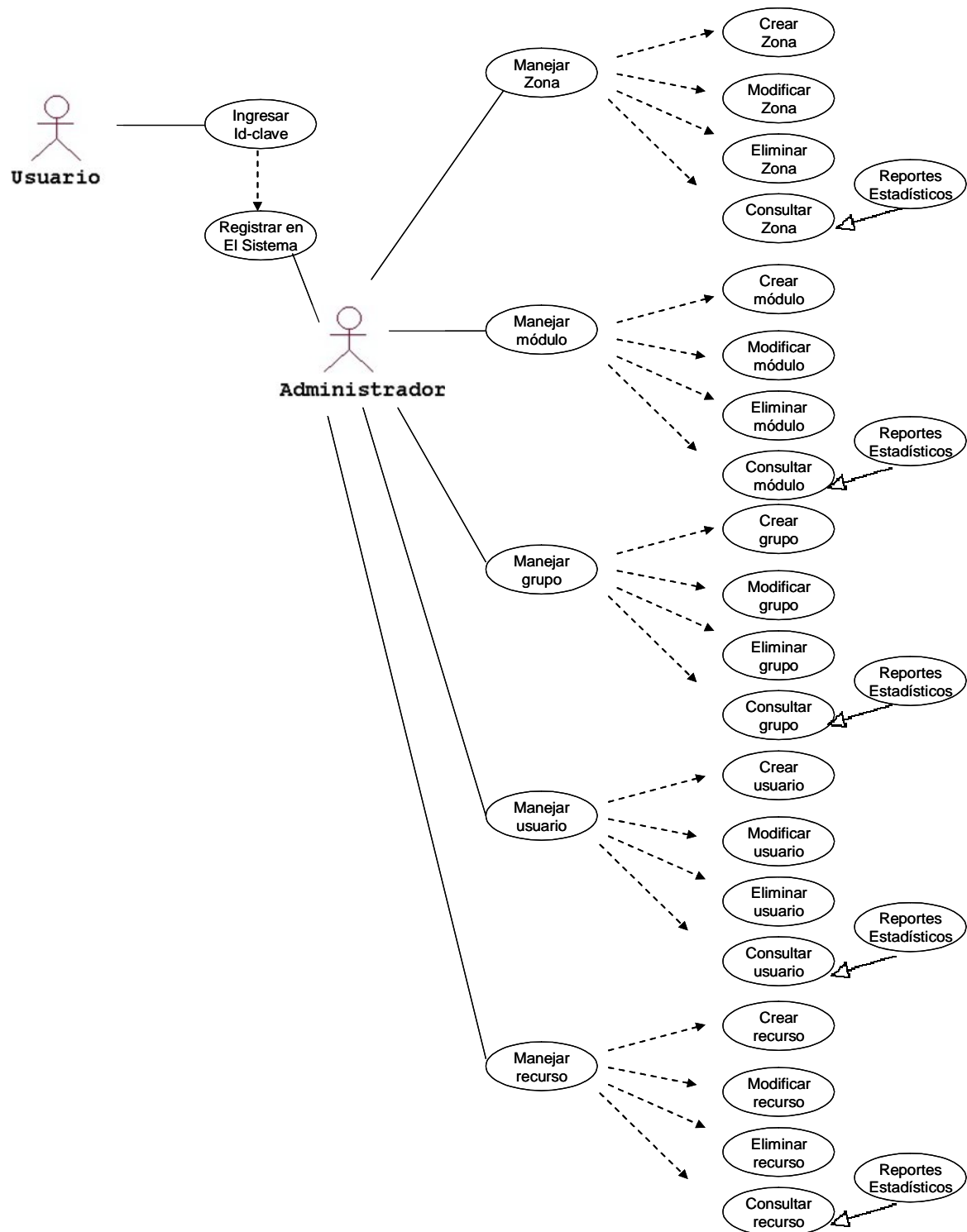
**Tabla 3 . Elementos de Diagramas UML utilizados en el Diseño.**

#### 4.2.1 Diagramas de Casos de Uso

Los diagramas de casos de uso sirven para especificar la funcionalidad y el comportamiento de un sistema mediante su interacción con los usuarios y/o otros sistemas<sup>27</sup>. Un Caso-Usado está muy relacionado con lo que pudiera ser considerado un escenario en el sistema, esto es, lo que ocurre cuando alguien interactúa con el sistema<sup>28</sup>.

<sup>27</sup> Tomado de [http://es.wikipedia.org/wiki/Caso\\_de\\_uso](http://es.wikipedia.org/wiki/Caso_de_uso)

<sup>28</sup> Tomado de <http://www.osmosislatina.com/lenguajes/uml/casos.htm>



**Figura 10. Diagrama General Caso Uso**

En la elaboración de casos de uso se tiene en cuenta la interacción de los actores con el caso de uso y la forma excepcional en que el actor puede interactuar con dicho elemento, es decir, representa las acciones que realiza

cada tipo de usuario del sistema, tales usuarios son: Administrador de la Sala, Auxiliares y Usuarios en general (directores de grupo, estudiantes, visitantes).

### Caso de uso para el Administrador de la Sala

- **Caso de Uso Crear Entidades (zona, modulo, grupo, usuario, recurso)**

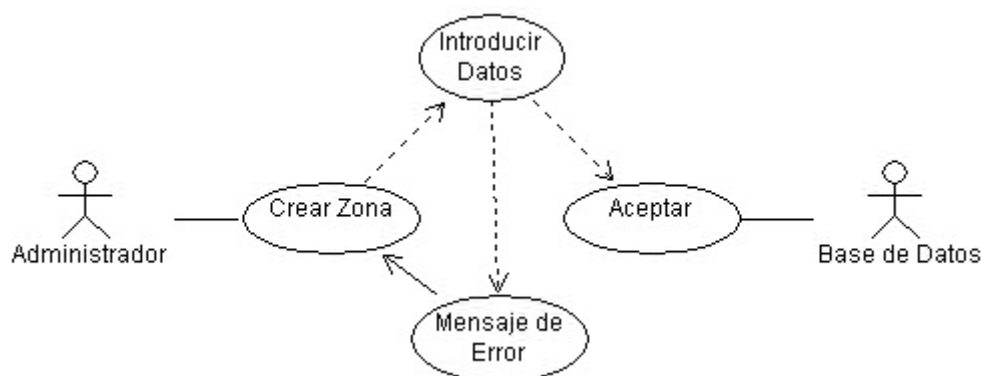
El Administrador tendrá acceso a un formulario en el cual se crea oficialmente una nueva zona, un nuevo modulo, grupo, usuario o recurso. Es importante tener en cuenta que las zonas están divididas por módulos, donde cada módulo esta conformado por grupos y estos a su vez por estudiantes.

Como se muestra en las Figuras 11 al 15, en estos casos de uso se incluyen las siguientes operaciones:

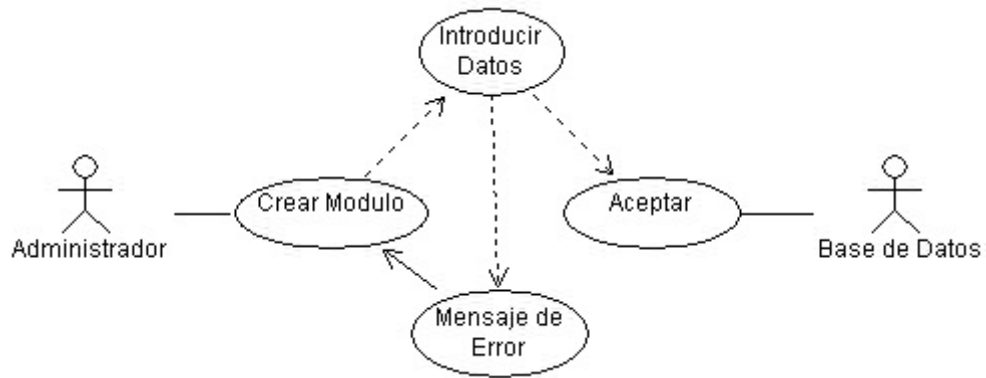
Introducir Datos: El administrador incluirá los datos respectivos.

Aceptar: El administrador guarda los datos correspondientes en la base de datos.

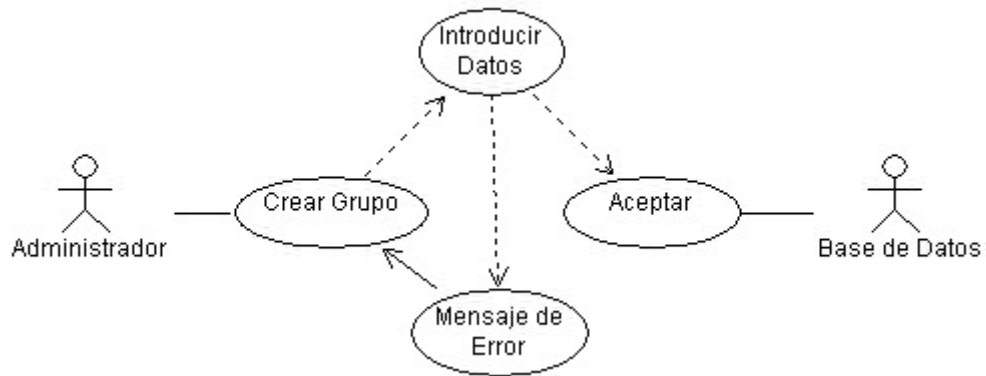
Mensaje de Error: En caso de no haber llenado los campos necesarios para el registro se generara la advertencia de que existen campos vacíos o datos no válidos.



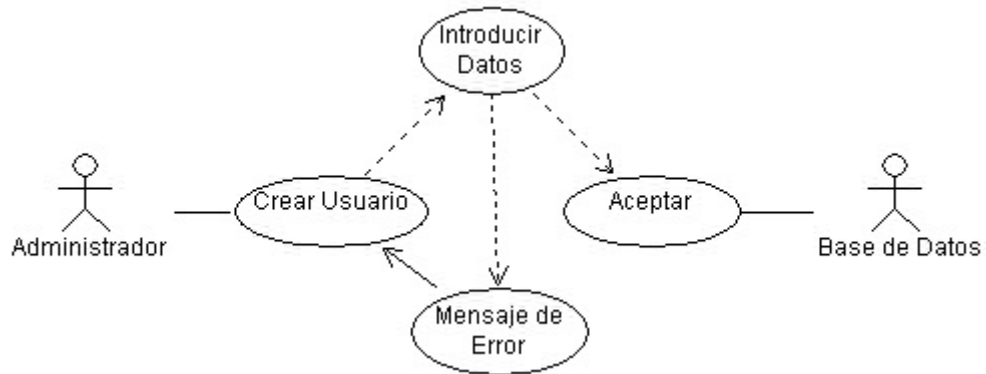
**Figura 11. Caso de Uso Nueva Zona**



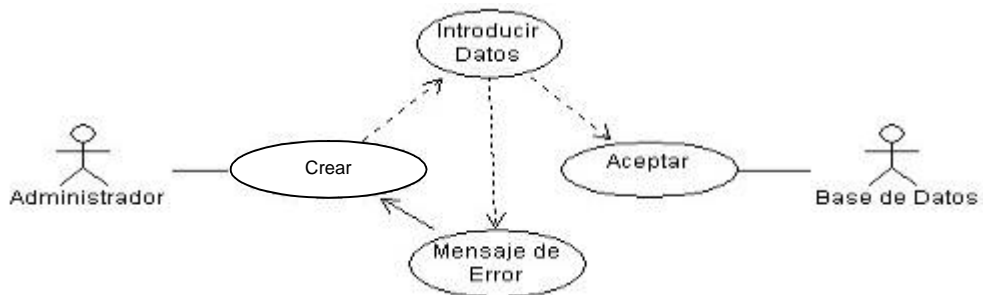
**Figura 12. Caso de Uso Nuevo Módulo**



**Figura 13. Caso de Uso Nuevo Grupo**



**Figura 14. Caso de Uso Nuevo Usuario**



**Figura 15. Caso de Uso Nuevo Recurso**

- **Caso de Uso Actualizar Entidades (zona, modulo, grupo, usuario, recurso)**

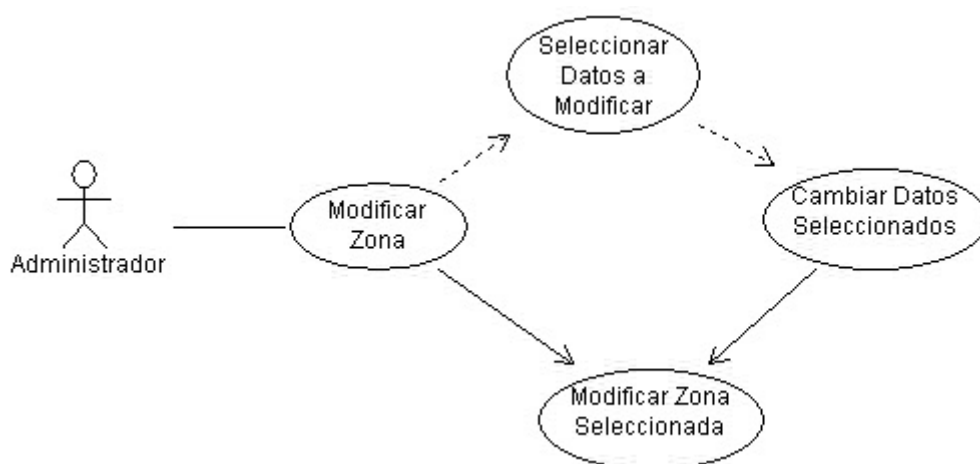
El administrador podrá modificar datos necesarios para llevar un control eficiente sobre el área restringida.

En los siguientes casos de uso correspondientes a las figuras 16 a 20, se incluyen las siguientes operaciones:

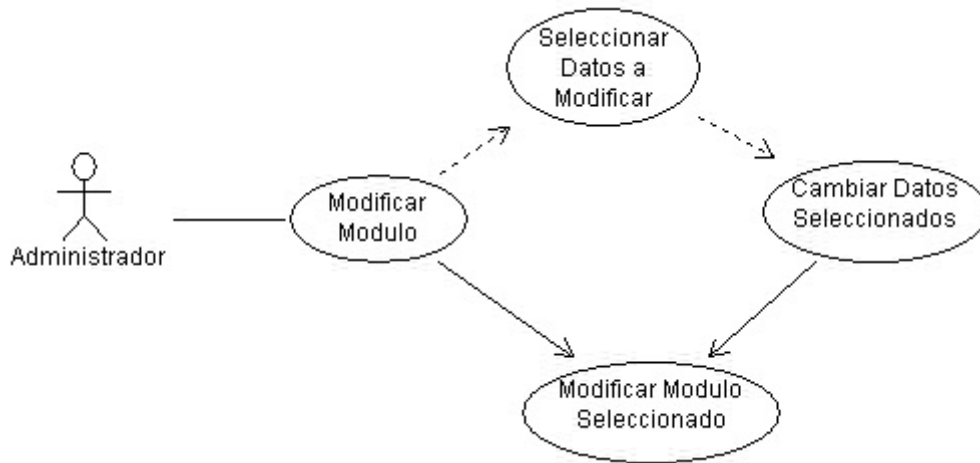
Seleccionar dato a modificar: Tendrá acceso a los datos de la base de datos del sistema por medio de los formularios los que previamente había utilizado para llenar los registros y así seleccionar el dato que desea modificar.

Llenar dato a modificar: El administrador cambiará los datos que considere necesarios llenando el campo al cual corresponde dicho dato.

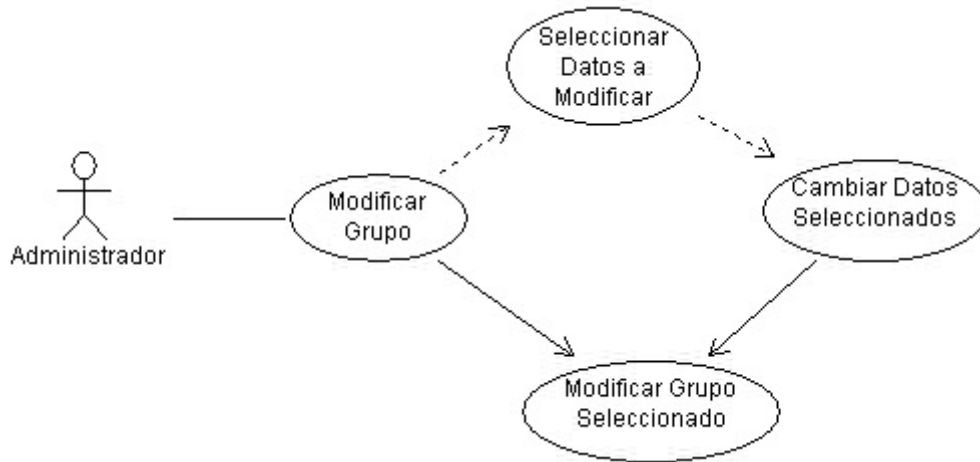
Aceptar: El administrador guarda estos datos que serán actualizados en la base de datos.



