



**UNIVERSIDAD POLITÉCNICA SALESIANA**

**SEDE GUAYAQUIL**

**FACULTAD DE INGENIERIAS**

**CARRERA:**

**INGENIERIA DE SISTEMAS**

**TESIS PREVIA A LA OBTENCION DEL TITULO DE:**

**Ingeniería de Sistemas con mención de Telemática**

**TEMA**

**“Sistema de Control de Acceso y Monitoreo con la Tecnología RFID para el Departamento de Sistemas de la Universidad Politécnica Salesiana Sede Guayaquil”**

**AUTORA:**

**ZYNNIA VERÓNICA VARGAS VERGARA**

**TUTOR:**

**ING. DARIO HUILCAPI SUBIA**

**Guayaquil, 2013**

## DECLARATORIA DE RESPONSABILIDAD

Los conceptos desarrollados, análisis realizados y las conclusiones del presente proyecto, son de exclusiva responsabilidad de mi autoría y el patrimonio intelectual le pertenece a la Universidad Politécnica Salesiana.

Guayaquil, Julio del 2013

f.....

Zynnia Verónica Vargas Vergara

## **DEDICATORIA**

El presente proyecto va dedicado a mis padres por el apoyo incondicional, porque creyeron en mí, porque gracias a ustedes he llegado hasta aquí para poder alcanzar mi meta a mi abuelita, a mis tíos, a mi hermana, a mi novio y a mis amigos gracias por ser parte de mi anhelo de triunfar en la vida.

## **AGRADECIMIENTO**

### **A DIOS**

Por permitirme haber llegado hasta ésta etapa de mi vida con salud.

### **A MI PADRE**

El Dr. Ing. Leonardo Echeverría Fabre que aunque no somos de la misma sangre ha cuidado de mi económicamente y afectivamente siendo un padre ejemplar e incondicional.

### **A MI MAMI**

Zynnia Vergara Torres por estar conmigo en las buenas y en las malas apoyándome en mis triunfos y en mis fracasos aconsejándome cada día, siendo siempre lo más importante y más valioso de mi vida.

### **A MI ABUELITA Y TODA MI FAMILIA**

Por creer en mí, por las desveladas y amanecidas durante toda mi etapa estudiantil y mi carrera universitaria.

### **A MIS PROFESORES**

En especial al Ing. Darío Huilcapi S. que me ha ayudado en mi carrera universitaria impartiendo su conocimiento tanto en clases como para la elaboración de mi tesis.

### **A MIS AMIGOS**

Carlos Rosero, Cristina Dávila, Cristhian Astudillo que fueron parte de mi etapa de estudio.

### **A MI NOVIO**

Darío Bernal por acompañarme a mis clases y ayudarme en uno que otro deber y sobre todo por el apoyo diario para poder cumplir una de mis metas.

**UNIVERSIDAD POLITECNICA SALESIANA**

**FACULTAD DE INGENIERIA EN SISTEMAS**

**TEMA**

Sistema de Control de Acceso y Monitoreo con la Tecnología RFID para el Departamento de Sistemas de la Universidad Politécnica Salesiana Sede Guayaquil.

**AUTORA:** ZYNNIA VERÓNICA VARGAS VERGARA

**TUTOR:** ING. DARÍO HUILCAPI SUBIA

**FECHA:** JULIO DEL 2013

**RESUMEN**

El proyecto nació de la necesidad de brindar seguridad a los equipos del laboratorio de Telemática de la Universidad Politécnica Salesiana. Este diseño consiste en un Sistema de Control de Acceso al Laboratorio, permitiendo un monitoreo constante de los equipos y el acceso controlado del personal autorizado. El sistema SCAL utiliza un módulo de identificación inalámbrica denominado RFID (Radio Frequency Identification) que tiene como fin identificar, gestionar y controlar al personal docente y de mantenimiento autorizado. El módulo de acceso que se realizó es un seguro método destinado a controlar el ingreso y egreso del personal al laboratorio. Fue elaborado en Netbeans mediante una conexión UDP. El software de control de acceso permite configurar el hardware desde la PC y elaborar cuadros estadísticos.

El Monitoreo se efectuó en LabVIEW empleando la tecnología RFID el cual permite identificar una etiqueta electrónica a distancia, que emite periódicamente una señal de radiofrecuencia hacia el módulo lector RFID.

El sistema SCAL y la tecnología RFID, ayudará en la reducción del costo por reposición de los equipos perdidos, tener precisión a la hora de necesitar los laboratorios y evitar la supervisión por parte del personal de mantenimiento.

**Palabras claves:** Acceso autorizado, equipos de laboratorio, tecnología RFID, monitoreo inalámbrico, lector de tarjetas, tag, controladores y microcontroladores.

**UNIVERSIDAD POLITECNICA SALESIANA  
FACULTY OF ENGINEERING SYSTEM**

**TOPIC**

System Monitoring and Access Control with RFID Technology for Systems  
Department Headquarters Salesian Polytechnic University Guayaquil.

**AUTHOR:** VERONICA VARGAS ZYNNIA VERGARA

**TUTOR:** ING. HUILCAPI DARÍO SUBIA

**DATE:** JULY 2013

**ABSTRACT**

The project arose from the need to provide security for laboratory equipment Telematics Salesian Polytechnic University. This design consists of a Control System Laboratory Access, allowing constant monitoring of equipment and controlled access to authorized personnel. SCAL system uses a wireless identification module called RFID (Radio Frequency Identification) that aims to identify, manage and control the maintenance staff and authorized. The access module is held safe method for controlling the entry and exit of the laboratory staff. It was developed in Netbeans using a UDP connection. The access control software to configure the hardware from the PC and elaborate statistical tables. Monitoring was done in LabVIEW using RFID technology which allows identifying a remote electronic tag, which periodically emits a radio frequency signal to the RFID reader module.

**SCAL SYSTEM AND RFID TECHNOLOGY WILL HELP IN REDUCING THE COST OF REPLACING LOST EQUIPMENT, BE ACCURATE AT THE TIME OF NEED LABORATORIES AND AVOID SUPERVISION BY THE MAINTENANCE STAFF.**

**KEYWORDS: UNAUTHORIZED ACCESS, LABORATORY EQUIPMENT, RFID, WIRELESS MONITORING, CARD READER, TAG, CONTROLLERS AND MICROCONTROLLERS.**

## INDICE INICIAL

<b>Tema</b>	<b>Pág.</b>
<b>Resumen.....</b>	<b>V</b>
<b>Índice General.....</b>	<b>VII</b>
<b>Índice de Gráficos.....</b>	<b>IX</b>
<b>Índice de Tablas.....</b>	<b>XII</b>
<b>Introducción.....</b>	<b>XIV</b>

## INDICE GENERAL

<b>CAPITULO 1 .....</b>	<b>1</b>
<b>1. Diseño de la investigación. ....</b>	<b>1</b>
<b>1.1 Antecedentes de la investigación. ....</b>	<b>1</b>
<b>1.2 Problema de investigación. ....</b>	<b>1</b>
<b>1.2.1 Planteamiento del problema.....</b>	<b>1</b>
<b>1.2.2 Formulación del problema de investigación.....</b>	<b>2</b>
<b>1.2.3 Sistematización del problema de investigación. ....</b>	<b>2</b>
<b>1.3 Objetivos de la investigación. ....</b>	<b>2</b>
<b>1.3.1 Objetivo general. ....</b>	<b>2</b>
<b>1.3.2 Objetivos específicos.....</b>	<b>2</b>
<b>1.4 Justificación de la investigación.....</b>	<b>3</b>
<b>1.4.1 Análisis FODA.....</b>	<b>4</b>
<b>1.5 Marco de referencia de la investigación.....</b>	<b>4</b>
<b>1.5.1 Marco teórico .....</b>	<b>5</b>

1.5.2 Herramientas de Desarrollo.....	35
1.5.3 Acceso.....	38
1.5.5 Marco conceptual.....	68
1.6 Formulación de la hipótesis y variables.....	69
1.6.1 Hipótesis general.....	69
1.6.2 Hipótesis Particulares.....	70
1.6.3 MATRIZ CAUSA – EFECTO.....	70
1.6.4 Variables.....	71
1.6.4.1 Variables independientes.....	72
1.6.4.2 Variables dependientes.....	72
1.7 Aspectos metodológicos de la investigación.....	72
1.7.1 Tipo de estudio.....	72
1.7.2 Método de investigación.....	73
1.7.2.1 Método Observativo.....	73
1.7.2.2 Método Analítico.....	73
1.7.2.3 Método Experimental.....	73
1.7.3 Fuentes y técnicas para la recolección de información.....	73
1.8 Resultados e impactos esperados.....	74
<b>CAPITULO 2.....</b>	<b>74</b>
<b>2. Análisis presentación de resultados y diagnóstico.....</b>	<b>75</b>
2.1 Análisis de la Situación Actual.....	75
2.2 Herramientas de análisis.....	77
2.2.1. Encuestas.....	77
2.2.1.2 Análisis Estadístico de las encuestas.....	78