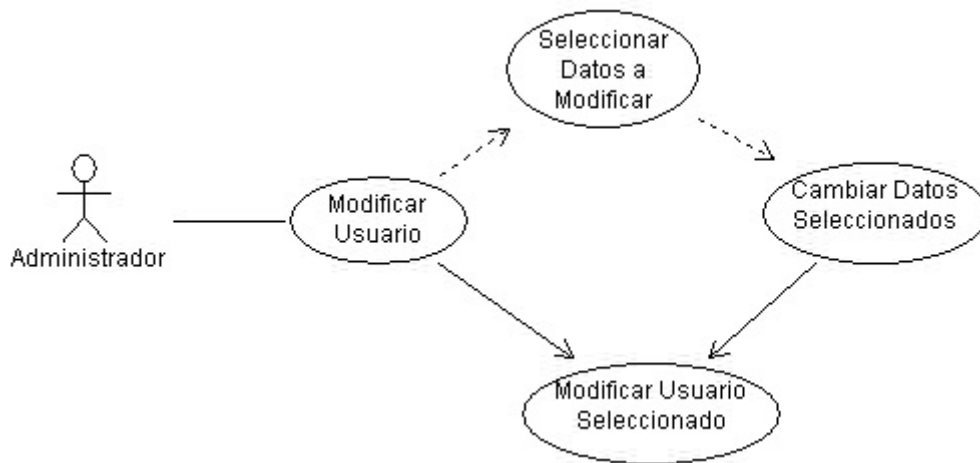
**Figura 16. Caso de Uso Actualizar Zona**



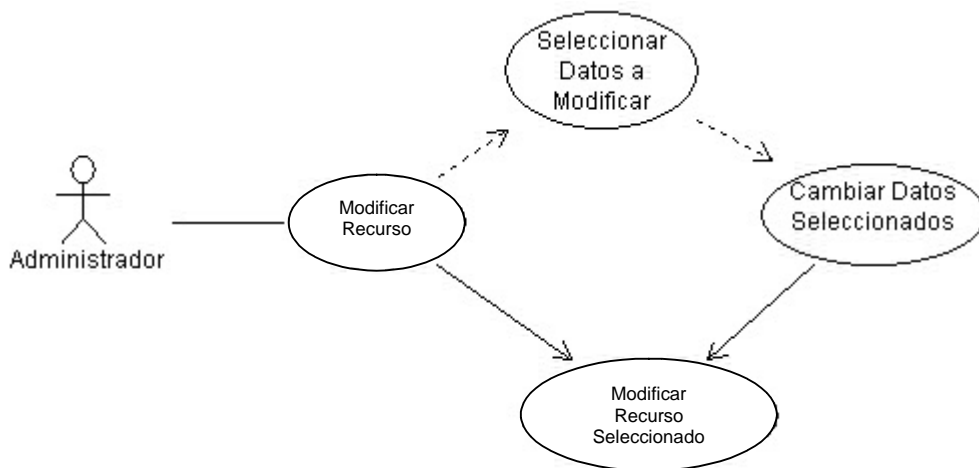
**Figura 17. Caso de Uso Actualizar Módulo**



**Figura 18. Caso de Uso Actualizar Grupo**



**Figura 19. Caso de Uso Actualizar Usuario**



**Figura 20. Caso de Uso Actualizar Recurso**

- **Caso de Uso Consultar Entidades (zona, modulo, grupo, usuario, recurso)**

En este caso de uso el administrador podrá llevar a cabo operaciones como: realizar consultas o generar reportes estadísticos, según las necesidades de la dirección de escuela de Ingeniería de Sistemas o según sus propias necesidades para la administración del laboratorio Jose A. Villabona.

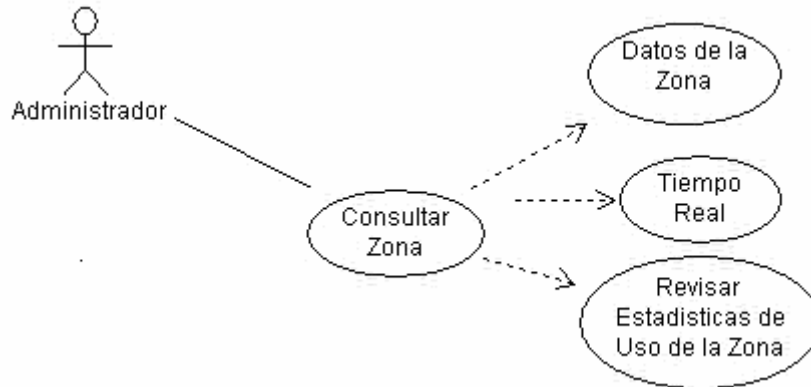
En los casos de uso referentes a la Figura 21 a 25, se presentan las siguientes operaciones:

**Realizar Consulta:** El administrador puede visualizar el resultado de la consulta deseada, ya sea solicitando todo el contenido de la base de datos o filtrando la información según los parámetros de tiempo para los informes estadísticos.

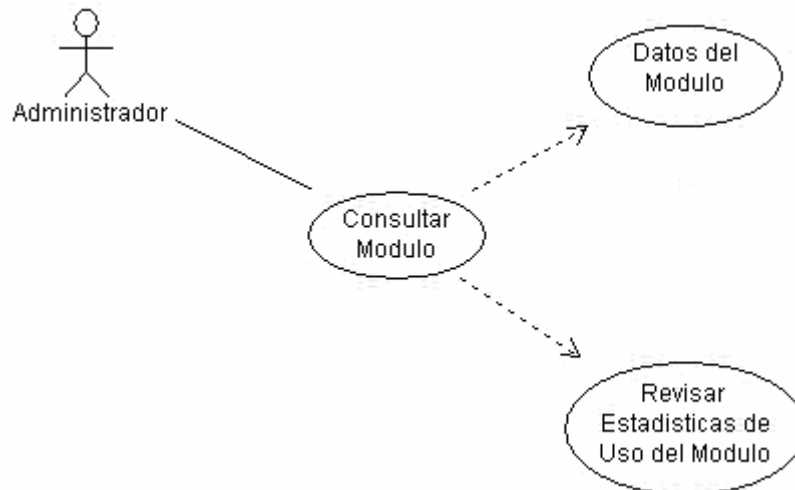
**Seleccionar Parámetro de Consulta:** El administrador elegirá el intervalo de tiempo deseado para conocer el uso que se ha dado al laboratorio en dicho periodo.

**Realizar Consultas en Tiempo Real:** El administrador tendrá la opción de

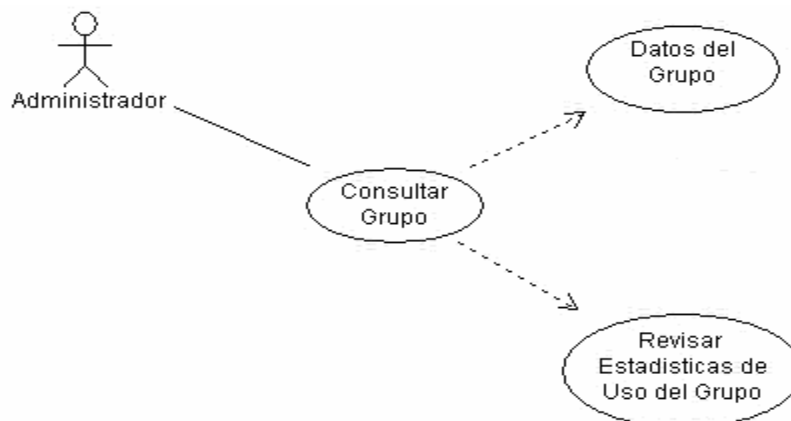
obtener información en tiempo real sobre las zonas que están en funcionamiento y sobre los usuarios que ingresaron y se encuentran actualmente en el área.



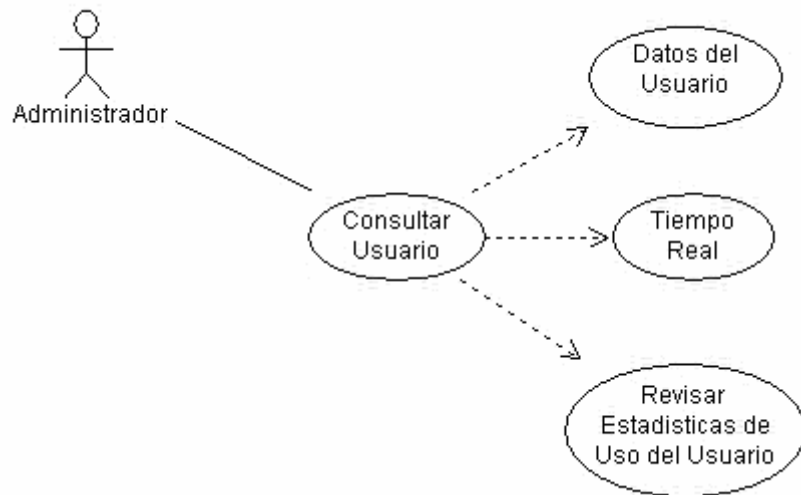
**Figura 21. Caso de Uso Consultar Zona**



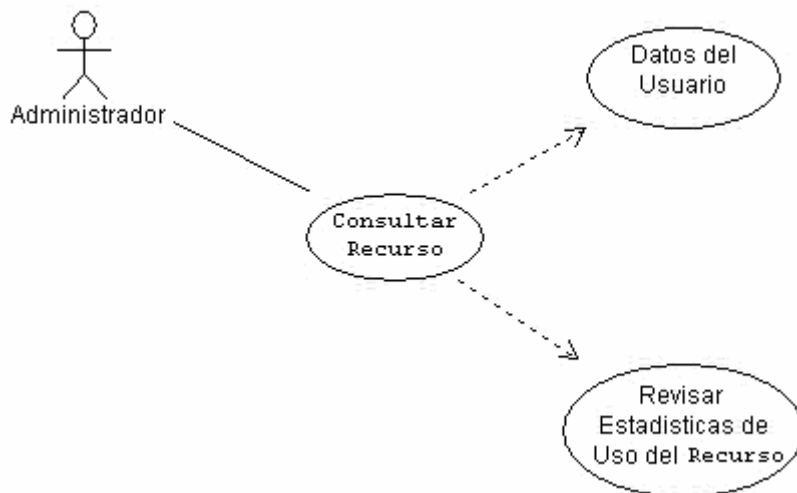
**Figura 22. Caso de Uso Consultar Módulo**



**Figura 23. Caso de Uso Consultar Grupo**



**Figura 24. Caso de Uso Consultar Usuario**



**Figura 25. Caso de Uso Consultar Recurso**

- **Caso de Uso Eliminar Entidades (zona, modulo, grupo, usuario)**

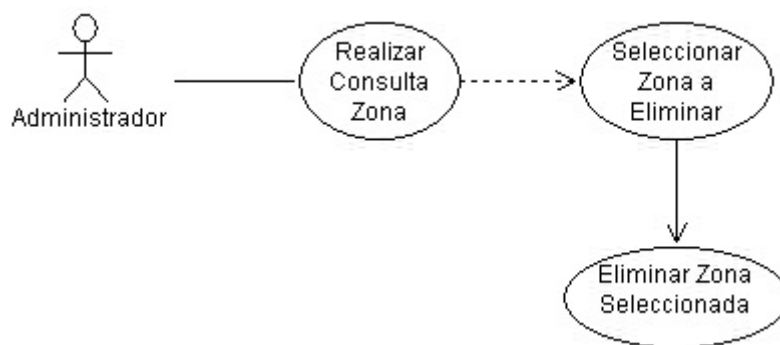
A través del listado de resultados generado al realizarse una consulta, el administrador tendrá la opción de seleccionar cualquier número de registros

para ser eliminados de la base de datos. En los siguientes casos de uso (figuras 26 a 30) se presentan las siguientes operaciones:

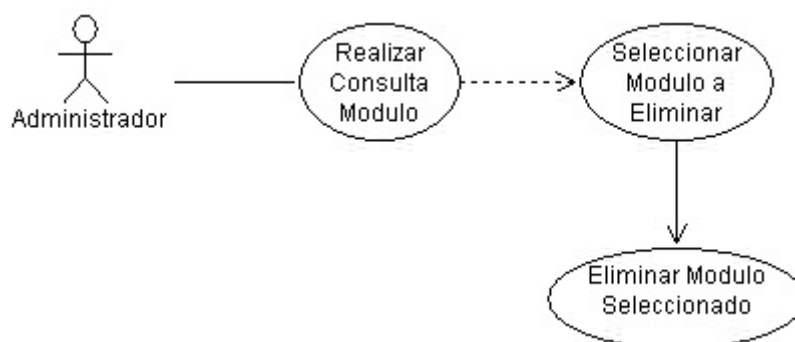
**Realizar Consulta:** El administrador puede visualizar el resultado de la consulta deseada, ya sea solicitando todo el contenido de la base de datos o filtrando la información según los parámetros de uso de sala para los informes estadísticos.

**Seleccionar Registro(s) a eliminar:** El administrador elegirá el registro o registros que desee sean eliminados de la base de datos.

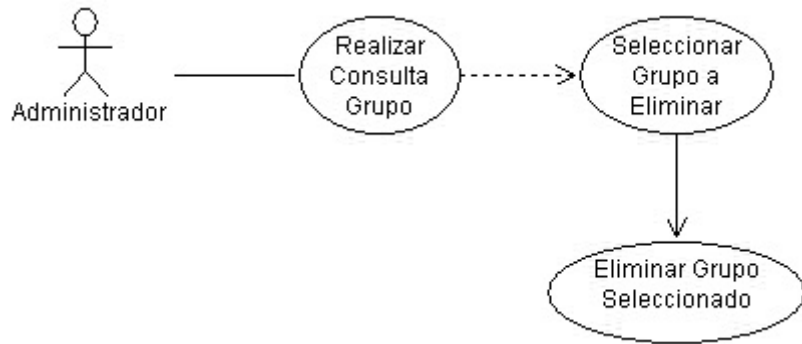
**Aceptar:** El administrador elimina los registros seleccionados.



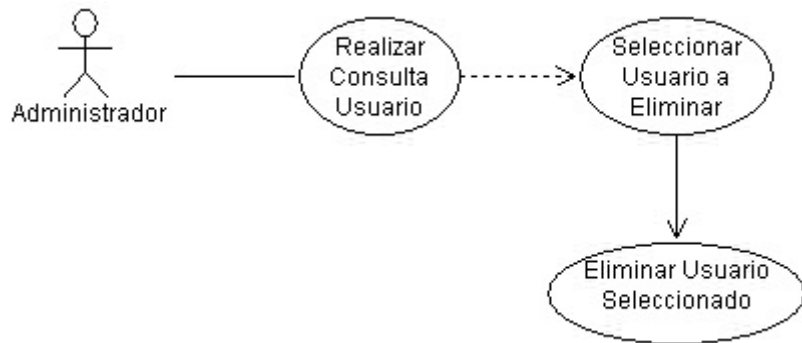
**Figura 26. Caso de Uso Eliminar Zona**



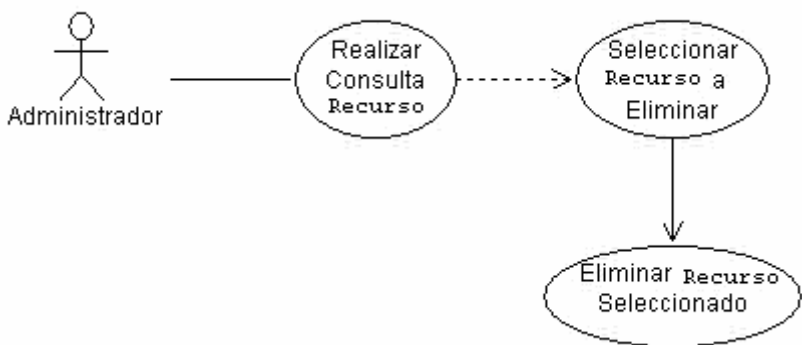
**Figura 27. Caso de Uso Eliminar Módulo**



**Figura 28. Caso de Uso Eliminar Grupo**



**Figura 29. Caso de Uso Eliminar Usuario**



**Figura 30. Caso de Uso Eliminar Recurso**

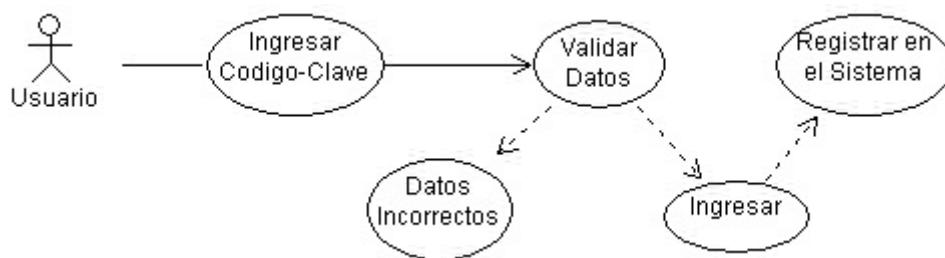
## Caso de uso para el usuario

- **Caso de uso Ingresar**

El acceso al área restringida se realiza con las siguientes operaciones:

Introducir datos: El usuario introducirá los datos requeridos para el ingreso (código, clave), a través del teclado ubicado en la entrada de la sala.