<b>CAPÍTULO 3</b> .....	<b>82</b>
<b>3. Análisis y diseño del sistema propuesto</b> .....	<b>83</b>
<b>3.1 Arquitectura del Sistema</b> .....	<b>83</b>
<b>3.1.1 Descripción General del Sistema</b> .....	<b>84</b>
<b>3.2 Modelo de Análisis</b> .....	<b>87</b>
<b>3.2.1 Diagrama de Clases</b> .....	<b>87</b>
<b>3.2.2 Diagrama de Actividad</b> .....	<b>88</b>
<b>3.2.3 Diagrama de Despliegue del Software</b> .....	<b>90</b>
<b>3.2.4 Diagramas de Casos de Usos</b> .....	<b>90</b>
<b>3.2.5 Diagramas de Interacción</b> .....	<b>97</b>
<b>3.3 Diseño de la Arquitectura</b> .....	<b>98</b>
<b>3.3.1 Capa de Base de Datos</b> .....	<b>99</b>
<b>3.4 Módulo de Inicio de Sesión de Administrador</b> .....	<b>104</b>
<b>3.5 Descripción del Monitoreo Inalámbrico</b> .....	<b>121</b>
<b>4. CONCLUSIONES</b> .....	<b>123</b>
<b>5. RECOMENDACIONES</b> .....	<b>125</b>
<b>6. BIBLIOGRAFIA</b> .....	<b>126</b>

#### INDICE DE GRÁFICOS

Gráfico 1: FODA .....	4
Gráfico 2: Componentes básicos en un sistema de RFID .....	5
Gráfico 3: Frecuencia de Resonancia de los Tag Pasivos .....	10
Gráfico 4: Lector RFID con modulador EM4095 .....	10
Gráfico 5: Conexión del lector RFID con el Microcontrolador 18F4550 .....	12

Gráfico 6: Diagrama de un Microcontrolador .....	12
Gráfico 7 : Placa con un microcontrolador ATmel. ....	13
Gráfico 8: Diagrama de Bloques de la Compilación .....	17
Gráfico 9: Microcontrolador PICKIT 2.....	18
Gráfico 10: Descripción de Pines del PicKIT2 .....	19
Gráfico 11: Funcionamiento de un Relé .....	20
Gráfico 12: Tipos de Relés .....	21
Gráfico 13: Estructura de un Relé .....	21
Gráfico 14: Diagrama de un sistema microcontrolado.....	22
Gráfico 15: Representación en bloques del microcontrolador .....	23
Gráfico 16: Arquitectura de Von Neumann.....	23
Gráfico 17: Arquitectura Harvard.....	24
Gráfico 18: Distribución de Pines Pic 18F4455 .....	25
Gráfico 19: Configuración del módulo oscilador de 4MHz a 48Mhz .....	27
Gráfico 20: Distribución de Pines PIC 16F887 .....	28
Gráfico 21: Distribución de Pines 18F4550 .....	28
Gráfico 22 : Esquema para la simulación del circuito .....	29
Gráfico 23: Simulación de un tren de pulsos del dispositivo RFID .....	29
Gráfico 24: Simulación Verificando el Bit de Paridad .....	30
Gráfico 25: Monitoreo Inalámbrico.....	31
Gráfico 26: Transceptor inalámbrico RFM12.....	32
Gráfico 27: Configuración de Pines del Transceptor inalámbrico RFM12 .....	32
Gráfico 28: Lector RFid .....	38
Gráfico 29: Gráfico del Circuito .....	40
Gráfico 30: Placa del circuito del Lector .....	41
Gráfico 31: Circuito del Lector .....	41
Gráfico 32 PIC18F4550.....	42

Gráfico 33 Tag RFid Pasivas.....	51
Gráfico 34 Cerradura electromagnética.....	52
Gráfico 35: Cable convertidor USB a RS-232 .....	53
Gráfico 36: Esquema del Dispositivo del Monitoreo .....	54
Gráfico 37: Esquema del Dispositivo del Monitoreo .....	55
Gráfico 38: Placa del Monitoreo .....	55
Gráfico 39: Módulo RFM12 .....	65
Gráfico 40: PIC16F628A .....	66
Gráfico 41: Módulo UART USB .....	67
Gráfico 42: Resistencia de 10 K .....	67
Gráfico 43: Organigrama de la Universidad Politécnica Salesiana .....	77
Gráfico 44: Diagrama de Barras pregunta 1 .....	78
Gráfico 45 Diagrama de Barras pregunta 2.....	79
Gráfico 46: Diagrama de Barras pregunta 2.....	80
Gráfico 47: Diagrama de Barras pregunta 2.....	81
Gráfico 48: Diagrama de Barras pregunta 2.....	82
Gráfico 49: Arquitectura del Sistema .....	83
Gráfico 50: Proceso de ejecución del Sistema .....	86
Gráfico 51: Diagrama de Clases .....	87
Gráfico 52: Diagrama de Actividad – Función Administrador .....	88
Gráfico 53: Diagrama de Actividad – Función Supervisor .....	89
Gráfico 54: Diagrama de Despliegue de Software .....	90
Gráfico 55: Caso de Uso - Autenticación de Usuario .....	92
Gráfico 56: Casos de Usos Módulo Administrador .....	93
Gráfico 57: Módulo Supervisor .....	94
Gráfico 58: Caso de Uso - Ingreso De Docentes .....	95
Gráfico 59: Casos de Uso - Creación de Registro de Horarios .....	95

Gráfico 60: Caso de Uso-Conexión .....	96
Gráfico 61: Casos de Usos - Solicitud de Reportes.....	97
Gráfico 62: Diagrama de Interacción .....	98
Gráfico 63: Capa de Base de Datos .....	99
Gráfico 64: Ingreso al Sistema .....	104
Gráfico 65: Menú Principal - Sistema SCAL.....	111
Gráfico 66: Registro De Usuario - Sistema SCAL .....	111
Gráfico 67: Formulario de Registro SCAL .....	112
Gráfico 68: Almacenamiento de Datos - Sistema SCAL.....	112
Gráfico69: Interfaz Visual del Programa.....	121
Gráfico 70 : Diagrama de Bloques.....	122
Gráfico 71: Codificación del monitoreo.....	122
Gráfico 72: Instrucciones del monitoreo .....	123

#### INDICE DE TABLAS

Tabla 1 Diferencias técnicas entre tecnologías RFID activa y pasiva. ....	9
Tabla 2 Especificación Técnica Lector RFid.....	39
Tabla 3 Especificaciones Técnicas PIC18F4550 .....	42
Tabla 4 Especificaciones Técnicas RFid Pasivas .....	51
Tabla 5 : Especificaciones Técnicas Cerradura Electromagnética.....	52
Tabla 6 : Especificaciones Técnicas Cable convertidor USB a RS-232 .....	53
Tabla 7 Especificaciones Técnicas Módulo RFM12 .....	65
Tabla 8 Especificaciones PIC16F628A .....	66
Tabla 9 Matriz Causa - Efecto .....	71
Tabla 10 Tabulación 1 .....	78
Tabla 11: Tabulación 2 .....	79
Tabla 12: Tabulación 3 .....	79

Tabla 13: Tabulación 4 .....	80
Tabla 14: Tabulación 5 .....	81
Tabla 15: Casos de Uso Autenticación de Usuario .....	93
Tabla 16: Casos de Uso Módulo Administrador .....	93
Tabla 17: Casos de Uso Módulo Supervisor .....	94
Tabla 18: Casos de Usos Ingreso de Docentes .....	95
Tabla 19: Casos de Uso Ingreso de Docentes .....	96
Tabla 20: Casos de Uso Conexión .....	96
Tabla 21: Casos de Uso Solicitud de Reportes .....	97
Tabla 22: Diccionario de Datos – Tabla Usuarios .....	100
Tabla 23: Diccionario de Datos – Marcaciones Diarias de los Usuarios.....	100
Tabla 24: Diccionario de Datos - Designación.....	101
Tabla 25: Diccionario de Datos - Marcación.....	101
Tabla 26: Diccionario de Datos - Ndiarias .....	102
Tabla 27: Diccionario de Datos Reportes .....	102
Tabla 28: Diccionario de Datos – Días trabajados .....	103
Tabla 29: Diccionario de Datos - Equipos .....	103
Tabla 30: Diccionario de Datos Reportes - Grupos .....	104

## INTRODUCCIÓN

La tecnología de RFID es un sistema de autoidentificación inalámbrico, ha tenido mucho auge en los últimos años debido a la relativa reducción de precios en el mercado, consiste en transmitir la identidad de un objeto (similar a un número de serie único) mediante ondas de radio. Las etiquetas RFID (RFID Tag, en inglés) son unos dispositivos pequeños, similares a una pegatina, que pueden ser adheridas o incorporadas a un producto, un animal o una persona. Contienen antenas para permitirles recibir y responder a peticiones por radiofrecuencia desde un emisor-receptor RFID.

El presente proyecto tiene como objetivo principal el desarrollo de un sistema para el laboratorio de Telemática de la Universidad Politécnica Salesiana.

El diseño del sistema consiste en dos partes:

- Acceso
- Monitoreo

El Acceso que se realizó es un seguro método destinado a controlar el ingreso y egreso de los docentes y los encargados del mantenimiento de dicho laboratorio (es decir, controlar el personal). El software de control de acceso permite configurar el hardware desde la PC, y elaborar estadísticas.