Validación de Datos: Es un procedimiento de control para el acceso al sistema, es decir, el sistema hará la validación correspondiente para permitir o no el ingreso a la sala.



**Figura 31. Caso de Uso Ingresar Usuario**

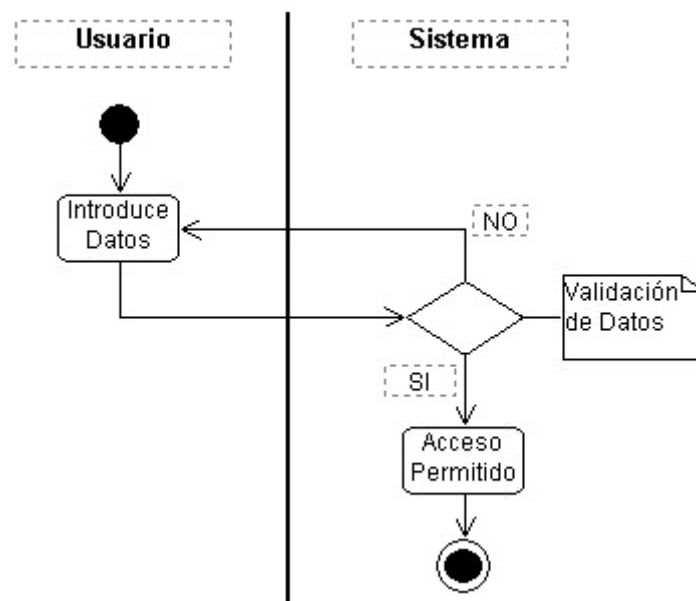
### 3.2.1 Diagramas de Actividades

Un diagrama de Actividad demuestra la serie de actividades que deben ser realizadas en un caso-usuario, así como las distintas rutas que pueden irse desencadenando en el caso-usuario. Un diagrama de actividad es utilizado en conjunción de un diagrama caso-usuario para auxiliar a los miembros del equipo de desarrollo a entender cómo es utilizado el sistema y cómo reacciona en determinados eventos.

Los diagramas de actividades muestran el orden en el que se van realizando tareas dentro de un sistema (el flujo de control de las actividades)<sup>29</sup>.

- **Diagrama de Actividades Ingresar Usuario**

En el momento en que el usuario desee ingresar al área, debe digitar (en el dispositivo hardware ubicado en la entrada de la sala) su código de usuario y su clave. Al hacerlo, el sistema chequea que los valores suministrados sean los correctos, que dicho usuario exista y que la zona se encuentre activa y dentro de horarios de funcionamiento. Si los datos suministrados son los que se encuentran registrados en la base de datos para dicho id de usuario, el seguro de la puerta se desactiva permitiendo la entrada al usuario. De caso contrario el seguro no se desactiva y el usuario no puede ingresar al área. Este diagrama puede apreciarse en la siguiente figura:



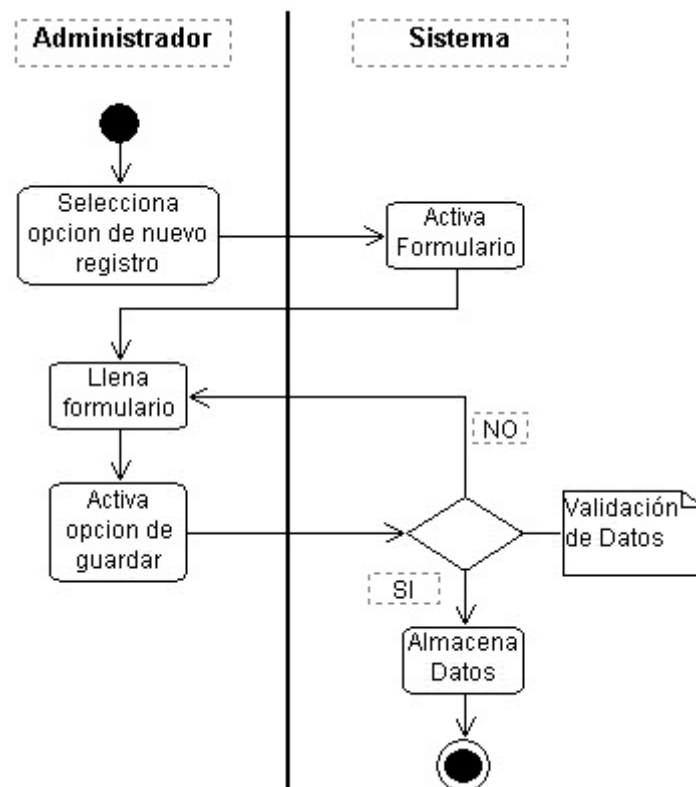
**Figura 32. Diagrama de Actividades Ingresar Usuario**

<sup>29</sup> Tomado de <http://elvex.ugr.es/decsai/java/pdf/3E-UML.pdf>

- **Diagrama de Actividades Nuevo Registro**

En la figura 33 se detallan las actividades que realizan el administrador y el sistema para crear un nuevo registro (de zona, módulo, grupo, usuario, recurso) en la base de datos.

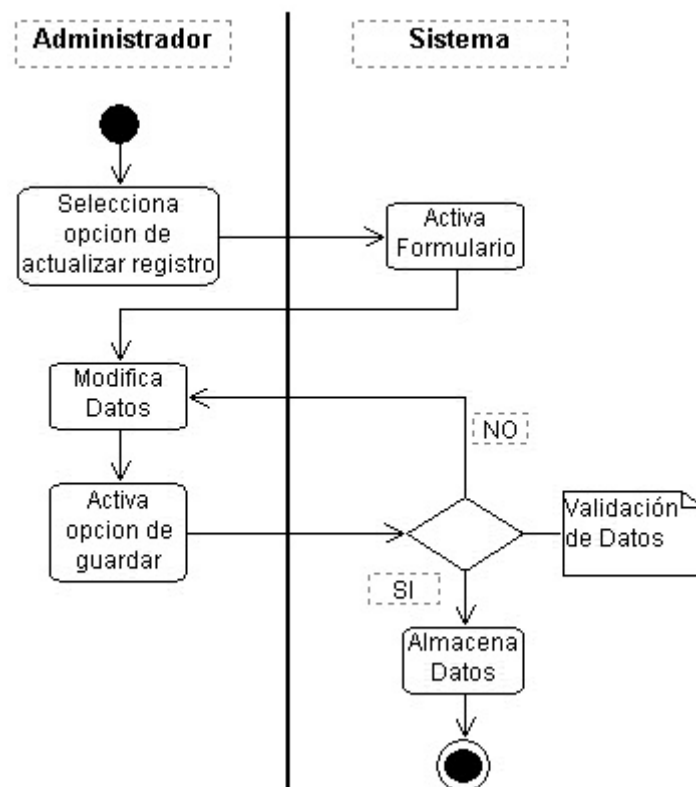
Para crear un nuevo registro se ingresan los datos en el formulario correspondiente y se da aceptar para que el sistema los valide y solicite correcciones de ser necesarias; de no serlas, se crea el nuevo registro en la Base de Datos.



**Figura 33. Diagrama de Actividades Nuevo Registro**

- **Diagrama de Actividades Actualizar Registro**

El administrador elige la opción en la que se encuentra el registro que desea actualizar (zona, modulo, grupo, usuario, recurso). El sistema carga los datos correspondientes para que el administrador localice los campos a modificar, llene el formulario con los nuevos datos y acepte los cambios realizados para actualizar los datos en la Base de Datos.

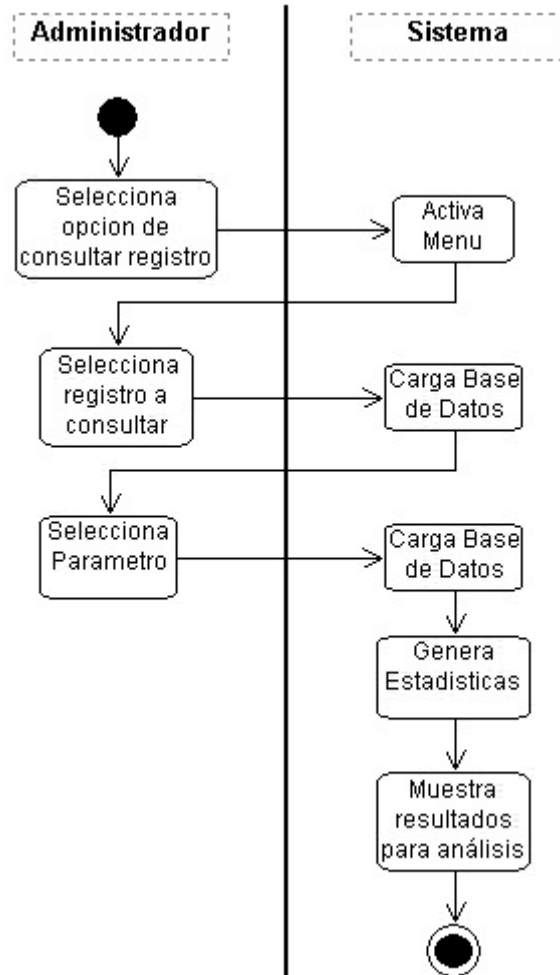


**Figura 34. Diagrama de Actividades Actualizar Registro**

- **Diagrama de Actividades Consultar Registro**

En el proceso de consulta el administrador puede visualizar todos los registros almacenados en la base de datos (que sean de importancia para la administración), donde podrá elegir el parámetro de consulta (zona, módulo,

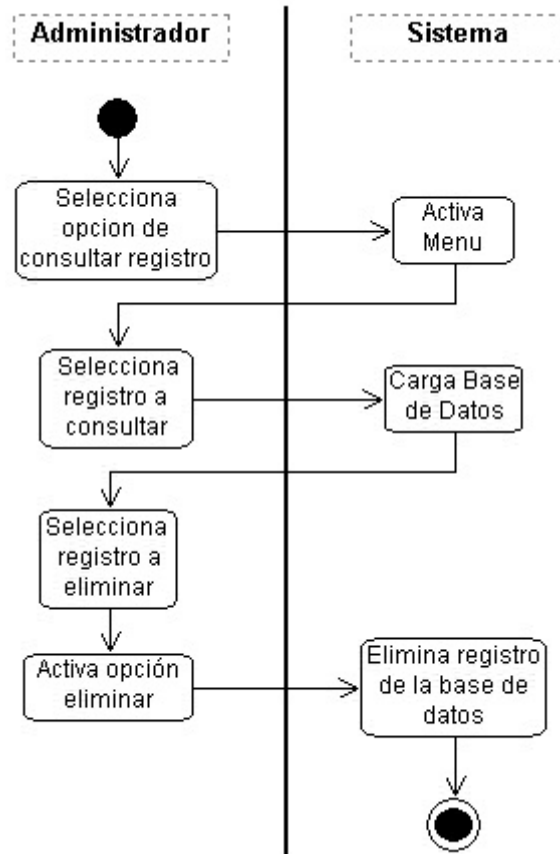
grupo, usuario, recurso) para los periodos de tiempo establecidos (fecha inicial de consulta y fecha final de consulta) y así generar informes estadísticos para un posterior análisis.



**Figura 35. Diagrama de Actividades Consultar Registro**

- **Diagrama de Actividades Eliminar Registro**

Dentro del proceso de consulta se puede llevar a cabo la eliminación de registros seleccionando del listado que se genera, después de realizar alguna consulta, los registros a eliminar y presionando el botón Eliminar.



**Figura 36. Diagrama de Actividades Eliminar Registro**

### 4.3 DISEÑO DETALLADO

Por medio de un estudio detallado de los distintos casos de uso y diagramas de actividades para el sistema y siguiendo el modelo de prototipado evolutivo, se plantean los diferentes prototipos a seguir para el desarrollo del sistema software.

#### 4.3.1 Prototipo Básico Módulo de Administración

En este prototipo se planteó una visión global de usuarios cuyo propósito principal fuese una apropiada y eficiente administración de la información de las personas que interactúan con las salas de desarrollo de software. Por esto,

para iniciar en la interfaz grafica de la herramienta java, se diseñó una ventana principal conformada por una barra de menú a través de la cual se accedía a cada una de las opciones generales como: crear, eliminar, actualizar y consultar la información relacionada de las zonas, módulos, grupos, recursos y usuarios autorizados para utilizar los recursos del laboratorio José A. Villabona.

Para cada opción se creó el formulario correspondiente, por medio del cual el administrador podría:

- Introducir los datos necesarios para crear nuevas zonas, módulos, grupos, recursos y/o usuarios.
- Eliminar registros.
- Modificar la información almacenada en la Base de Datos.
- Consultar toda la información (de importancia para la administración) relacionada con las zonas, módulos, grupos, recursos y usuarios creados.

Con estos datos y con el programa desarrollado para establecer la comunicación entre la computadora local y el microcontrolador, se implementó el procedimiento para controlar el acceso de las personas autorizadas a ingresar a la sala, el cual consistía en introducir el código y la clave a través de un teclado ubicado en la entrada del área, datos que eran enviados a la computadora local para ser validados.

De igual manera para los informes estadísticos se implementó el procedimiento para calcular el tiempo que dura cada usuario dentro del área, basado en la hora de ingreso y de salida de cada persona.

Este prototipo tiene como fin aclarar dudas con el usuario sobre lo que podría ser la base de evolución para el producto final sin perder de vista todos sus objetivos, definir los requerimientos hardware y software básicos para la implementación.

### **4.3.2 Prototipo Básico Modulo del Microcontrolador**

En este prototipo se planteó una visión global de las funciones a realizar el PIC, como lo era: comunicación con el software de la computadora local para enviar o recibir datos, detectar si la puerta se encuentra o no cerrada, activar zumbador cada vez que se oprimía una tecla, activar la alarma cuando se detectara la apertura de la puerta sin haber realizado el debido proceso, y atender el teclado, usado para introducir el código y la clave. En este primer prototipo se implementó esta aplicación con un teclado, utilizado para llevar el registro de entrada y de salida de la sala.

### **4.3.3 Prototipo de Interfaces**

El desarrollo de este prototipo se realizó sobre los prototipos anteriores y basándose en los requerimientos previamente obtenidos y organizados en los diagramas de actividades y casos de uso. El propósito era mostrar las ventanas, su navegación, interacción, controles y botones al usuario y obtener una realimentación que permitiera mejorar el Diseño de Interfaz respecto a la cantidad de información sobre la pantalla, colores, organización y otros aspectos, usando controles y herramientas de Java. Con los diagramas de actividades se realizó la interfase entre el administrador y la pantalla de visualización, en esta se presentó la información como va a ser proporcionada al sistema o como es recuperada desde éste.

### **4.3.4 Prototipo de Control**

El objetivo de este prototipo es suministrar al administrador control y seguridad en cada uno de los procesos y actividades que realiza el sistema. El desarrollo de este prototipo se realizó sobre los prototipos anteriores y basándose en los requerimientos establecidos. La finalidad de este segundo prototipo fue la de implementar dos teclados, uno ubicado fuera de la sala y el otro dentro de la

sala. Con el objetivo de llevar control sobre el tiempo de uso del área, registrando en el sistema la hora en que ingresa cierto usuario y la hora en que sale; y así generar los informes estadísticos basados en el tiempo de permanencia de los usuarios en la sala sobre el tiempo total disponible.

Para calcular el tiempo total de servicio de la sala, es decir, el número de horas diarias que dura disponible la sala, se añadió una función que le permite al administrador establecer el horario de uso de las mismas. Cabe aclarar que el sistema sólo les permitirá a los usuarios acceder al área dentro del horario establecido para las salas.

En este prototipo el PIC atiende dos teclados, donde el control sobre ellos se hace de la siguiente manera: en el momento en que una tecla es oprimida, el microcontrolador sabe de que teclado se trata e inmediatamente deja de atender el otro, esto con el fin de no generar conflicto.

Como última modificación, se le permitió al administrador registrar el horario de uso de la sala para cada usuario teniendo como única condición que cada usuario usara uno y solo un computador del área y dicho computador fuera usado un sólo un usuario a una hora determinada. Con esto se permitió que el administrador determinará el uso de los recursos del área para que tomara medidas de asignación de equipos a quienes lo solicitaran pero basándose con soportes estadísticos que permitieran corroborar si el usuario que hace la solicitud requiere o no dicho computador (recurso). De igual forma, esta última modificación del sistema permitió que el administrador quitara equipos a usuarios que no estaban haciendo uso debido de los mismos y que no necesitaban (igualmente basado en el soporte estadístico). Todo esto permite que se aproveche en mayor medida los recursos del área, aumentando el número de personas que pueden usarla, y comprometiendo a los usuarios en el cumplimiento de los horarios definidos para cada uno, lo cual a su vez permite que no existan tiempos “muertos” de los equipos (por lo menos, no tan altos como anteriormente a la implementación del sistema) y que la sala esté en

constante funcionamiento. Si al pasar el tiempo se requiere de mayor recursos existirá un soporte estadístico que lo sustente. Las nuevas exigencias tendrán ahora soporte estadístico para cualquier periodo de tiempo (generado con los registros de la base de datos) previniendo conflictos innecesarios.

En esta refinación del prototipo final hubo características nuevas del sistema que no fueron contempladas previamente.

## 5. IMPLEMENTACIÓN

La metodología elegida para el desarrollo de SCAAR trae implícito una etapa de implementación basada en la revisión, innovación, sugerencias, reacciones iniciales y corrección de cada uno de los prototipos desarrollados. Ésta etapa se inicia con los resultados obtenidos en la etapa de diseño y guía las operaciones y funciones que debe realizar el software mediante código.

Nombre de la Herramienta:	<b>SCAAR</b>
Tamaño:	<b>3 MB</b>

### 5.1 MODELO DE DATOS

La organización de los datos es necesaria en el desarrollo del proyecto, la construcción de un modelo de datos asociado al manejo de la aplicación donde se almacenan los datos que controlan su funcionamiento se realizó de acuerdo a los parámetros requeridos para la administración de la información de las instancias relacionadas con la sala de desarrollo de software.

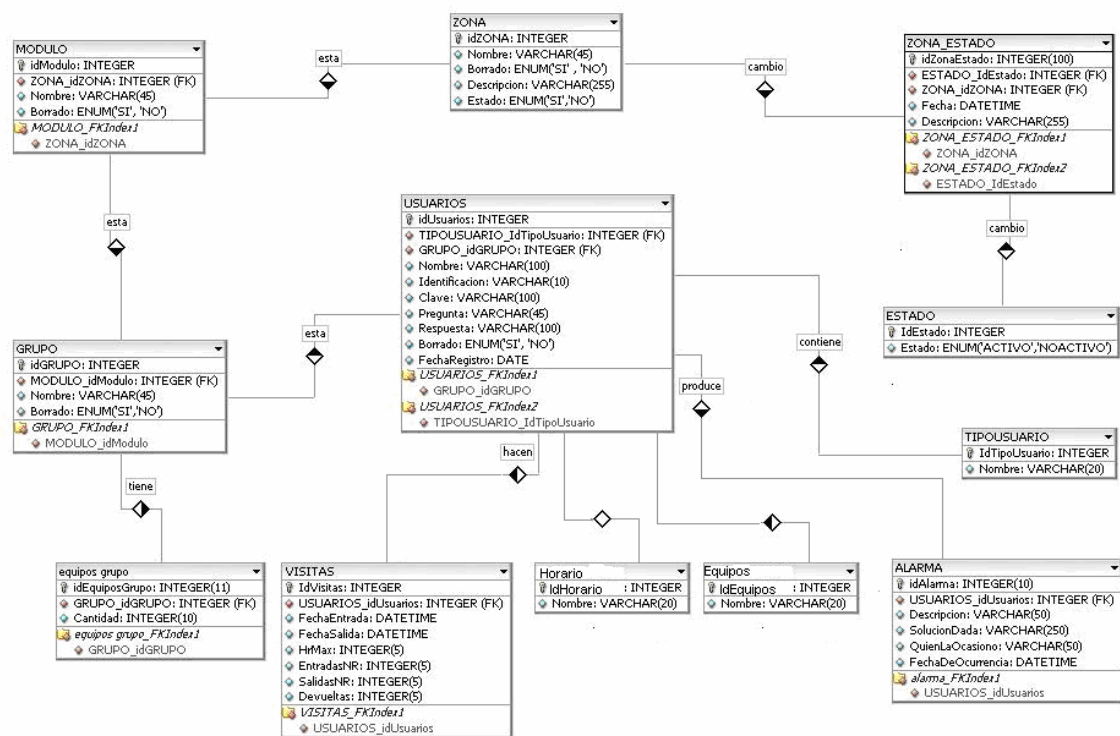
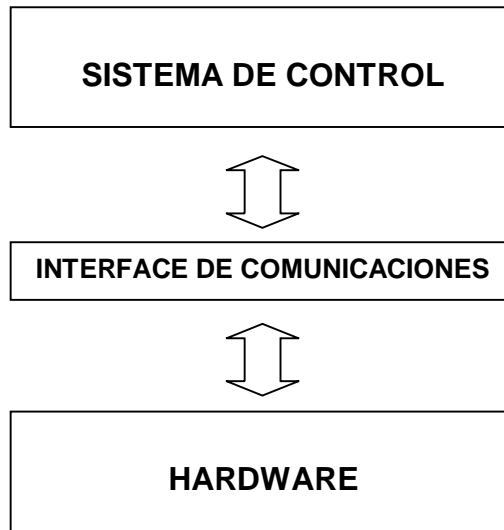


Figura 37. Modelo Entidad-Relación SCAAR

## 5.2 DESCRIPCIÓN GENERAL DE SCAAR

El Sistema de Control de Acceso a Áreas Restringidas (SCAAR), cuenta con un dispositivo hardware que esta en permanente comunicación con el software implementado en la PC Local (Administrador) para enviar y recibir datos que requiera el dispositivo hardware o que requiera el software. Por lo tanto SCAAR esta dividido en dos grandes partes: sistema de control (software) y dispositivo hardware.



**Figura 38. Descripción General de SCAAR**

### **Sistema de Control**

Es el encargado de:

- Guardar los datos
- Validar los datos introducidos a través del dispositivo hardware (identificación de acceso, clave de seguridad, señales de alarma).
- Procesar la información relacionada con los usuarios autorizados para utilizar los recursos del área.
- Dar información en tiempo real de los usuarios que ingresaron y que se encuentran actualmente en el área.
- Notificar al administrador del área de señales de alarma que se presenten (puerta abierta sin autorización, usuario forzado a ingresar clave para acceder al área, tiempo de espera de puerta abierta agotado, uso indebido de la clave acceso de algún usuario).
- Permitir el acceso de usuarios al área dependiendo del horario de las salas y del cumplimiento del horario por parte del usuario.
- Presentar informes estadísticos sobre el uso del laboratorio.

Este sistema se encuentra implantado en la computadora del administrador del área (PC Local) y se encuentra en permanente comunicación con el dispositivo hardware para enviar y recibir información necesaria para llevar un control eficiente sobre el área.

### **Interfase de Comunicaciones**

El puerto serial de las computadoras, conocido también como RS-232, es muy útil que permite la comunicación entre el software ubicado en la PC local y el dispositivo hardware.

### **Hardware**

El sistema consta de una tarjeta principal, donde se ubica el microcontrolador, y en la cual se hallan dispuestos los conectores que lo comunican con el teclado, los sensores, el zumbador y la interfaz de comunicaciones con la CPU local.

Esta parte se encarga de capturar datos requeridos por el Software de la computadora (identificación de acceso, clave de seguridad, señales de alarma), recibir datos, como órdenes (enviadas por el Software) de pedir clave de seguridad, activar/desactivar seguro de la puerta del área restringida, activar/desactivar alarma, activar/desactivar zumbador, efectuar acciones físicas, como activar-desactivar seguro de la puerta del área restringida, entre otras.

# DIAGRAMA GENERAL DE SCAAR

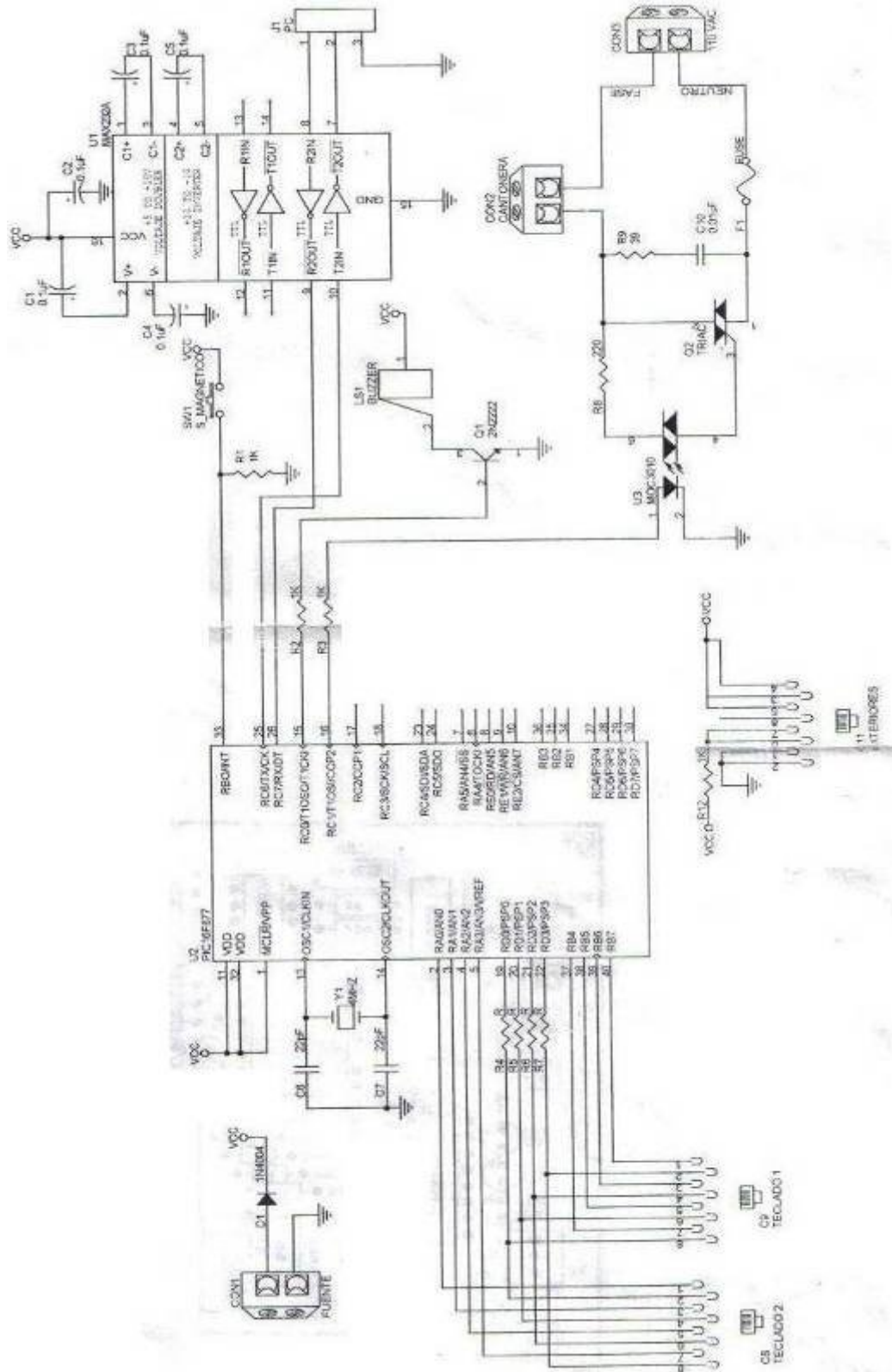


Figura 39. Diagrama General SCAAR

## ASPECTO EXTERNO

La interfaz con el usuario, consta de un dispositivo de entrada (un teclado) por el cual el usuario introduce su identificación y su clave de acceso para que el sistema lo valide y desactive el seguro de la puerta.



Figura 40. Teclado sistema SCAAR

## CIRCUITO INTERNO

El circuito que permite la conexión de cada uno de los componentes que conforman la interfaz con el usuario se muestra en la figura 41, el cual está conectado a la computadora local a través de un cable DB9 (puerto serial). Por medio de este se establece la interfaz con el PC local.

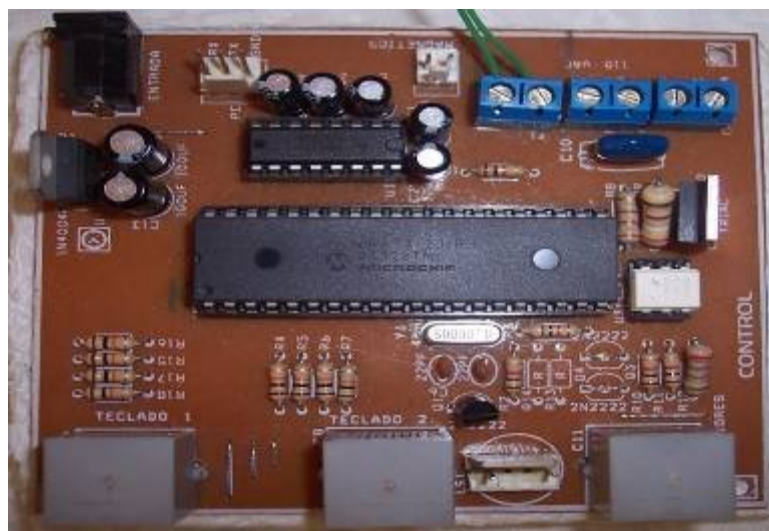


Figura 41. Circuito Interno SCAAR

## **5.3 PRESENTACIÓN DEL SOFTWARE**

### **Módulo De Administración**

El entorno de trabajo para la programación de la aplicación de administración se desarrolló en Java, y consta de 2 programas: el software encargado de gestionar la base de datos, y el software que permite interactuar con el dispositivo hardware (lenguaje de programación ASSEMBLER o lenguaje máquina) y así obtener información en tiempo real sobre el área restringida.

### **Módulo De Comunicación**

El módulo de interfase serial de comunicaciones asincrónico es utilizado para transmitir señales a través del cable RS-232, entre la computadora local y el dispositivo hardware.

Por medio de una rutina especial son inicializados los registros utilizados en la comunicación tanto en la recepción como en la transmisión de datos. Dichos registros son: estado de la recepción, estado de la transmisión, generador de tasa de velocidad en la comunicación.

Esta tasa de transmisión debe estar sincronizada con la velocidad de recepción del puerto serial.

### **Software Del Dispositivo Hardware**

El programa quemado en la EPROM del microcontrolador es en sí un proceso cíclico de interrupciones. El sistema se encuentra en modo SLEEP, método que permite el ahorro considerable de energía (sólo se requiere de una interrupción para salir de éste estado).

El diseño del algoritmo general de funcionamiento del programa consta de un programa principal, el módulo de interrupción y varias subrutinas. Las cuales se describen a continuación:

### **Programa principal**

El programa principal es una rutina que inicializa los puertos, registros y contadores. El programa maneja básicamente una interrupción: por el sensor magnético (si se abre la puerta sin autorización).