El Monitoreo se efectuó en LabVIEW utiliza un módulo de identificación inalámbrica denominado RFID (Radio Frequency Identification) el cual permite identificar una etiqueta electrónica a distancia, que emite periódicamente una señal de radiofrecuencia hacia el módulo lector RFID.

## **CAPITULO 1**

### **1. Diseño de la investigación.**

#### **1.1 Antecedentes de la investigación.**

La Universidad Politécnica Salesiana siempre se ha caracterizado por su calidad de enseñanza, por su desarrollo de la cultura, de la ciencia y tecnología es por esto que el objetivo principal de este proyecto es el de desarrollar un sistema de monitoreo y de control de acceso para poder ofrecer una mejor seguridad en los equipos que se obtienen para seguir manteniendo la excelencia académica.

El sistema que se propone es basado en la tecnología de identificación por radio frecuencia, con este sistema se pretende evitar los hurtos y controlar el acceso al laboratorio ya sea para control de asistencia o para llevar un registro de las personas que accedieron a él, en caso de suceder algún contratiempo.

#### **1.2 Problema de investigación.**

##### **1.2.1 Planteamiento del problema.**

En la actualidad, han surgido nuevas tecnologías que la sociedad ha ido adoptando para obtener un mayor desempeño en las tareas diarias.

La Universidad Politécnica Salesiana está en constante cambio, puesto que esto ha ayudado a la calidad del servicio académico y para seguir en esa transición institucional, se ha detectado que los docentes en su horario de clases, necesitan los laboratorios.

El dilema de ésta deducción es que el docente pierde tiempo en buscar al encargado de abrir las puertas de dicho laboratorio, además no se tiene la debida seguridad puesto que al terminar su clase estos quedan abiertos hasta que el docente de la siguiente clase lo vuelva a ocupar.

Es por este motivo y también por la falta de seguridad en las instalaciones, que han ocurrido pérdidas de equipos que son necesarios para seguir prestando el servicio a los estudiantes.

La solución más factible es diseñar e implementar un sistema para controlar y monitorear mediante la tecnología RFID los laboratorios y a su vez facilitar el acceso del personal docente para el dictado de clases.

### **1.2.2 Formulación del problema de investigación.**

¿Cómo optimizar el acceso y seguridad de los equipos del laboratorio mediante un sistema de control de acceso y monitoreo RFID?

### **1.2.3 Sistematización del problema de investigación.**

¿Cuál es la medida que se debe de aplicar para que los docentes no utilicen su tiempo completo?

¿Deben los docentes perder tiempo en sus actividades y no aprovechar sus horas al máximo?

¿Cómo facilitar el acceso de los docentes al laboratorio?

¿Por qué hay inseguridad en los laboratorios si estos quedan abiertos durante horas?

¿Qué consecuencia tiene la inseguridad y la pérdida de tiempo en los laboratorios?

¿Cómo podríamos mejorar la seguridad en los laboratorios y la optimización del tiempo de los docentes?

## **1.3 Objetivos de la investigación.**

### **1.3.1 Objetivo general.**

Optimizar el control de acceso a los laboratorios garantizando la estadía de los equipos en el área de trabajo mediante el monitoreo constante, utilizando la tecnología RFID para administrar eficientemente horario de entradas y salidas de los docentes y mejorando la eficacia de la cátedra.

### **1.3.2 Objetivos específicos.**

- ❖ Implementar dispositivos que permitan el uso seguro y adecuado a los laboratorios.



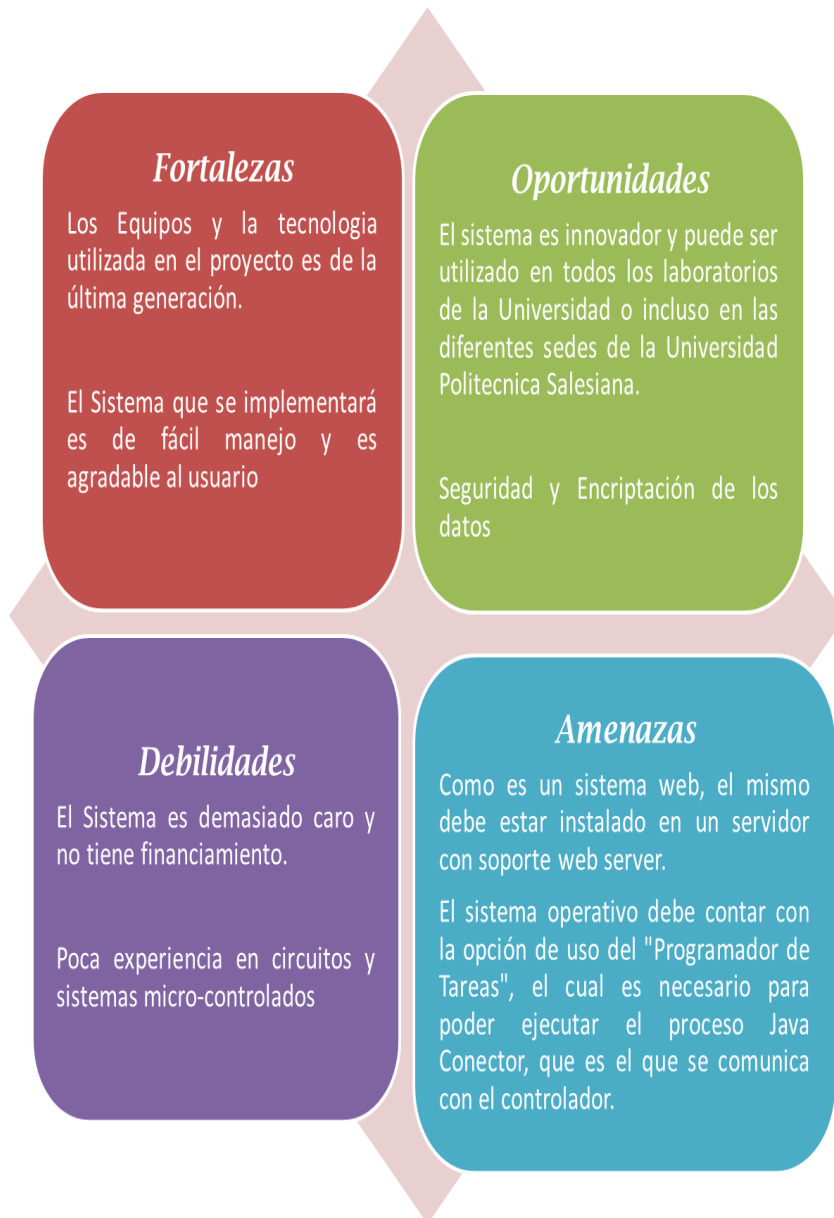
- ❖ Facilitar el control de ingreso y acceso al laboratorio de acuerdo a los horarios establecidos en el sistema académico.
- ❖ Optimizar el tiempo de clase al apertura oportunamente los laboratorios
- ❖ Determinar las razones y las consecuencias de la inseguridad en los laboratorios.
- ❖ Implementar una aplicación que permitan dar agilidad, seguridad y control de los laboratorios a través de la identificación de Radiofrecuencia.
- ❖ Ejecutar el sistema de control de acceso y monitoreo a los laboratorios.

#### **1.4 Justificación de la investigación.**

En la actualidad la Universidad Politécnica Salesiana ha crecido en infraestructura como en el número de estudiantes; por lo que sería necesario aplicar los conocimientos que se han impartido de docente a estudiante y poner en práctica lo aprendido, como también contribuir a la Universidad y en especial el Departamento de Sistemas en la implementación de un sistema de identificación automática por radiofrecuencia para así identificar, gestionar, controlar y localizar al personal docente puesto que RFID tiene varias aplicaciones y beneficios ya que mediante ésta tecnología se puede tener precisión a la hora de necesitar los laboratorios y ayudarnos a la reducción de los esfuerzos de supervisión por parte del personal de mantenimiento y limpieza de la Institución.

Ésta propuesta no sólo pretende ayudar, sino dejar un precedente en el manejo de la seguridad de los equipos de trabajo que son adquiridos para seguir brindando la excelencia académica de lo que siempre se caracteriza la Universidad Politécnica Salesiana.

### 1.4.1 Análisis FODA



**Gráfico 1: FODA**

**FUENTE: EL AUTOR**

## 1.5 Marco de referencia de la investigación

### 1.5.1 Marco teórico

#### *Tecnología RFID.*

#### ¿Qué es la identificación por radiofrecuencia?

El sistema de identificación por radiofrecuencia o RFID (Radio Frequency Identification) es un sistema inalámbrico de almacenamiento y recuperación de datos que usa ondas de radio para determinar la identificación de pequeños dispositivos denominados etiquetas o tags RFID.

Existen 3 componentes básicos en un sistema de RFID:

1. **El tag, etiqueta o transponder de RFID** consiste en un pequeño circuito, integrado con una pequeña antena, capaz de transmitir un número de serie único hacia un dispositivo de lectura, como respuesta a una petición. Algunas veces puede incluir una batería.
2. **El lector**, (el cual puede ser de lectura o lectura/escritura) está compuesto por una antena, un módulo electrónico de radiofrecuencia y un módulo electrónico de control.
3. **Un controlador o un equipo anfitrión**, comúnmente una PC o Workstation, en la cual corre una base de datos y algún software de control.