### **Interrupción por sensor magnético**

El módulo de interrupciones por sensor magnético, ocurre cuando se ha detectado un cambio en el valor lógico del puerto donde se encuentra conectado dicho sensor (pin 33 del microcontrolador). El evento se genera cuando se abre la puerta sin previa autorización.

Una vez el ciclo del programa entra en la interrupción por sensor magnético, se preparan los códigos de envío que identifican el tipo de alarma y se invoca la subrutina Interrupción.

### **Evento Comunicación serial**

El evento por comunicación serial sucede cuando la computadora local envía algún tipo de orden al dispositivo hardware o viceversa, el flujo del programa entra en las subrutinas enviar y escuchar, y pasa a ejecutar la orden recibida de la computadora local.

## Subrutina Teclado

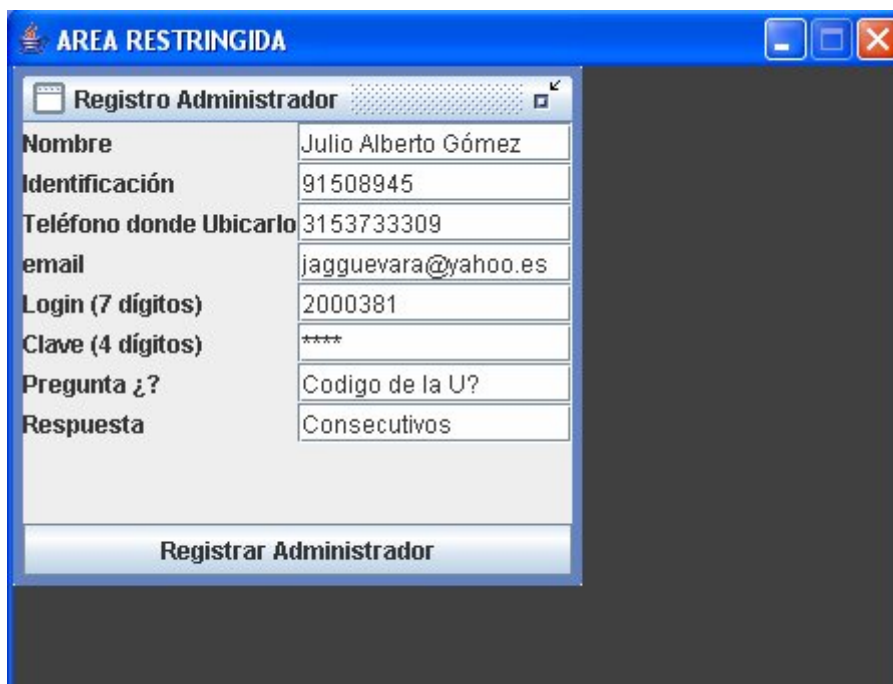
El evento sucede cuando se ha presionado alguna tecla de uno de los teclados que hacen parte del dispositivo hardware, en este momento el flujo del programa entra en la subrutina SIPRESIONO.

## 5.4 ESPECIFICACIÓN DE LA INTERFASE

A continuación, se define el aspecto externo de la aplicación y la manera como el administrador va a interactuar con este.

### SISTEMA DE CONTROL DE ACCESO A ÁREAS RESTRINGIDAS

#### INICIO DEL SISTEMA POR PRIMERA VEZ:



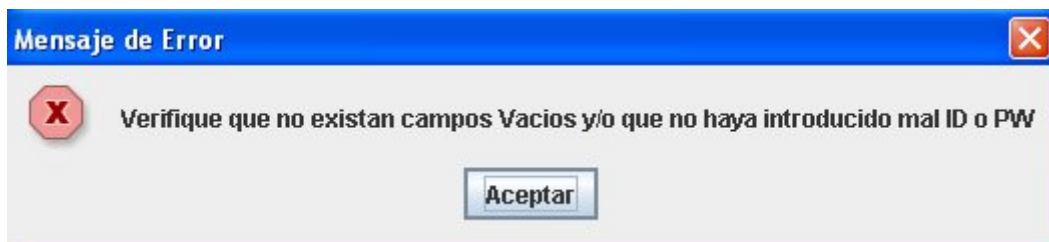
The image shows a screenshot of a software interface titled "AREA RESTRINGIDA". Inside this window, there is a sub-window titled "Registro Administrador". This sub-window contains a form with the following fields and values:

Nombre	Julio Alberto Gómez
Identificación	91508945
Teléfono donde Ubicarlo	3153733309
email	jagguvara@yahoo.es
Login (7 dígitos)	2000381
Clave (4 dígitos)	****
Pregunta ¿?	Codigo de la U?
Respuesta	Consecutivos

At the bottom of the "Registro Administrador" sub-window, there is a button labeled "Registrar Administrador".

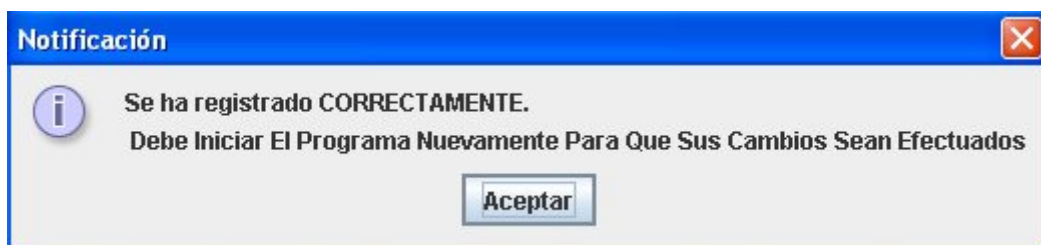
Si es la primera vez que se inicia el sistema de control de acceso se despliega el módulo de registro del administrador. Éste debe registrarse indicando su nombre, su identificación (cédula por ejemplo), un teléfono donde se le pueda ubicar en caso de emergencia, email, el login y clave con los que ingresará al área, una pregunta y una respuesta que le faciliten recordar la clave introducida.

Si el administrador no llena todos los campos necesarios para su registro y/o introduce un Login y/o Clave erróneos le aparece el mensaje



Un Login (ID) y una Clave (PW) se consideran correctos cuando están compuestos por solo siete y cuatro caracteres numéricos respectivamente.

Una vez registrado correctamente el administrador se despliega el mensaje



Nota: El administrador debe crear un grupo de Auxiliar, Director, para asignarle a estos los auxiliares y directores que manejará el sistema respectivamente.

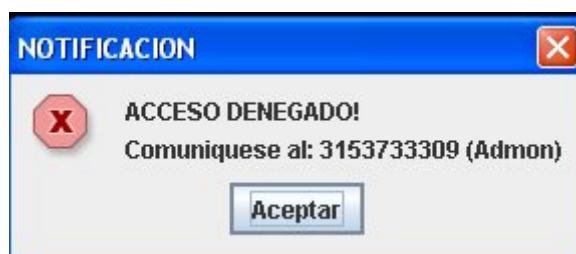
#### INICIO DEL SISTEMA:



El Administrador debe ingresar su clave de acceso al sistema. Si éste no la recuerda debe seleccionar la opción de RECORDAR en cuyo caso se le desplegará la información que el administrador registró para recordar su clave de acceso en caso de olvido:



Si la persona que intenta ingresar no introduce la clave correcta se le despliega el mensaje



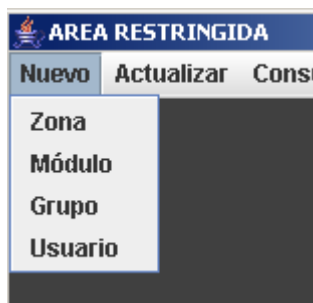
En donde se le informa cómo localizar al administrador para que de inicio al sistema.

UNA VEZ INICIADO EL SISTEMA el administrador cuenta con el menú de:



Los cuales se describirán a continuación.

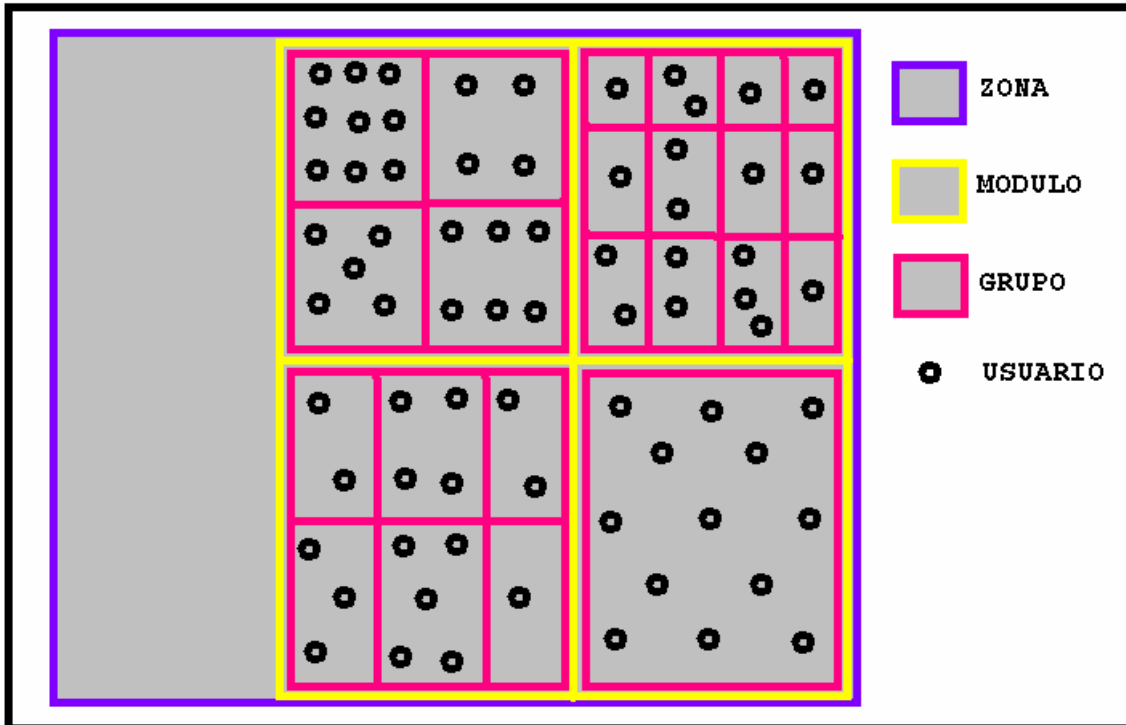
## MENÚ NUEVO



El administrador cuenta con la opción de crear de forma jerárquica zonas que están compuestas por módulos que a su vez están compuestos por grupos en los cuales se encuentran usuarios. Esta jerarquía se debe respetar entendiendo la lógica en la que se desarrolló el sistema.

Esto se puede apreciar en la siguiente gráfica:

### ESQUEMA DE DISTRIBUCIÓN



La jerarquía sugiere que un módulo, grupo y/o usuario pertenece a una y solo una zona, módulo y/o grupo respectivamente.

### MENÚ NUEVA ZONA

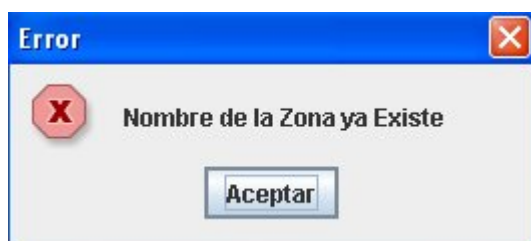
La imagen muestra una interfaz de usuario con un menú desplegable y una ventana de diálogo. El menú desplegable, titulado 'AREA REST...', muestra opciones: 'Nuevo', 'Actualizar', 'Zona', 'Módulo', 'Grupo' y 'Usuario'. La ventana de diálogo, titulada 'AREA RESTRINGIDA', tiene pestañas 'Nuevo', 'Actualizar' y 'Consultar'. La pestaña 'Nuevo' está activa y muestra un formulario para 'Nueva Zona'. El formulario contiene los siguientes campos: 'Nombre' con el valor 'Jose Villabona', 'Estado' con 'Activa' seleccionada (radio), 'Inactiva' no seleccionada (radio), y 'Descripcion' con el valor 'Grupos de Investigación'. En la parte inferior de la ventana hay dos botones: 'Aceptar' y 'Cerrar'.

El administrador cuenta con la opción de crear una zona indicando su nombre, el estado de dicha zona (activa / inactiva) y una breve descripción de la misma.

Si una zona es creada en estado inactivo ésta se crea virtualmente (en la Base de Datos) pero no está actualmente en funcionamiento. Si se desea cambiar su estado (a estado activo) se debe actualizar su información en el menú ACTUALIZAR descrito más adelante (Ver MENÚ ACTUALIZAR ZONA).

Una breve descripción ayuda al administrador a relacionar el porqué de la creación de la zona y el objetivo de la misma.

Si se introduce un nombre de zona ya registrado en la Base de Datos aparece el mensaje



Si el administrador no llena los campos necesarios para crear la zona aparece el mensaje



Una vez creada la zona se despliega el mensaje

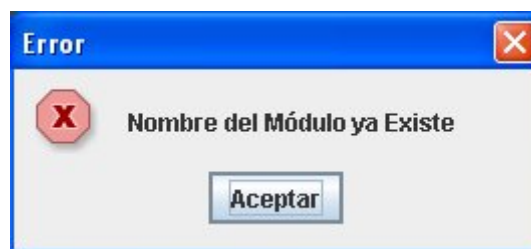


## MENÚ NUEVO MÓDULO

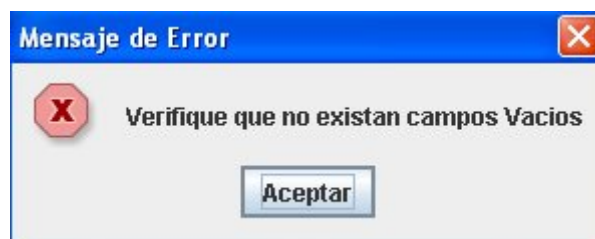




El administrador cuenta con la opción de crear un nuevo módulo indicando su nombre y la zona a la que pertenece. Éste puede crear cuántos módulos requiera dependiendo del ESQUEMA DE DISTRIBUCIÓN a implementar. Si el nombre del módulo a registrar ya se encuentra registrado en la Base de Datos se despliega el mensaje



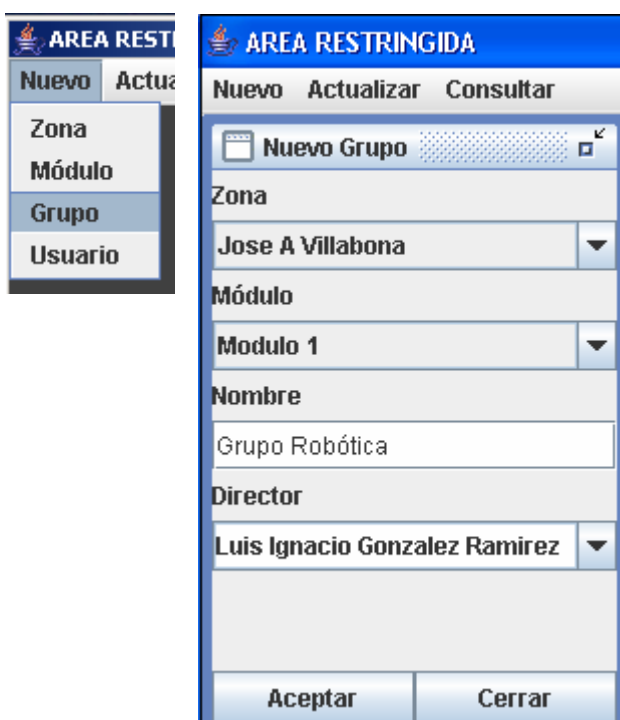
Si el administrador no llena los campos necesarios para la creación del nuevo módulo se despliega el mensaje



Una vez creado el módulo se despliega el mensaje



## MENÚ NUEVO GRUPO



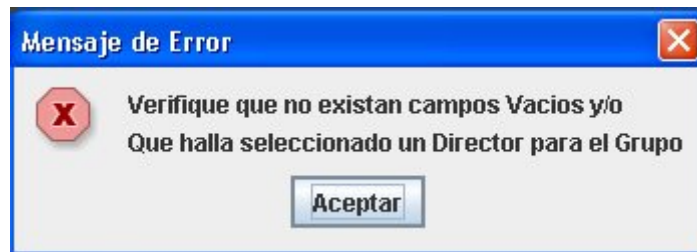
El administrador cuenta con la opción de crear un nuevo grupo indicando su nombre, la zona y módulo al que pertenece y el director al que pertenece dicho grupo. Éste puede crear cuántos grupos requiera dependiendo del ESQUEMA DE DISTRIBUCIÓN a implementar.

Nota: Si aún no se ha registrado ningún Director en la Base de Datos desde el menú NUEVO USUARIO el sistema no exige que lo indique del grupo a crear. El administrador puede crear el grupo y luego desde el menú ACTUALIZAR GRUPO indica, cuando exista, el director.

Si el nombre del grupo que se intenta crear ya se encuentra registrado en la Base de Datos se despliega el mensaje



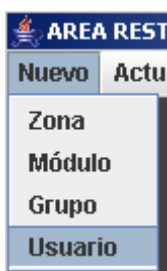
Si el administrador no introdujo los datos necesarios para la creación del nuevo grupo y/o no seleccionó un director para el grupo (existiendo disponibles) se despliega el mensaje



Una vez creado el grupo se despliega el mensaje



## MENÚ NUEVO USUARIO



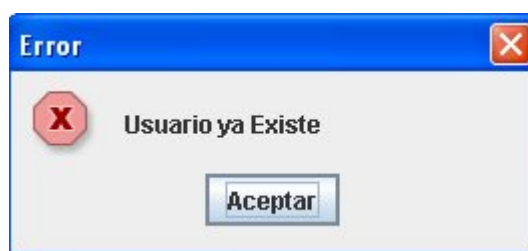
The screenshot shows a software window titled "AREA RESTRINGIDA" with a menu bar containing "Nuevo", "Actualizar", and "Consultar". Below the menu is a tabbed interface with the "Nuevo Usuario" tab selected. The form contains the following fields:

Zona	Jose Villabona
Módulo	Modulo 1
Grupo	Grupo Robótica
Tipo Usuario	Estudiante
Nombre	Silvia Cristina Carreño
Identificación (7 dígitos)	2000380

At the bottom of the form are two buttons: "Aceptar" and "Cerrar".

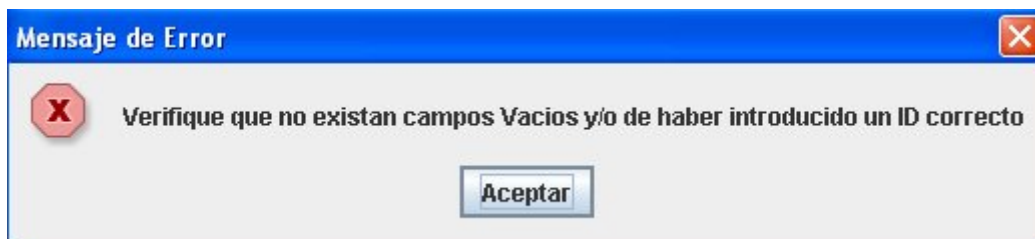
El administrador cuenta con la opción de crear un nuevo usuario indicando la zona, el módulo y el grupo al que pertenece, el tipo de usuario (Auxiliar, Director, Estudiante, Visitante), el nombre e identificación.

El número de identificación del usuario debe ser único e irrepetible y con esto se asegura que un usuario no pertenezca a más de un grupo de investigación. Si se introduce un número de identificación que ya se encuentra registrado en la Base de Datos aparece el mensaje:



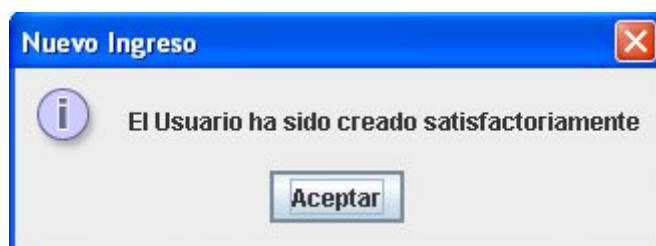
Nota: de ser necesario que un usuario pertenezca a más de un grupo de investigación se pueden crear políticas de manejo del sistema creando, por ejemplo, una codificación interna del número de identificación, en donde, a modo de ejemplo, los tres primeros dígitos sea el número de grupo al que pertenece y los cuatro números restantes sea los cuatro últimos dígitos del carné de la UIS. Actualmente el número de identificación es el código del carné de la Universidad Industrial de Santander.

Existen dos tipos más de validación en este módulo Nuevo Usuario los cuales son: Si el administrador no llena todos los campos necesarios para crear al usuario y/o si el usuario introduce un mal número de identificación aparece el mensaje

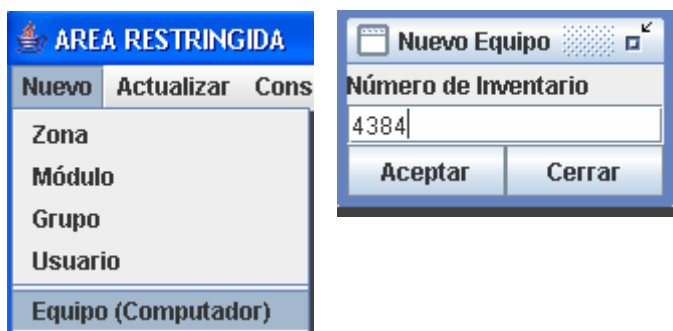


El número de identificación es correcto si éste es formado por solo siete caracteres numéricos (ningún alfa numérico).

Una vez creado el usuario se despliega el mensaje



### MENÚ NUEVO EQUIPO (Computador)



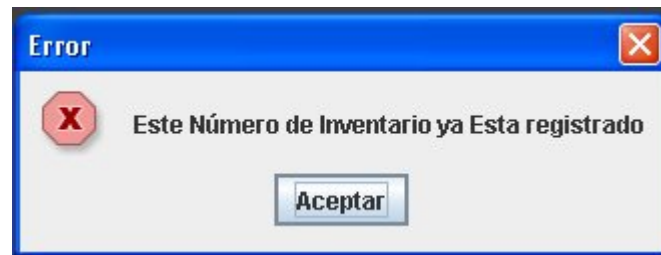
El administrador cuenta con la opción de crear un nuevo equipo, recurso, indicando el número de inventario que éste tiene. El recurso que se esta manejando actualmente en el laboratorio José Villabona es el computador el cuál se le asigna a cada usuario en un horario determinado. (Ver MENU ACTUALIZAR – Horario por Estudiante ).

Si el administrador no introduce algún valor de número de inventario se despliega el mensaje

Nota: Actualmente se considera un valor correcto de número de inventario una cadena formada por caracteres numéricos solamente, es decir, un valor numérico que describa a cada equipo (recurso).



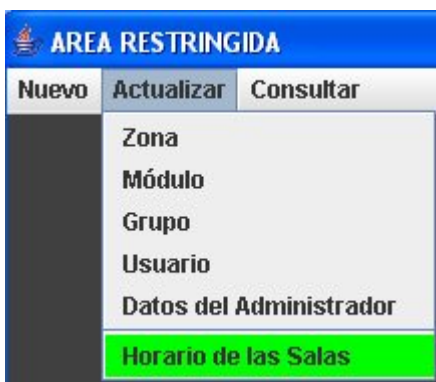
Si el administrador introduce un número de inventario que ya se encuentra registrado en la Base de Datos se despliega el mensaje



Cuando el administrador ha introducido un valor correcto de número de inventario se despliega el mensaje



## MENÚ ACTUALIZAR



El administrador cuenta con la opción de actualizar la información de zonas, módulos, grupos, usuarios, recursos, de sus datos como administrador y de modificar o consultar el Horario establecido para el área y para cada usuario.

## MENÚ ACTUALIZAR ZONA

The image shows two parts of a software interface. On the left is a menu with options: 'Actualizar', 'Consultar', 'Zona', 'Módulo', 'Grupo', 'Usuario', 'Datos del Administrador', and 'Horario de las Salas' (highlighted in green). On the right is a window titled 'AREA RESTRINGIDA' with a sub-window 'Actualizar Zona'. The form contains the following fields: 'Zona' (dropdown menu with 'Jose A Villabona'), 'Nombre' (text box with 'Jose A Villabona'), 'Estado' (radio buttons for 'Activa' and 'Inactiva', with 'Inactiva' selected), 'Descripcion' (text box with 'Salas de Grupos de Inv'), and '¿Razón de Cambio?' (text box with 'Mantenimiento'). At the bottom are 'Aceptar' and 'Cerrar' buttons.

El administrador cuenta con la opción de modificar el nombre dado a la zona, su descripción y su estado. Éste debe dar una pequeña descripción del por qué se hizo el cambio el cual le sirve como soporte a quien desee conocer por qué se activa o desactiva el área en un momento determinado, por ejemplo.

Si la zona se encuentra en estado Inactivo, nadie de los que pertenecen en la escala jerárquica hacia abajo pueden entrar al área. Solo el sistema permite que salgan del área los que se encuentren aún adentro.

Si el administrador no introduce los datos necesarios para la actualización se despliega el mensaje



Esto debido a que no se puede actualizar la información de una zona por valores nulos.

## MENÚ ACTUALIZAR MÓDULO

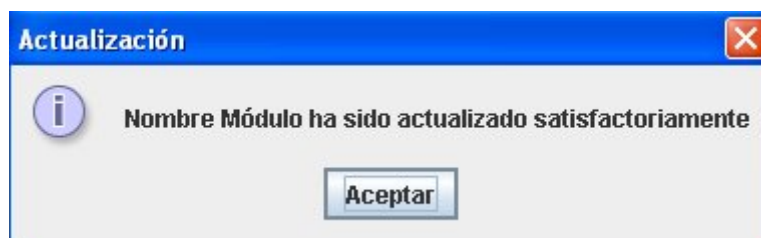
The image shows a menu with two tabs: 'Actualizar' and 'Consultar'. The 'Actualizar' tab is active, showing a list of options: 'Zona', 'Módulo', 'Grupo', 'Usuario', 'Datos del Administrador', and 'Horario de las Salas'. The 'Horario de las Salas' option is highlighted in green. To the right is a dialog box titled 'Actualizar Módulo'. It contains a 'Zona' dropdown menu with 'Jose A Villabona' selected, a 'Módulo' dropdown menu with 'Modulo 1' selected, and a text input field for 'Nombre' containing 'Modulo Investigacion'. At the bottom of the dialog are 'Aceptar' and 'Cerrar' buttons.

El administrador cuenta con la opción de modificar el nombre del módulo por cualquier otro. Éste debe indicar la zona a la que pertenece dicho módulo y seleccionarlo de la lista.

Si el administrador no introduce los datos necesarios para actualizar la información del módulo se despliega el siguiente mensaje



Una vez actualizado el módulo se despliega el siguiente mensaje

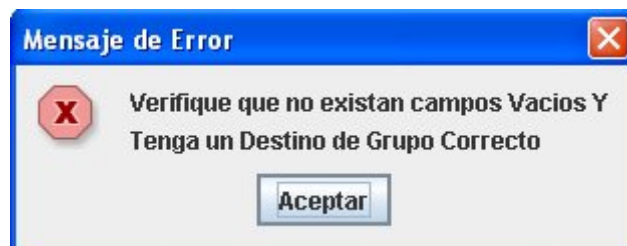


## MENÚ ACTUALIZAR GRUPO

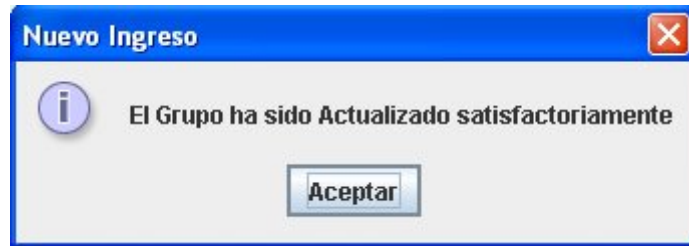
The screenshot shows a web application interface. On the left, there is a vertical menu with two tabs: 'Actualizar' (selected) and 'Consultar'. Under the 'Actualizar' tab, there are several menu items: 'Zona', 'Módulo', 'Grupo', 'Usuario', 'Datos del Administrador', and 'Horario de las Salas' (highlighted in green). The main content area is titled 'AREA RESTRINGIDA' and contains a sub-menu with 'Nuevo', 'Actualizar', and 'Consultar'. The 'Actualizar Grupo' option is selected, opening a form. The form has the following fields: 'Zona' (Jose A Villabona), 'Módulo' (Modulo 1), 'Grupo' (Grupo Robotica), 'Nombre' (Grupo Robotica V2, highlighted in red), 'Director' (Luis Ignacio Gonzalez Ramirez), and 'Módulo al Que desea moverlo' (Modulo 1). At the bottom of the form are two buttons: 'Aceptar' and 'Cerrar'.

El administrador cuenta con la opción de cambiar el nombre, el director asignado y el módulo al que pertenece un grupo. Para ello, debe indicar la zona y el módulo al que pertenece y de la lista seleccionar el grupo que desea modificar.

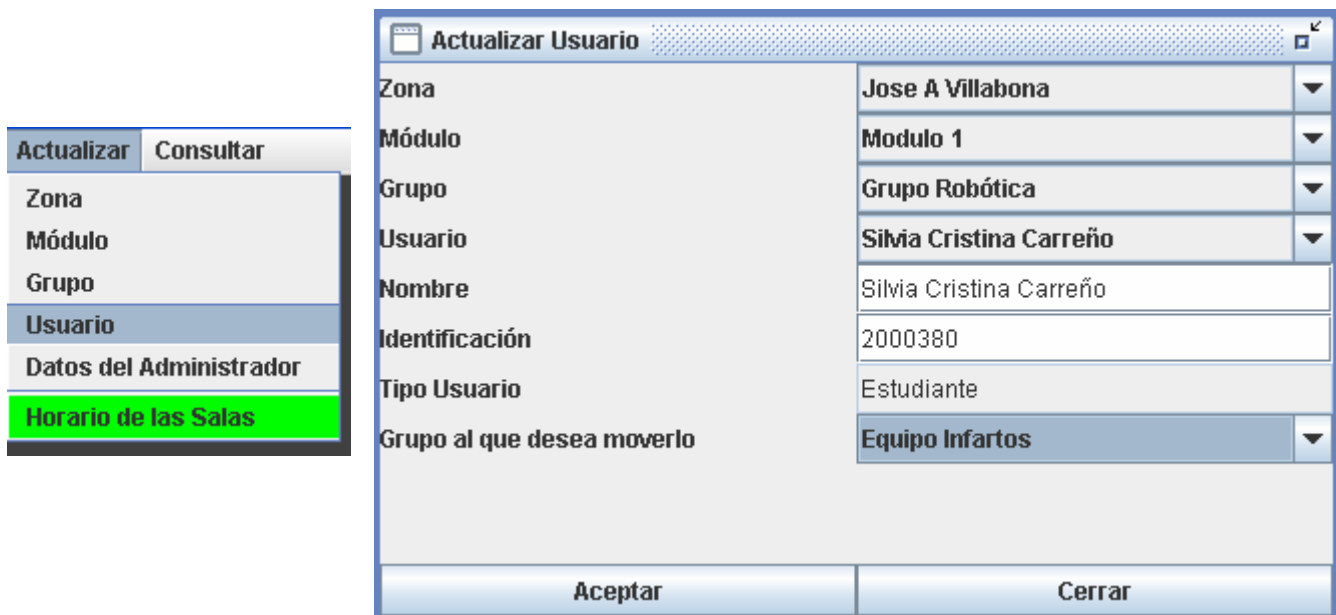
Si el administrador no introduce la información necesaria para actualizar el grupo y/o el valor del módulo al cual se moverá el grupo no es correcto se despliega el mensaje



Una vez actualizado el grupo correctamente se despliega el siguiente mensaje

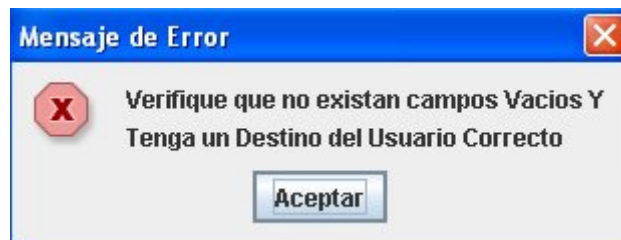


## MENÚ ACTUALIZAR USUARIO



El administrador cuenta con la opción de modificar el nombre, la identificación y el grupo al que pertenece el usuario. Para ello, el administrador debe indicar la zona, el módulo y el grupo y seleccionar de la lista el usuario que desea modificar.