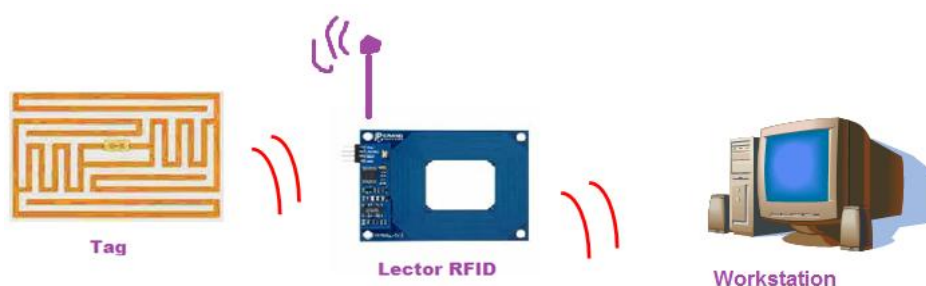


Gráfico 2: Componentes básicos en un sistema de RFID

Fuente: El Autor

## ***Clasificación de la tecnología RFID***

Las tecnologías de auto identificación por radio frecuencia se clasifican en 3 tipos según el tipo del tag:

- ✓ **Sistemas pasivos:** En los cuales las etiquetas de RFID no cuentan con una fuente de poder. Su antena recibe la señal de radiofrecuencia enviada por el lector y almacena ésta energía en un capacitor. La etiqueta utiliza ésta energía para habilitar su circuito lógico y para regresar una señal al lector. Éstas etiquetas pueden llegar a ser muy económicas y pequeñas, pero su rango de lectura es muy limitado.
  
- ✓ **Sistemas activos:** Utilizan etiquetas con fuentes de poder integradas, como baterías. Este tipo de etiquetas integra una electrónica más sofisticada, lo que incrementa su capacidad de almacenamiento de datos, interfaces con sensores, funciones especializadas, además de que permiten que exista una mayor distancia entre lector y etiqueta (20m a 100m). Este tipo de etiquetas son más costosas y tienen un mayor tamaño.
  
- ✓ **Sistemas Semi-Activos:** Emplean etiquetas que tienen una fuente de poder integrada, la cual energiza al tag para su operación, sin embargo, para transmitir datos, una etiqueta semi-activa utiliza la potencia emitida por el lector.

En este tipo de sistemas, el lector siempre inicia la comunicación. La ventaja de Éstas etiquetas es que al no necesitar la señal del lector para energizarse (a diferencia de las etiquetas pasivas), pueden ser leídas a mayores distancias, y como no necesita tiempo para energizarse, Éstas etiquetas pueden estar en el rango de lectura del lector por un tiempo substancialmente menor para una apropiada lectura. Esto permite obtener lecturas positivas de objetos moviéndose a altas velocidades.

## **Estándares que usa la Tecnología**

Existen algunos tipos de estándares pero para el presente trabajo necesitaremos los siguientes:

- a) **El estándar ISO 14443.** Tarjetas de identificación, circuitos integrados para tarjetas sin contactos. **ISO 15693.** Tarjetas de proximidad de hasta 1.5 m. **ISO 18000.** Información tecnológica. RFID para gestión de objetos.
- b) **Estándar EPCGlobal Gen2** Es una regulación que permite una compatibilidad mundial de este protocolo en banda UHF<sup>1</sup>. Mediante este estándar podemos seleccionar tags como si estuviéramos accediendo a una base de datos. Esto implica que si tengo una producción completa de millones de objetos ya empaquetados, y necesito localizar una parte de mi producción pequeña, por ejemplo, podré ordenar al reader que sólo seleccione dichos tags. Luego un tag no será más que un objeto en una base de datos que podemos leer, escribir; es decir, modificarlo de estado.
- c) **ANSI** Se trata de un organismo privado con fines no lucrativos que administra y coordina el organismo de estándares americano.
- d) **ONS Similar** a (DNS) organismo.

## **Frecuencias de la Tecnología**

Las frecuencias de RFID pueden ser divididas en 4 rangos:

- 1) **Baja Frecuencia (9-135 KHz).** Los sistemas que utilizan este rango de frecuencia tienen la desventaja de una distancia de lectura de sólo unos cuantos centímetros. Sólo pueden leer un elemento a la vez.
- 2) **Alta Frecuencia (13.56 MHz).** Ésta frecuencia es muy popular y cubre distancias de 1cm a 1.5 m. Típicamente las etiquetas que trabajan en ésta frecuencia son de tipo pasivo.
- 3) **Ultra High Frequency (0.3-1.2GHz).** Este rango se utiliza para tener una mayor distancia entre la etiqueta y el lector (de hasta 4 metros, dependiendo del fabricante y del ambiente). Estas frecuencias no pueden penetrar el metal ni los líquidos a diferencia de las bajas frecuencias pero

---

<sup>1</sup> UHF: Ultra High Frequency, 'frecuencia ultra alta'

pueden transmitir a mayor velocidad y por lo tanto son buenos para leer más de una etiqueta a la vez.

- 4) **Microondas (2.45-5.8GHz).** La ventaja de utilizar un intervalo tan amplio de frecuencias es su resistencia a los fuertes campos electromagnéticos, producidos por motores eléctricos, por lo tanto, estos sistemas son utilizados en líneas de producción de automóviles. Sin embargo, Estas etiquetas requieren de mayor potencia y son más costosas, pero es posible lograr lecturas a distancias de hasta 6 metros.

### ***Características técnicas de RFID Activo y Pasivo***

RFID Activo y Pasivo son tecnologías fundamentalmente diferentes. Mientras los dos usan energía de radio frecuencia para comunicarse entre una etiqueta y un lector, el método de activar las etiquetas es diferente. RFID activo usa una fuente de poder interna (batería) dentro de la etiqueta para activar continuamente la etiqueta y su circuitería de comunicación RF, considerando que RFID Pasivo confía en energía de RF transferida del lector a la etiqueta para activar la etiqueta.

Mientras ésta distinción puede parecer menor en la superficie, su impacto en la funcionalidad del sistema es significativo. RFID pasivo hace cualquiera 1) refleja la energía del lector o 2) absorbe y temporalmente guarda una cantidad muy pequeña de energía de la señal del lector para generar su propia respuesta rápida. En cualquier caso, el funcionamiento de RFID Pasivo requiere señales muy fuertes del lector, y la fuerza de la señal devuelta de la etiqueta es forzada a niveles muy bajos por la energía limitada. Por otro lado, RFID Activo permite señales de muy bajo nivel para ser recibidos por la etiqueta (porque el lector no necesita activar la etiqueta), y la etiqueta puede generar señales de alto nivel controladas desde su fuente de poder interna, para devolver al lector.

Adicionalmente, la etiqueta de RFID Activa es continuamente activada, si está en el campo del lector o no. Éstas diferencias afectan el rango de comunicación, capacidad de recolección multietiqueta, habilidad para agregar sensores y datos registrados, y muchos otros parámetros funcionales.

	<b>RFID Activa</b>	<b>RFID Pasiva</b>
<b>Fuente de poder de las etiquetas</b>	Interna de la tarjeta	Energía transferida desde el lector vía RF
<b>Batería de las etiquetas</b>	Sí	No
<b>Disponibilidad de alimentación para etiquetas</b>	Continua	Sólo dentro del campo del lector
<b>Potencia de la señal requerida desde el lector a la etiqueta</b>	Baja	Alta (puede activar la etiqueta)
<b>Potencia de la señal disponible desde la etiqueta al lector</b>	Alta	Baja

Tabla 1 Diferencias técnicas entre tecnologías RFID activa y pasiva.

Fuente: Escuela Politécnica Nacional

### **Aplicabilidad de RFID Activo y Pasivo para activos fijos del laboratorio.**

Basados en la funcionalidad que provee cada tecnología, RFID Activo y Pasivo dirigen diferentes, pero a menudo complementarios aspectos de la visibilidad de los activos fijos del laboratorio. RFID Pasivo es el más apropiado cuando el movimiento de los activos es altamente consistente y controlado, y cuando es requerida una capacidad de almacenamiento pequeña, sin seguridades.

### ***Tecnología RFID Pasiva y desarrollo de la misma***

La señal que les llega de los lectores induce una corriente eléctrica pequeña y suficiente para operar el circuito integrado CMOS del tag, de forma que puede generar y transmitir una respuesta. La mayoría de tags pasivos utiliza backscatter<sup>2</sup> sobre la portadora recibida; esto es, la antena ha de estar diseñada para obtener la energía necesaria para funcionar a la vez que para transmitir la respuesta por backscatter.

<sup>2</sup> "backscatter" una señal no modulada es transmitida por la Antena en dirección al TAG.

El propósito fundamental de la tecnología RFID es transmitir la identidad de un objeto (similar a un número de serie único) mediante ondas de radio. Las tecnologías RFID se agrupan dentro de las denominadas Auto ID

Los tag Rfid pasivos con frecuencia de resonancia de 125 Khz poseen las siguientes características:

- ✓ 9 bits de cabecera
- ✓ 14 bits de paridad
- ✓ 40 bits de datos
- ✓ 1 bit de parada

1	1	1	1	1	1	1	1	1
	ROW0	D00	D01	D02	D03	P0		
	ROW1	D10	D11	D12	D13	P1		
	ROW2	D20	D21	D22	D23	P2		
	ROW3	D30	D31	D32	D33	P3		
	ROW4	D40	D41	D42	D43	P4		
	ROW5	D50	D51	D52	D53	P5		
	ROW6	D60	D61	D62	D63	P6		
	ROW7	D70	D71	D72	D73	P7		
	ROW8	D80	D81	D82	D83	P8		
	ROW9	D90	D91	D92	D93	P9		
	PC0	PC1	PC2	PC3	0			

Gráfico 3: Frecuencia de Resonancia de los Tag Pasivos

Por este motivo nuestro microcontrolador debe ser capaz de interpretar cuales son los datos y cuáles son los bits de paridad, estos últimos sirven para descartar errores. Sobre los bits de cabecera sirven para alertar al microcontrolador que se va a recibir una serie de bits que representan la información que nos interesa, el bit de parada ésta para establecer el fin de la transmisión.

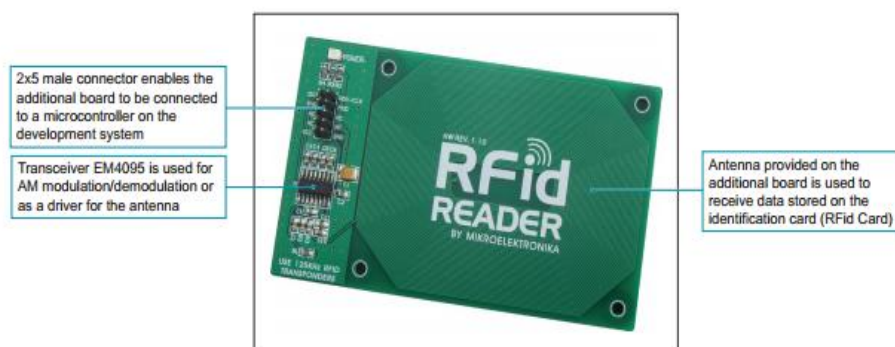


Gráfico 4: Lector RFID con modulador EM4095

Fuente: Microelectrónica



Para la implementación usaremos tags de sólo de lectura es decir el código de identificación que contiene es único y es personalizado durante la fabricación de la etiqueta.

Este lector Rfid posee internamente un modulador<sup>3</sup> que genera una modulación AM, ésta modulación de la señal es a una portadora de 125Khz, como las dimensiones de la antena serían muy grandes ésta se encuentra enrollada en la misma PCB<sup>4</sup>. Permitiendo así disminuir el tamaño del lector rfid. Ésta antena es la que recibe la señal del tag pasivo que responde cuando se ve polarizado por la energía que irradia dicha antena.

Este módulo trae consigo 4 pines que debemos describir:

1. OUT: Señal de salida desde el circuito que contiene el Em4095 (El código de lectura de la tarjeta de identificación).
2. RDY/CLK: Reloj de sincronización y bandera de activación.
3. SHD: Para establecer el modo Sleep de bajo consumo de energía
4. MOD: Para empezar la modulación en la antena.

El esquema sencillo pero sin la fuente de alimentación sería algo así:

El lector Rfid está conectado directamente a nuestro microcontrolador y es quien recibe la señal para luego enviarla al computador para su procesamiento.

---

<sup>3</sup> Modulador conocido con el nombre EM4095.

<sup>4</sup> PCB: Printed Circuit Board; tarjeta de circuito impreso.

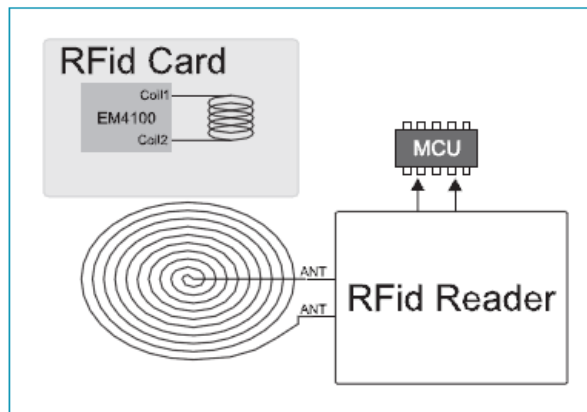


Gráfico 5: Conexión del lector RFID con el Microcontrolador 18F4550

Fuente: Mikrotik

### Microcontroladores

Explicando mediante términos sencillos, podemos definir a un Microcontrolador como un circuito integrado (chip) que incluye en su interior las tres partes fundamentales de una computadora: CPU, memoria y Unidades de E/S, es decir, se trata de una computadora completa en un sólo circuito integrado.

Explicando de forma técnica tenemos que el microcontrolador es un circuito integrado que en su interior contiene una unidad central de procesamiento (CPU), unidades de memoria (RAM y ROM), puertos de entrada y salida y periféricos. Estas partes están interconectadas

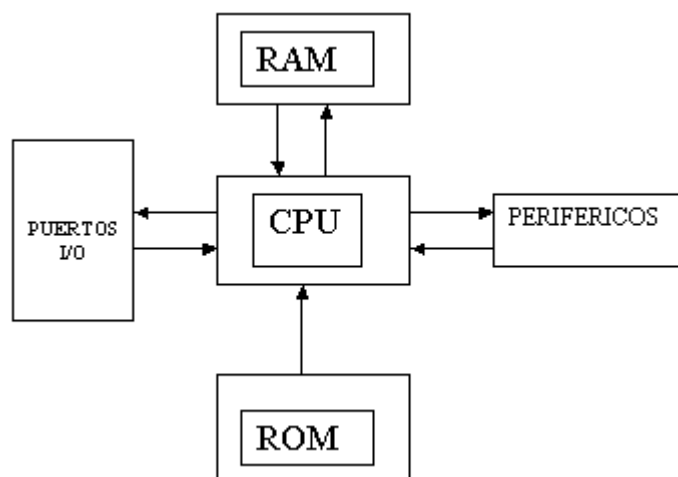


Gráfico 6: Diagrama de un Microcontrolador

Fuente: Electrónica Estudio

dentro del microcontrolador, y en conjunto forman lo que se le conoce como microcomputadora.



**Gráfico 7 : Placa con un microcontrolador ATmel.**

**Fuente: ATmel.**

Se puede decir con toda propiedad que un microcontrolador es una microcomputadora completa encapsulada en un circuito integrado.

Un microcontrolador difiere de un microprocesador de los que estamos acostumbrados a ver (por ejemplo un Athlon o un Pentium), no sólo físicamente si no en que es más fácil convertirlo en una “computadora” en funcionamiento, con un mínimo de chips externos de apoyo.

La idea es que el chip se coloque en el dispositivo, enganchado a la fuente de energía y de información (por medio de sus puertos de entrada/salida integrados) que necesite, y eso es todo. Un microprocesador convencional no le permitirá hacer esto, ya que normalmente requiere que todas estas tareas sean manejadas por otros chips.

Típicamente, un microcontrolador puede disponer de un generador de reloj integrado y una pequeña cantidad de memoria RAM y ROM/EPROM/EEPROM, significando que para hacerlo funcionar, todo lo que se necesita son unos pocos

programas de control y un cristal de sincronización. Los microcontroladores disponen generalmente también de una gran variedad de dispositivos de entrada/salida, como convertidores de analógico a digital, temporizadores, UARTs<sup>5</sup> y buses de interfaz serie especializados, como I2C<sup>6</sup> y CAN<sup>7</sup>, entre otros. Frecuentemente, estos dispositivos integrados pueden ser controlados por instrucciones de procesadores especializados. Los modernos microcontroladores a veces incluyen un lenguaje de programación integrado, como el BASIC que se utiliza bastante con este propósito. En caso de que no dispongan de un intérprete “on-board”, se pueden programar desde el ordenador, usando también alguna versión de Basic, C, Pascal, MiKroC, assembler u otros dentro de la larga lista de lenguajes disponibles, muchas veces en forma gratuita.

Los microcontroladores negocian la velocidad y la flexibilidad para facilitar su uso. Debido a que se utiliza bastante sitio en el chip para incluir funcionalidad, como los dispositivos de entrada/salida o la memoria que incluye el microcontrolador, se ha de prescindir de cualquier otra circuitería. Incluso en una de las familias más difundidas, sólo se dispone de un set de 35 instrucciones básicas.

### ***Características de los Microcontroladores***

El propósito fundamental de los microcontroladores es el de leer y ejecutar los programas que el usuario le escribe, es por esto que la programación es una actividad básica e indispensable cuando se diseñan circuitos y sistemas que los incluyan. El carácter programable de los microcontroladores simplifica el diseño

---

<sup>5</sup> UART: Universal Asynchronous Receiver-Transmitter

<sup>6</sup> I2C: Inter Integrated Circuits Bus de Comunicaciones Serial Síncrono

<sup>7</sup> CAN: Controller Area Network

de circuitos electrónicos. Permiten modularidad y flexibilidad, ya que un mismo circuito se puede utilizar para que realice diferentes funciones con sólo cambiar el programa del microcontrolador.

### ***Aplicaciones de los Microcontroladores***

Las aplicaciones de los microcontroladores son vastas, se puede decir que sólo están limitadas por la imaginación del usuario. Es común encontrar microcontroladores en campos como la robótica y el automatismo, en la industria del entretenimiento, en las telecomunicaciones, en la instrumentación, en el hogar, en la industria automotriz, etc.