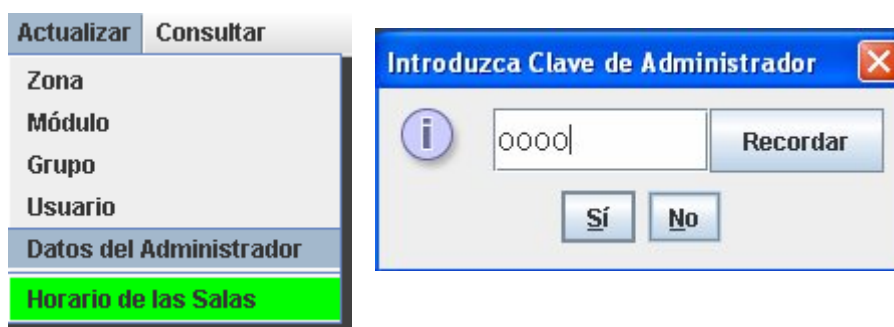
Si el administrador no introduce la información necesaria para realizar la actualización se le despliega el siguiente mensaje



Una vez actualizada la información de usuario se muestra el mensaje



## MENÚ ACTUALIZAR DATOS DEL ADMINISTRADOR



Cuando el administrador desea actualizar su información se le pide antes de entrar al módulo de actualización su clave.

Actualizar Datos del Administrador	
Nombre	Julio Alberto Gomez Guevara
Identificación	91508945
Teléfono donde Ubicarlo	3153733309
email	jagguevara@yahoo.es
Login (7 dígitos)	2000381
Clave (4 dígitos)	****
Pregunta ¿?	Codigo de la U?
Respuesta	Consecutivos
Actualizar Info Administrador	
Cerrar	

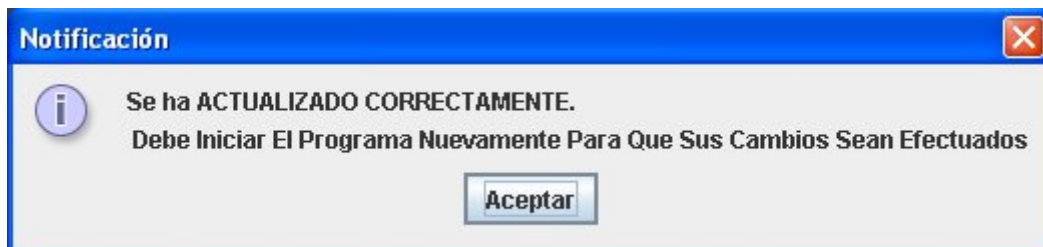
Una vez introducida su clave ingresa al módulo de actualización en donde puede cambiar cualquier información antes registrada.

Si el administrador no suministra la información correcta para que se puedan actualizar sus datos se despliega el siguiente mensaje

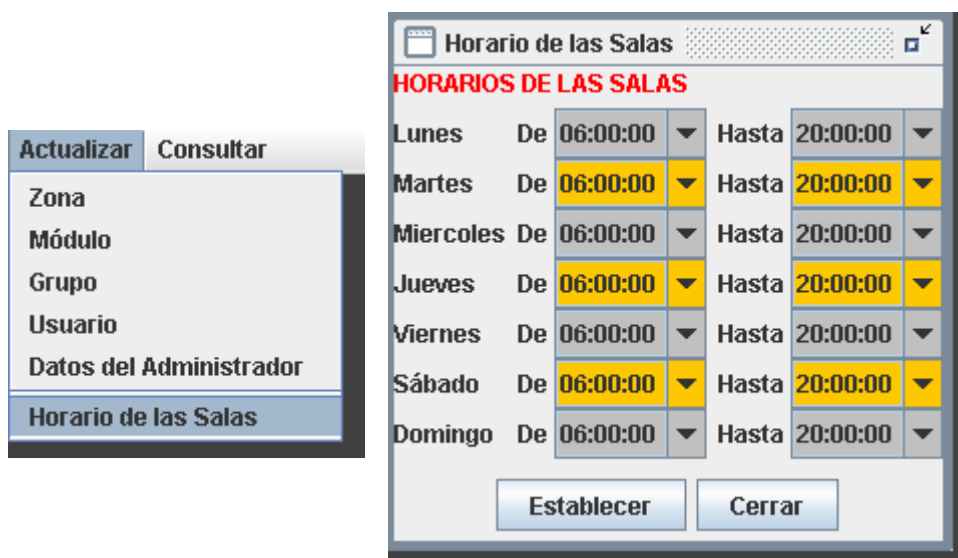


El ID y el PW como se comentó anteriormente deben tener respectivamente siete y cuatro caracteres numéricos. De no ser así, se muestra el mensaje anterior.

Al actualizar la información el administrador debe ingresar nuevamente al sistema para que los cambios sean efectuados. Se despliega el mensaje

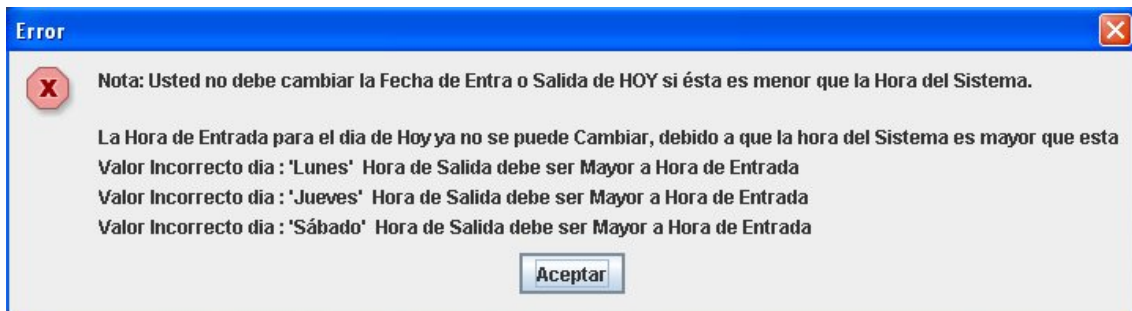


## MENÚ ACTUALIZAR HORARIO DE LAS SALAS

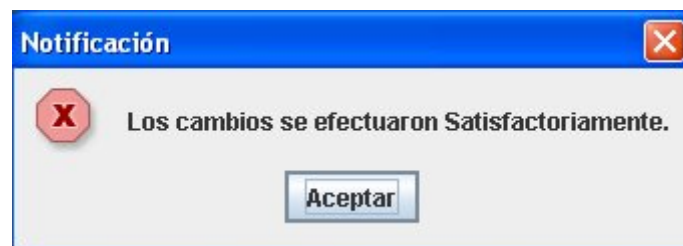


El administrador puede entrar a este módulo como simple consulta de los horarios que ha establecido en la sala o para modificarlos.

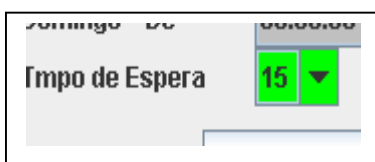
La gráfica anterior muestra los horarios establecidos para cada día. Si el administrador intenta modificarlos y comete algún error, el sistema le indicará la razón del error y lo subrayará en la ventana del módulo. Por ejemplo:



Si el administrador establece los valores correctos para la hora de entrada y salida de cada día se despliega el siguiente mensaje



El administrador cuenta con la opción de asignar el tiempo máximo de espera para cada usuario en una hora determinada de su horario para ingreso a la sala. Si el usuario llega después de este tiempo en su hora de horario el sistema le negará el acceso.



Nota: Un usuario solo debe activar la hora de su horario para poder entrar y salir libremente durante

esa hora. La forma de activarla es entrando antes del tiempo máximo de espera permitido, o saliendo del área cuando la hora ya esta activa.

**Nota:** Al actualizar la información el administrador debe ingresar nuevamente al sistema para que los cambios sean efectuados.

## MENÚ ACTUALIZAR HORARIO POR ESTUDIANTE

Actualizar	Consultar
Zona	
Módulo	
Grupo	
Usuario	
Datos del Administrador	
Horario de las Salas	
Horario Por Estudiante	
Equipo (Computador)	

El administrador cuenta con el opción de establecer para cada usuario el equipo a utilizar y el horario en que el usuario se compromete a usar dicho equipo.

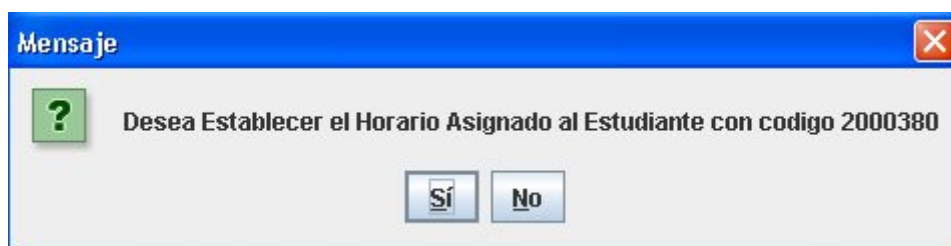
El administrador debe tener en cuenta que cada equipo debe ser usado por un solo usuario a una hora determinada. El sistema previene que el administrador cometa errores al momento de registrar los horarios, pues va reservando el tiempo a cada equipo y solo deja disponible el tiempo aún no asignado.

Horario Por Estudiante											
Nombre Usuario		Julio Alberto Gomez Guevara		Consultar		60128					
LUNES	MARTES	MIERCOLES	JUEVES	VIERNES	SABADO						
<input checked="" type="radio"/> 6	<input checked="" type="radio"/> 6	<input checked="" type="radio"/> 6	<input checked="" type="radio"/> 6	<input checked="" type="radio"/> 6	<input checked="" type="radio"/> 6	1991921	1991921	1991921	1991921	1991921	Dis
<input checked="" type="radio"/> 7	<input checked="" type="radio"/> 7	<input checked="" type="radio"/> 7	<input checked="" type="radio"/> 7	<input checked="" type="radio"/> 7	<input checked="" type="radio"/> 7	1991921	1991921	1991921	1991921	1991921	Dis
<input checked="" type="radio"/> 8	<input checked="" type="radio"/> 8	<input checked="" type="radio"/> 8	<input checked="" type="radio"/> 8	<input checked="" type="radio"/> 8	<input checked="" type="radio"/> 8	1991957	1991957	1991957	1991957	1991957	Dis
<input checked="" type="radio"/> 9	<input checked="" type="radio"/> 9	<input checked="" type="radio"/> 9	<input checked="" type="radio"/> 9	<input checked="" type="radio"/> 9	<input checked="" type="radio"/> 9	1991957	1991957	1991957	1991957	1991957	Dis
<input checked="" type="radio"/> 10	<input checked="" type="radio"/> 10	<input checked="" type="radio"/> 10	<input checked="" type="radio"/> 10	<input checked="" type="radio"/> 10	<input checked="" type="radio"/> 10	1991957	1991957	1991957	1991957	1991957	Dis
<input checked="" type="radio"/> 11	<input checked="" type="radio"/> 11	<input checked="" type="radio"/> 11	<input checked="" type="radio"/> 11	<input checked="" type="radio"/> 11	<input checked="" type="radio"/> 11	1991957	1991957	1991957	1991957	1991957	Dis
<input type="radio"/> 12	<input type="radio"/> 12	<input type="radio"/> 12	<input type="radio"/> 12	<input type="radio"/> 12	<input type="radio"/> 12	Disponibile	Disponibile	Disponibile	Disponibile	Disponibile	Dis
<input type="radio"/> 13	<input type="radio"/> 13	<input type="radio"/> 13	<input type="radio"/> 13	<input type="radio"/> 13	<input type="radio"/> 13	Disponibile	Disponibile	Disponibile	Disponibile	Disponibile	Dis
<input checked="" type="radio"/> 14	<input checked="" type="radio"/> 14	<input checked="" type="radio"/> 14	<input checked="" type="radio"/> 14	<input checked="" type="radio"/> 14	<input checked="" type="radio"/> 14	1991957	1991957	1991957	1991957	1991957	Dis
<input checked="" type="radio"/> 15	<input checked="" type="radio"/> 15	<input checked="" type="radio"/> 15	<input checked="" type="radio"/> 15	<input checked="" type="radio"/> 15	<input checked="" type="radio"/> 15	1991957	1991957	1991957	1991957	1991957	Dis
<input checked="" type="radio"/> 16	<input checked="" type="radio"/> 16	<input checked="" type="radio"/> 16	<input checked="" type="radio"/> 16	<input checked="" type="radio"/> 16	<input checked="" type="radio"/> 16	1991957	1991957	1991957	1991957	1991957	Dis
<input checked="" type="radio"/> 17	<input checked="" type="radio"/> 17	<input checked="" type="radio"/> 17	<input checked="" type="radio"/> 17	<input checked="" type="radio"/> 17	<input checked="" type="radio"/> 17	1991957	1991957	1991957	1991957	1991957	Dis
<input type="radio"/> 18	<input type="radio"/> 18	<input type="radio"/> 18	<input type="radio"/> 18	<input type="radio"/> 18	<input type="radio"/> 18	Disponibile	Disponibile	Disponibile	Disponibile	Disponibile	Dis
<input type="radio"/> 19	<input type="radio"/> 19	<input type="radio"/> 19	<input type="radio"/> 19	<input type="radio"/> 19	<input type="radio"/> 19	Disponibile	Disponibile	Disponibile	Disponibile	Disponibile	Dis
<input type="radio"/> 20	<input type="radio"/> 20	<input type="radio"/> 20	<input type="radio"/> 20	<input type="radio"/> 20	<input type="radio"/> 20	Disponibile	Disponibile	Disponibile	Disponibile	Disponibile	Dis
<input type="radio"/> 21	<input type="radio"/> 21	<input type="radio"/> 21	<input type="radio"/> 21	<input type="radio"/> 21	<input type="radio"/> 21	Disponibile	Disponibile	Disponibile	Disponibile	Disponibile	Dis

El administrador debe consultar si el usuario tiene ya tiempo registrado para luego establecerle más horas o para quitarle

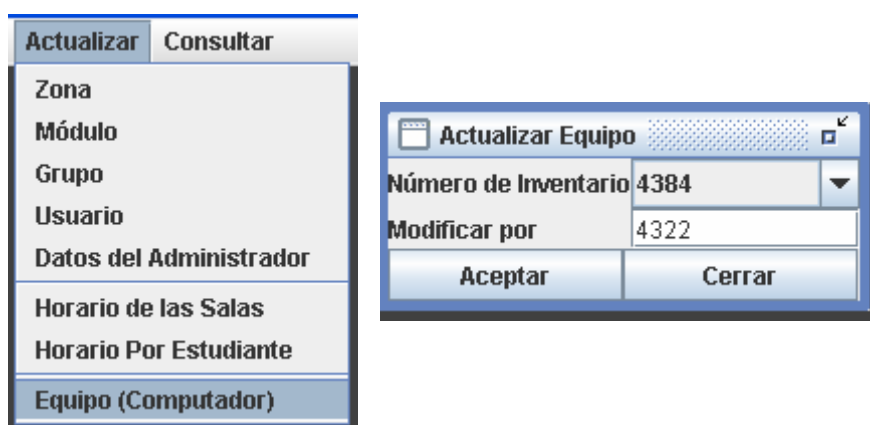
Cuando el administrador indica el horario para un usuario que se encuentra registrado se le despliega el mensaje de confirmación. Este mensaje es necesario para advertir al administrador de los cambios que esta realizando y así prevenir que modifique el horario de algún usuario por accidente.

Nota: cada cambio de horario queda registrado en la base de datos llevando historia de los horarios que se le han establecido al usuario desde el momento de su registro en el sistema. Esto permite que existan soportes estadísticos confiables de cada usuario evaluando el uso que le da al área comparándolo contra con su compromiso, es decir, contra su horario registrado en el tiempo de evaluación (Si hay más de un horario se evalúan períodos de tiempo separados. Esto lo hace el sistema sin que el administrador tenga que preocuparse por ello).



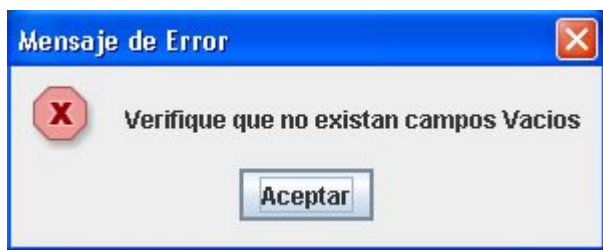
Nota: Si un usuario intenta entrar al área y no se le ha registrado aún un horario de uso, el sistema NO lo deja entrar.