### ***¿Cómo se escriben los programas?***

Los microcontroladores están diseñados para interpretar y procesar datos e instrucciones en forma binaria. Patrones de 1's y 0's conforman el lenguaje máquina de los microcontroladores, y es lo único que son capaces de entender. Estos 1's y 0's representan la unidad mínima de información, conocida como bit, ya que sólo puede adoptar uno de dos valores posibles: 0 ó 1.

La representación de datos, instrucciones y señales en forma de bits resulta dificultosa y tediosa para aquellas personas que no están familiarizadas con el sistema de numeración binario. Aún para los usuarios expertos no resulta tan evidente la interpretación de instrucciones en forma binaria o lenguaje máquina (el lenguaje máquina se le conoce también como lenguaje de bajo nivel debido a que las instrucciones no son propias del lenguaje humano). Es por esto que la programación comúnmente se lleva a cabo en un lenguaje de alto nivel, es decir, un lenguaje que utilice frases o palabras semejantes o propias del lenguaje humano. Las sentencias de los lenguajes de alto nivel facilitan enormemente la programación ya que son familiares a nuestra manera de

comunicarnos. Lenguajes como el C o BASIC son comúnmente utilizados en la programación de microcontroladores.

Otro tipo de lenguaje más especializado es el lenguaje ensamblador. El lenguaje ensamblador es una lista con un limitado número instrucciones a las cuales puede responder un microcontrolador. Estas instrucciones son palabras o abreviaciones que representan las instrucciones en lenguaje máquina del microcontrolador.

Las instrucciones en lenguaje ensamblador, también conocidas como nemotécnicos, son fáciles de entender y permiten operar directamente con los registros de memoria así como con las instrucciones intrínsecas del microcontrolador. Es por esto que el lenguaje ensamblador es sin lugar a dudas el lenguaje por excelencia en la programación de microcontroladores, ya que permite hacer un uso eficiente de la memoria y minimizar el tiempo de ejecución de un programa.

Cualquiera que sea el lenguaje que se utilice en la programación de microcontroladores, es lo más recomendable profundizar en su arquitectura interna, ya que con este conocimiento se pueden aprovechar más y mejor las capacidades de un microcontrolador dado.

### ***Compilación***

Todo programa escrito en un lenguaje de alto nivel debe ser transformado en código máquina. Los programas que escribimos los entendemos nosotros, no así el microcontrolador.

Un software de computadora, llamado compilador, traduce y transforma nuestro programa en código máquina, que es lo que realmente puede leer e interpretar el microcontrolador.

Una vez compilado el programa, es momento de transferir nuestro código máquina hacia la memoria interna del microcontrolador, usualmente hacia la ROM. Para ésta tarea se utiliza un programador físico, que es una pieza de hardware que tiene el propósito de escribir el programa en la memoria interna del micro.



**Gráfico 8: Diagrama de Bloques de la Compilación**

**Autor: Electrónica Estudio**

### ***Microcontroladores PIC®***

Los PIC, de Microchip, son una opción más dentro del variado mercado de microcontroladores. La popularidad de estos micros radica en su alta disponibilidad en el mercado y bajo precio.

El fabricante ha procurado una difusión exhaustiva de información relativa a sus productos, lo cual ha traído como consecuencia un proliferado uso de este tipo de microcontroladores. Algunos de los profesionales y aficionados que los

utilizan difunden sus desarrollos e inventos por Internet lo cual ha promovido su uso. Muchos consideran que los PIC son los más fáciles de programar.

Por otro lado, se han desarrollado una serie de herramientas de bajo costo por parte de terceros (empresas, profesionales y aficionados), como son programadores, software, etc., que facilitan el uso y programación de estos dispositivos.

Compiladores de C y Basic están disponibles para programar a los PIC y de reciente aparición son los PICAXE, que es un sistema que permite al usuario implementar una función con microcontrolador PIC sin siquiera conocer las instrucciones ni la arquitectura propia del microcontrolador.

Actualmente Electrónica Estudio, ofrece cursos especializados en estos microcontroladores. Dividido en varios niveles los cursos profundizan en la arquitectura de estos dispositivos, la programación y la aplicación de sistemas que los incluyan. Lo invitamos, querido lector a tomar los cursos, con los cuales usted aprenderá a programar y diseñar sistemas con microcontroladores PIC.

### ***Microcontrolador Pickit 2***

El microcontrolador programador PicKit2, es una herramienta de programación para desarrollo de bajo costo. Es capaz de programar la mayoría de los microcontroladores y memorias seriales EEPROM de Microchip.



**Gráfico 9: Microcontrolador PICKIT 2**

**Fuente: Microelectrónica**

### ***Conexión del puerto USB:***



El puerto de conexión USB, es un conector del tipo mini-B. Conecta el PicKit2 a la PC usando el cable suministrado.

### **LEDs de estado:**

Los leds de estado indican el estado del PicKit2.

**1 - Power (Verde).**- La alimentación es aplicada al PicKit2 vía USB.

**2 - Target (Amarillo).**- El PicKit2 está alimentando un dispositivo.

**3 - Busy (Rojo).**- El PicKit2 está ocupado con una función en progreso, tal como la programación.

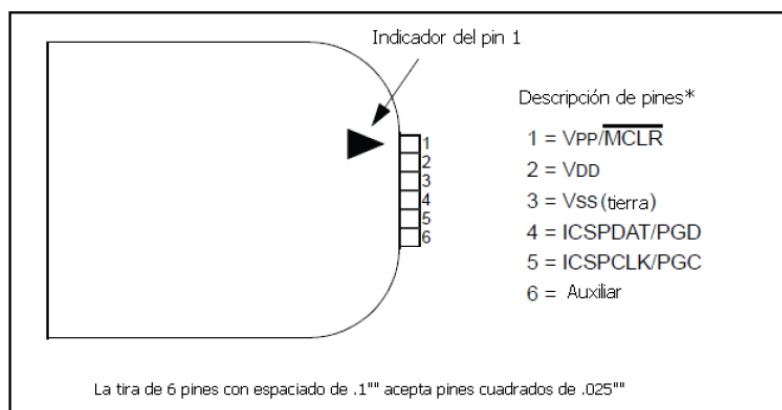
### **Pulsador**

El pulsador puede ser usado para iniciar la programación de un dispositivo cuando la función Programmer>Write on PICkit Button es seleccionada del menú del programa de aplicación del PicKit2.

El botón también puede ser utilizado para poner al sistema operativo del Pickit2 en modo gestor de arranque. Para más información de ésta prestación vea Actualizando el sistema operativo del PicKit2.

### **Conector de programación:**

El conector de programación es de 6 pines con un espacio entre pines de .1" o .25mm que se conecta al dispositivo a programar.



**Gráfico 10: Descripción de Pines del PickIT2**

**Fuente: Nextiafenix**

### **El relé**

## ***Principio de funcionamiento***

Un relé es un interruptor accionado por un electroimán. Un electroimán está formado por una barra de hierro dulce, llamada núcleo, rodeada por una bobina de hilo de cobre (Fig. 1). Al pasar una corriente eléctrica por la bobina (Fig. 2) el núcleo de hierro se magnetiza por efecto del campo magnético producido por la bobina, convirtiéndose en un imán tanto más potente cuanto mayor sea la intensidad de la corriente y el número de vueltas de la bobina. Al abrir de nuevo el interruptor y dejar de pasar corriente por la bobina, desaparece el campo magnético y el núcleo deja de ser un imán.

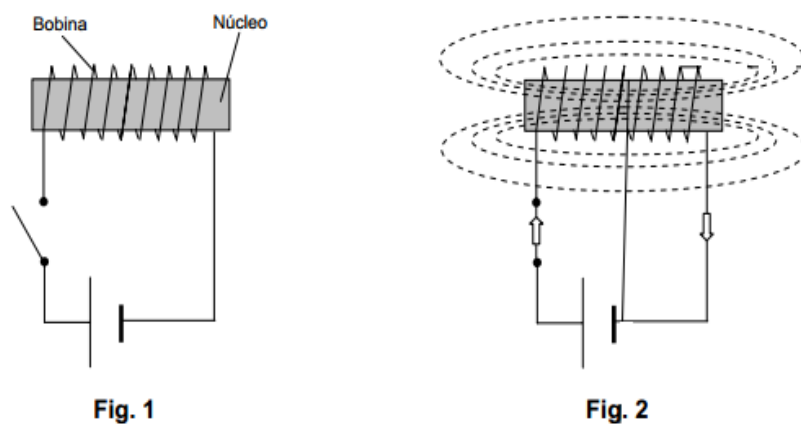


Gráfico 11: Funcionamiento de un Relé

Fuente: Documentos de scribd

## ***Tipos de relés***

Entre los tipos de relés tenemos los siguientes:

- Relés electromecánicos:
  - A) Convencionales.
  - B) Polarizados.
  - C) Reed inversores.
- Relés híbridos.
- Relés de estado sólido.

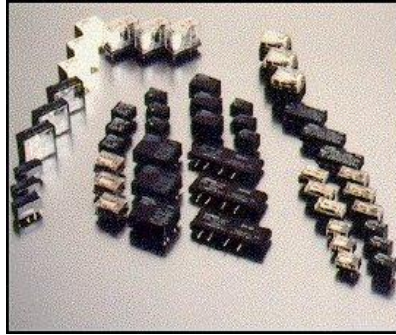


Gráfico 12: Tipos de Relés

Fuente: Electronicaugr

### Estructura de un relé

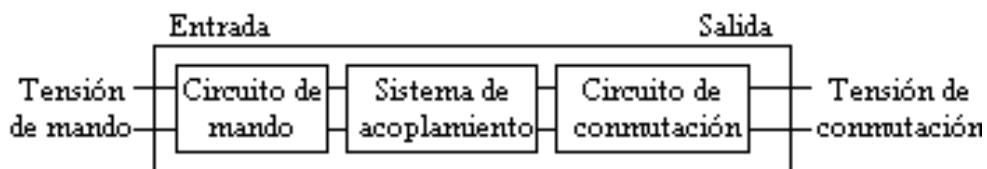


Gráfico 13: Estructura de un Relé

Fuente: Relés

En general, podemos distinguir en el esquema general de un relé los siguientes bloques:

- Circuito de entrada, control o excitación.
- Circuito de acoplamiento.
- Circuito de salida, carga o maniobra, constituido por:
  - circuito excitador.
  - dispositivo conmutador de frecuencia.
  - protecciones.

### ***Características generales***

Las características generales de cualquier relé son:

- El aislamiento entre los terminales de entrada y de salida.
- Adaptación sencilla a la fuente de control.
- Posibilidad de soportar sobrecargas, tanto en el circuito de entrada como en el de salida.
- Las dos posiciones de trabajo en los bornes de salida de un relé se caracterizan por:
  - En estado abierto, alta impedancia.
  - En estado cerrado, baja impedancia.

### **Sistemas Microcontrolados**

El diagrama de un sistema microcontrolado sería algo así:



**Gráfico 14: Diagrama de un sistema microcontrolado.**

**Fuente: Luis Xbot**

Los dispositivos de entrada pueden ser un teclado, un interruptor, un sensor, etc.

Los dispositivos de salida pueden ser LED's, pequeños parlantes, zumbadores, interruptores de potencia (tiristores, optoacopladores), u otros dispositivos como relés, luces, un secador de pelo, etc.

Aquí tenemos una representación en bloques del microcontrolador, para darnos cuenta o tener una idea, y poder observar que lo adaptamos tal y cual es un ordenador, con su fuente de alimentación, un circuito de reloj y el chip microcontrolador, el cual dispone de su CPU, sus memorias, y por supuesto, sus puertos de comunicación listos para conectarse al mundo exterior.

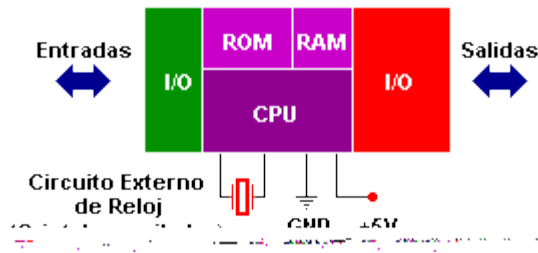


Gráfico 15: Representación en bloques del microcontrolador

Fuente: Luis Xbot

Definamos entonces al microcontrolador: es un circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria. Está compuesto de varios bloques funcionales, los cuales cumplen una tarea específica. Sus partes básicamente son las siguientes:

- Memoria ROM (Memoria de sólo lectura)
- Memoria RAM (Memoria de acceso aleatorio)
- Líneas de entrada/salida (I/O) También llamados puertos
- Lógica de control Coordina la interacción entre los demás bloques

### Arquitectura interna del PIC:

Hay dos arquitecturas conocidas; la clásica de Von Neumann, y la arquitectura Harvard.

**Arquitectura Von Neumann** Dispone de una sola memoria principal donde se almacenan datos e instrucciones de forma indistinta. A dicha memoria se accede a través de un sistema de buses único (direcciones, datos y control).

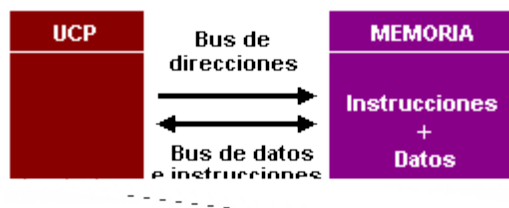


Gráfico 16: Arquitectura de Von Neumann

Fuente: Luis Xbot

**Arquitectura Harvard** Dispone de dos memorias independientes, una que contiene sólo instrucciones, y otra que contiene sólo datos. Ambas disponen de sus respectivos sistemas de buses de acceso y es posible realizar operaciones de acceso (lectura o escritura) simultáneamente en ambas memorias, ésta es la estructura para los PIC's.

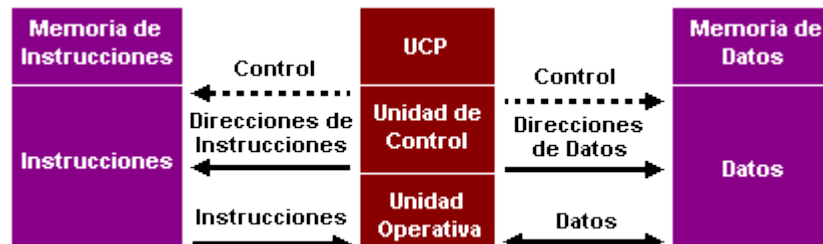


Gráfico 17: Arquitectura Harvard

Fuente: Luis Xbot

## PIC 18F4550

### Características:

- ✓ 77 instrucciones
- ✓ Pila de 32 niveles
- ✓ Hasta 3968 bytes de RAM y 1kB de EEPROM
- ✓ Frecuencia máxima de hasta 40Mhz
- ✓ Múltiples fuentes de interrupción
- ✓ Conectividad directa USB

Lo que necesitamos de un microcontrolador que pueda trabajar a una frecuencia superior a 32Mhz ya que así lo recomienda el fabricante del módulo Rfid y que tenga más de una fuente de interrupción .

Según la hoja de datos de este integrado la distribución de pines sería la siguiente:

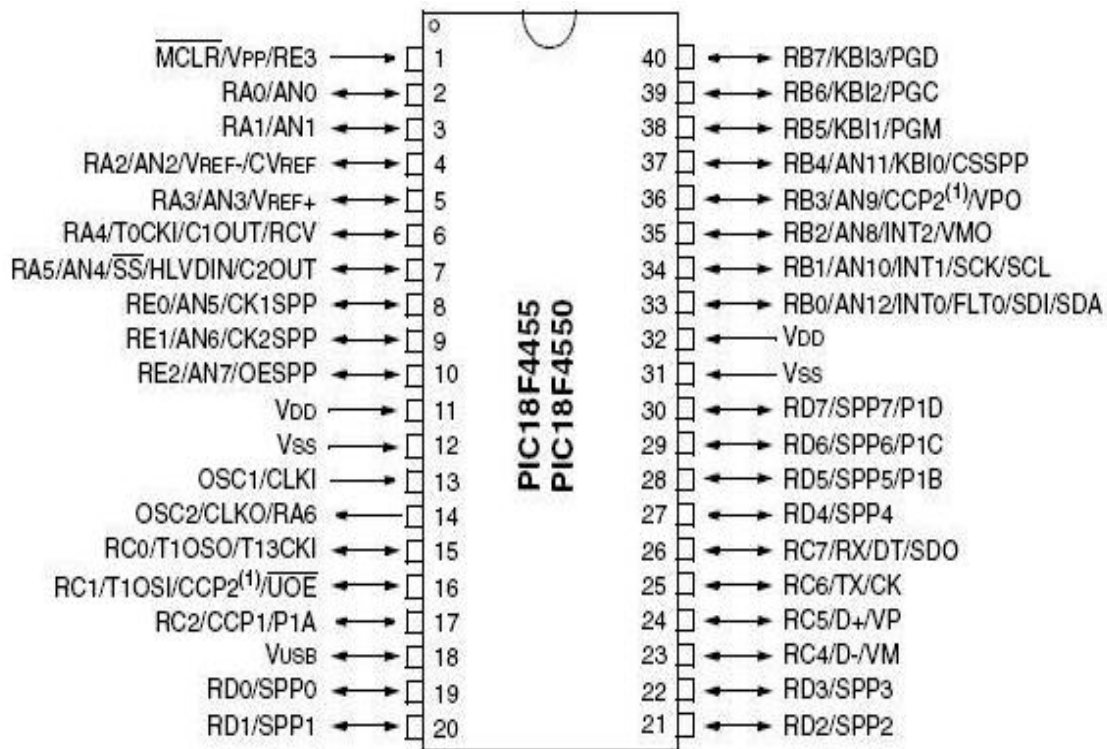


Gráfico 18: Distribución de Pines Pic 18F4455

Fuente: Picmania

Los pines de este microcontrolador están divididos por puertos de entrada y salida desde el puerto A que lo podemos identificar por sus 8 pines que van desde RA0 hasta R07 y el último puerto E. Así también cada puerto tiene distintas cualidades especiales en este caso el puerto C, es para comunicaciones seriales que es la que vamos a usar para hacer la transmisión de la información con el computador, el puerto B, que es para interrupciones externas que es por donde vamos a adquirir la información que viene del dispositivo RFid.