### **MENÚ ACTUALIZAR EQUIPO (Computador)**

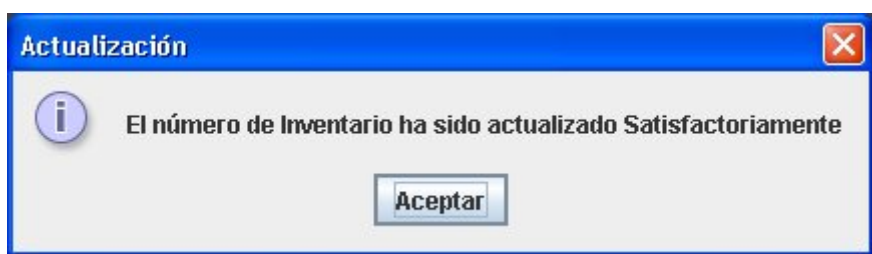


El administrador cuenta con la opción de modificar el número de inventario registrado por equipo (recurso).

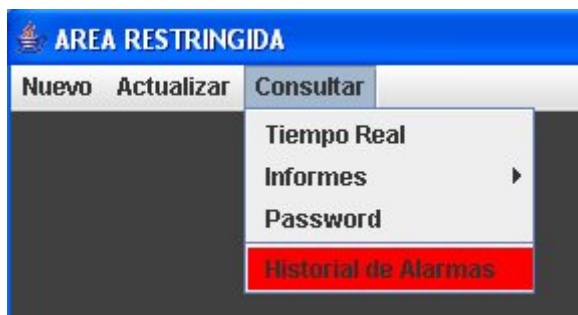
Si el administrador no suministra la información necesaria para la actualización se le despliega el mensaje



Una vez suministrada la información necesaria para la actualización se le despliega el mensaje al administrador indicándole que el registro ha sido actualizado correctamente



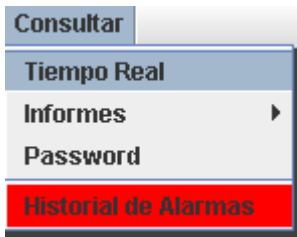
## MENÚ CONSULTAR



Por medio de este módulo el administrador puede:

- Conocer en tiempo real los usuarios que se han registrado en el sistema y se encuentran actualmente en el área.
- Conocer las zonas, módulos, grupos, usuarios, equipos creados y registrados en el sistema.
- Conocer informes estadísticos sobre el uso del área.
- Consultar el password de un usuario a petición para que éste pueda ingresar al área haciendo uso del mismo y de su número de identificación.
- Conocer los mensajes de alarma que se han presentado categorizados por fecha de ocurrencia.

## MENÚ CONSULTAR TIEMPO REAL



ÁREA RESTRINGIDA

Nuevo Actualizar Consultar

Consultar Tiempo Real

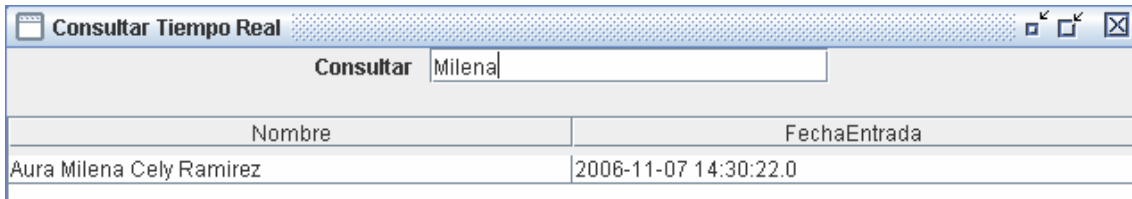
Consultar

Nombre	FechaEntrada
Oscar Villamizar Blanco	2006-11-07 14:19:47.0
Aura Milena Cely Ramirez	2006-11-07 14:30:22.0
Nidya Maritza Alarcon Tarazona	2006-11-07 14:33:16.0
Monica Janeth Blanco Diaz	2006-11-07 14:36:40.0
Hina Luz Garavito Robles	2006-11-07 14:47:15.0
Jhon Jairo Villamizar Socha	2006-11-07 15:01:29.0
Nicolas Estrada Cruz	2006-11-07 15:01:52.0
Julio Pastor Posada Orozco	2006-11-07 15:07:47.0
Clayderman Josue Rojas Jimenez	2006-11-07 15:25:26.0
William Leonardo Garcia Rueda	2006-11-07 15:25:41.0

Número de Usuarios en la Sala : 10

El administrado cuenta con la opción de conocer los usuarios que se han registrado y se encuentran actualmente en la sala. Se ordenan por orden de llegada siendo Oscar Villamizar, según el ejemplo, el primero en llegar y William Leonardo el último.

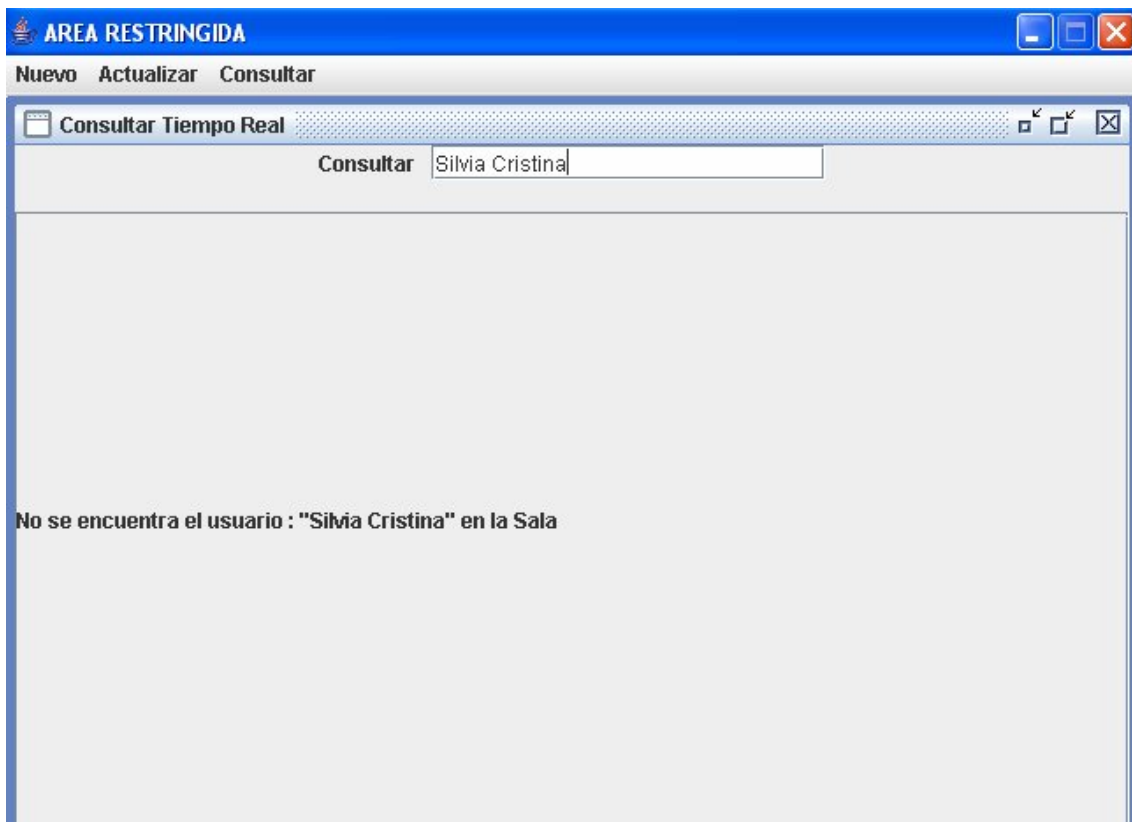
En el campo consultar se puede filtrar la información mostrada por nombre para lo cual el sistema nos muestra si la persona que estamos buscando se encuentra o no dentro de la sala.



Consultar	
Nombre	FechaEntrada
Aura Milena Cely Ramirez	2006-11-07 14:30:22.0

En este caso se filtraron los datos con la palabra “Milena” la cual se encuentra en el registro Aura Milena Cely Ramirez y por lo tanto se muestra en el resultado.

Si por el contrario, consultamos una persona que no se encuentre en la sala el sistema nos mostrará el mensaje de que dicho usuario no esta en la sala.



Nuevo Actualizar Consultar

Consultar Tiempo Real

Consultar Silvia Cristina

No se encuentra el usuario : "Silvia Cristina" en la Sala

## BIBLIOGRAFÍA

- Título:** Base de Datos Relacionales: Un enfoque Práctico de Diseño  
**Autor(es):** José Cárcamo Sepúlveda  
**Editorial:** Ediciones UIS  
**Año de Publicación:** 1994
- Título:** Java 2 Curso de Programación  
**Autor(es):** Francisco Javier Ceballos Sierra  
**Editorial:** Alfa-Omega  
**Año de Publicación:** 2002
- Título:** Curso avanzado de microcontroladores PIC  
**Autor(es):** Edison Duque C.  
**Editorial:** CENIT Compañía Editorial Tecnológica  
**Año de Publicación:** 1998
- Título:** MICROCONTROLADORES PIC  
**Editorial:** McGRAW-HILL/INTERAMERICANA DE ESPAÑA, S.A.U.  
**Año de Publicación:** 1999
- Título:** Análisis y Diseño de Sistemas  
**Autor(es):** Kenneth E. Kendall – Julie E. Kendall  
**Editorial:** Pearson Educación  
**Edición:** Tercera Edición  
**Año de Publicación:** 1997
- Título:** Ingeniería del Software: Un enfoque Práctico  
**Autor(es):** Roger S. Presuman  
**Editorial:** McGraw Hill  
**Edición:** Quinta Edición  
**Año de Publicación:** 2002
- Título:** UML Gota a Gota  
**Autor(es):** Martin Fowler, Kendail Scout  
**Editorial:** Pearson

**Título:** Como Programar en Java  
**Autor(es):** Deitel & Deitel  
**Edición:** Quinta Edición  
**Año de Publicación:** 2003

- ✓ <http://www.mysql-hispano.org/page.php?id=37>
- ✓ [en.wikipedia.org/wiki/Assembler](http://en.wikipedia.org/wiki/Assembler)
- ✓ [www.monografias.com/trabajos/assembler/assembler.shtml](http://www.monografias.com/trabajos/assembler/assembler.shtml)
- ✓ [www.frino.com.ar/assembler.htm](http://www.frino.com.ar/assembler.htm)
- ✓ [personales.ya.com/cepalacios/](http://personales.ya.com/cepalacios/)
- ✓ [perso.wanadoo.es/chyryes/tutoriales/pic0.htm](http://perso.wanadoo.es/chyryes/tutoriales/pic0.htm)
- ✓ [ww1.microchip.com/downloads/en/devicedoc/35007b.pdf](http://ww1.microchip.com/downloads/en/devicedoc/35007b.pdf)
- ✓ [www.pic16f84a.com/](http://www.pic16f84a.com/)
- ✓ [www.todopic.com.ar/pines\\_f84.html](http://www.todopic.com.ar/pines_f84.html)
- ✓ [members.fortunecity.es/electronico/tutoriales/tutopic16f84.html](http://members.fortunecity.es/electronico/tutoriales/tutopic16f84.html)
- ✓ [oros.emagister.com/tema-pic16f84-13582-363046.htm](http://oros.emagister.com/tema-pic16f84-13582-363046.htm)
- ✓ [www.ubasics.com/adam/pic/picprog.html](http://www.ubasics.com/adam/pic/picprog.html)
- ✓ [www.monografias.com/trabajos18/descripcion-pic/descripcion-pic.shtml](http://www.monografias.com/trabajos18/descripcion-pic/descripcion-pic.shtml)
- ✓ [ww1.microchip.com/downloads/en/DeviceDoc/30292c.pdf](http://ww1.microchip.com/downloads/en/DeviceDoc/30292c.pdf)
- ✓ [www2.ate.uniovi.es/fernando/Doc2005/Sed\\_05/Practicas/Descripcion/Practica\\_9/Practica%209.pdf](http://www2.ate.uniovi.es/fernando/Doc2005/Sed_05/Practicas/Descripcion/Practica_9/Practica%209.pdf)
- ✓ [www.solotutoriales.com/directory/details.aspx?id=2688](http://www.solotutoriales.com/directory/details.aspx?id=2688)

## CONCLUSIONES

- La automatización no debe verse solo en películas de ciencia ficción, sino llevarse a la realidad como muchos países desarrollados lo hacen hoy en día.
- Proyectos enfocados en la robótica son de interés nacional y dan una buena imagen a la universidad que propone la investigación en dichos campos.
- Las sugerencias provenientes de personas con experiencia en el campo en que estamos realizando el proyecto no se deben menospreciar, pues una vez llevadas a cabo se entiende la razón de las mismas.
- Los diagramas UML facilitan la comunicación con el usuario y permiten realizar cambios de ser necesarios.
- La recopilación de requerimientos es una de las facetas más importantes de un proyecto de prototipado evolutivo, pues enfocan el desarrollo inicial o primera fase y conllevan a nuevos requerimientos basados en lo ya hecho hasta el momento.
- Se debe promover a los estudiantes la realización de sus proyectos de grado en campos enfocados en la automatización ya que a nivel nacional e internacional esta es la tendencia para soluciones futuras.
- El uso de sistemas para el control del manejo de recursos limitados permiten su optimización y la disminución de costos de operación y mantenimiento.
- Una buena comunicación entre el director de proyecto de grado y los estudiantes comprometidos con la elaboración del mismo es importante pues permite la evolución del mismo y el alcance efectivo de los objetivos.
- El diseño efectivo de la Base de Datos permite el desarrollo adecuado del sistema de información y tener informes confiables para la toma de decisiones.
- El manejo de historia en la Base de Datos permite conocer el estado de una sistema de información en un tiempo determinado. Esto, en sistemas de información que manejan datos estadísticos es de vital importancia para tener resultados confiables a la hora de tomar decisiones.
- Ver la necesidad de las personas y plantear soluciones informáticas para las mismas es la labor de un ingeniero de sistemas.
- El uso de comentarios en los códigos fuente de los programas desarrollados permite la continuidad en las versiones sucesoras de los proyectos realizados.
- El periodo de implementación de un proyecto de grado debe ser igual o superior a la mitad del tiempo del total de desarrollo del mismo. Esto permite que el sistema sobrepase la resistencia al cambio y su estabilización en el uso por parte de los usuarios.

## RECOMENDACIONES

- El sistema esta implementado en el área Jose Alberto Villabona. Se recomienda crear políticas de manejo del sistema para prevenir errores y violaciones al mismo por parte de los usuarios.
- Se debe evaluar el sistema periódicamente enfocándose en los inconvenientes que presentan los usuarios para atenderlos en versiones posteriores.
- Para la instalación del sistema se debe leer el manual de Instrucciones para la Instalación del sistema SCAAR.
- El administrador debe periódicamente, desde el módulo Actualizar Datos del Administrador, depurar la base de datos para prevenir errores en la misma y consigo resultados no confiables en los reportes estadísticos.
- Contar con personas dispuestas y comprometidas con el sistema para mejorarlo y robustecerlo en las versiones sucesoras.
- Promover e incentivar el manejo adecuado del sistema, haciéndolo más industrial, poniendo los teclados en lugares cómodos para el usuario, poniendo la tarjeta del pic a prueba de estática para que no se active la alarma accidentalmente, realizar chequeos periódicos de los teclados para determinar si cada tecla funciona adecuadamente, de lo contrario, cambiar el teclado removiéndolo y poniendo otro. Esto es factible ya que el teclado es de bajo costo y de fácil adquisición en el centro de la ciudad.
- Ir limpiando periódicamente, según las políticas de manejo, la base de datos.
- Hacer chequeos periódicos de la gente que se encuentra actualmente en la sala para advertir a los usuarios que hacen uso indebido de su identificación y password. Esto es necesario hasta que se planteen soluciones mecánicas para prevenir que entren personas no autorizadas al área restringida.
- Consultar periódicamente, según políticas de manejo y administración, las estadísticas de los usuarios para optimizar el uso de los recursos disminuyendo o aumentando el compromiso de uso por parte de los usuarios.
- Tener la idea de optimización de recursos para la toma de decisiones y llevar a cabo la política de Solución de Extremos, expuesta en éste documento.