Nótese por ejemplo el pin RB0 también tiene una etiqueta que dice INT0, ésta significa interrupción 0.

Una interrupción, es un evento que hace que el microcontrolador deje de ejecutar la tarea que está realizando para atender dicho acontecimiento y luego regrese y continúe la tarea que estaba realizando antes de que se presentara la interrupción y gracias a ésta cualidad que tienen los microcontroladores no importa el momento en que se le envié el dato desde el dispositivo Rfid el siempre lo atenderá.

Externamente a los pics se los conecta con un oscilador para que hagan Mhz en el mercado local, pero este pic tiene una particular cualidad que es multiplicar su frecuencia interna, recuérdese que dijimos que necesitamos 32Mhz como mínimo.

### ***Configuración del módulo oscilador de 4MHz a 48Mhz***

Lo primero que debemos tomar en cuenta son los buses que se van a usar y la configuración del oscilador forma parte de ello. No es tan difícil, si se ve como si fuera el juego del laberinto, se pueden activar los bits de los registros necesarios y así configurar el oscilador para el cristal que usemos y el tiempo del ciclo de instrucción.

Como se observa en la siguiente imagen debemos configurarlo como indica la línea roja



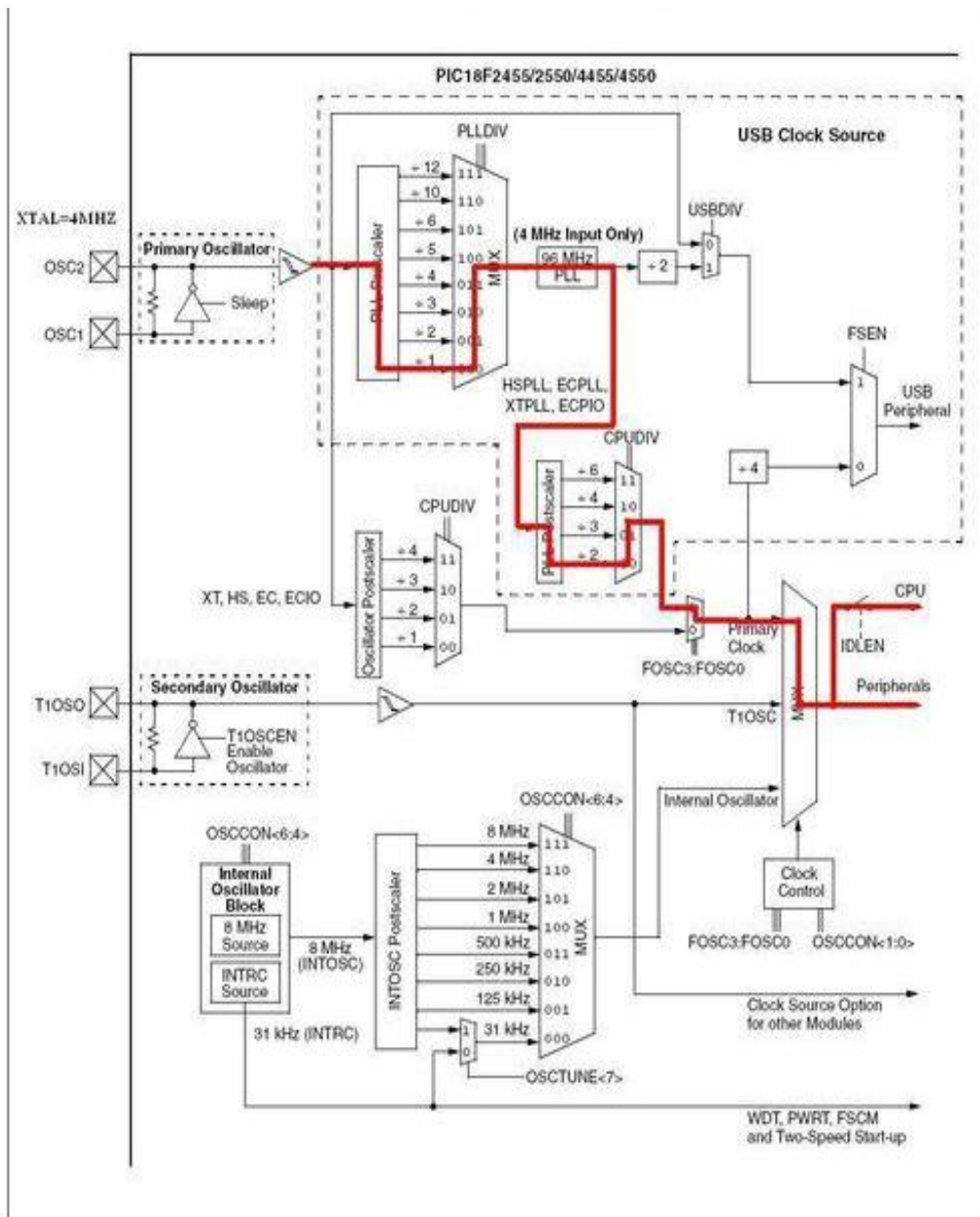


Gráfico 19: Configuración del módulo oscilador de 4MHz a 48Mhz

Fuente: Picmania.

De ésta manera se tienen 48Mhz de oscilación y es por ésta razón que no se puede usar pics de una gama más baja ya que ellos apenas llegan a 20Mhz como máximo.

### Comunicación Serial

Para establecer la comunicación con la pc existen dos posibles soluciones ya que este pic lo permite:

- ✓ RS232
- ✓ USB

Los esquemas a usar serian los siguientes:

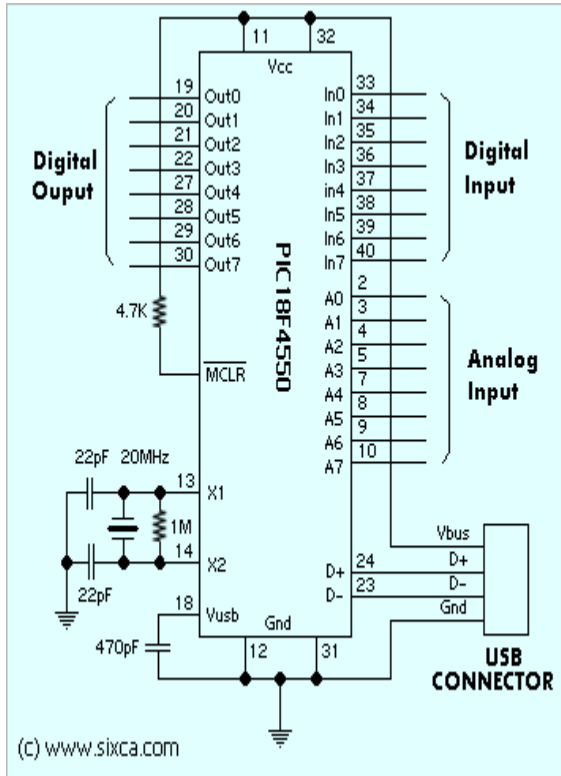


Gráfico 21: Distribución de Pines 18F4550

Fuente: Sixca

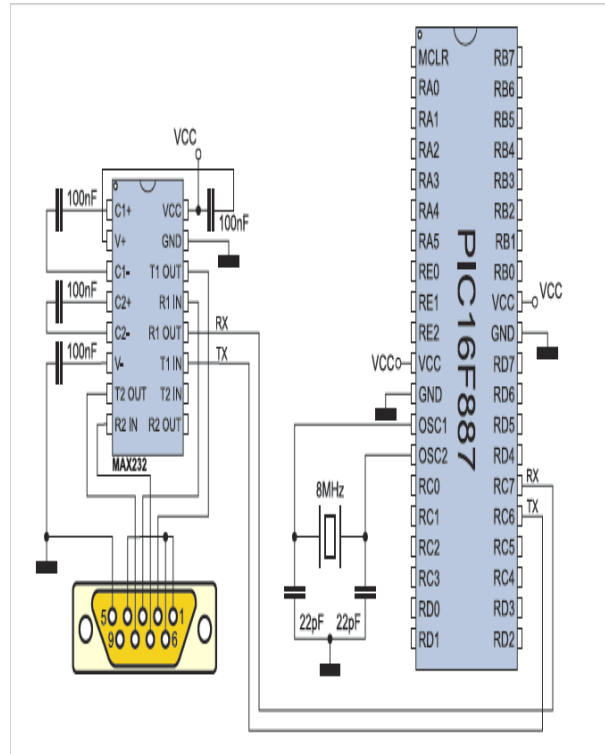


Gráfico 20: Distribución de Pines PIC 16F887

Fuente: Mikroe

La gráfica de la izquierda representa el esquema para conectarse por USB, el esquema de la derecha representa la solución con Rs232, la diferencia de las dos gráficas es la complejidad mientras que con Rs232 es sumamente sencillo, con Usb requiere más detalles para su uso. Sobre los costos con Usb resulta un poco más barato que con Rs232 pero no es una diferencia considerable.

Cualquiera de las dos soluciones se puede implementar pero la tecnología es distinta cambiando principalmente en su velocidad donde es superior la conexión por usb que llega hasta los 48Mbits/s de ancho de banda, mientras que Rs232 llega máximo hasta los 19.2Kbits/s.

## Simulación

Esquema:

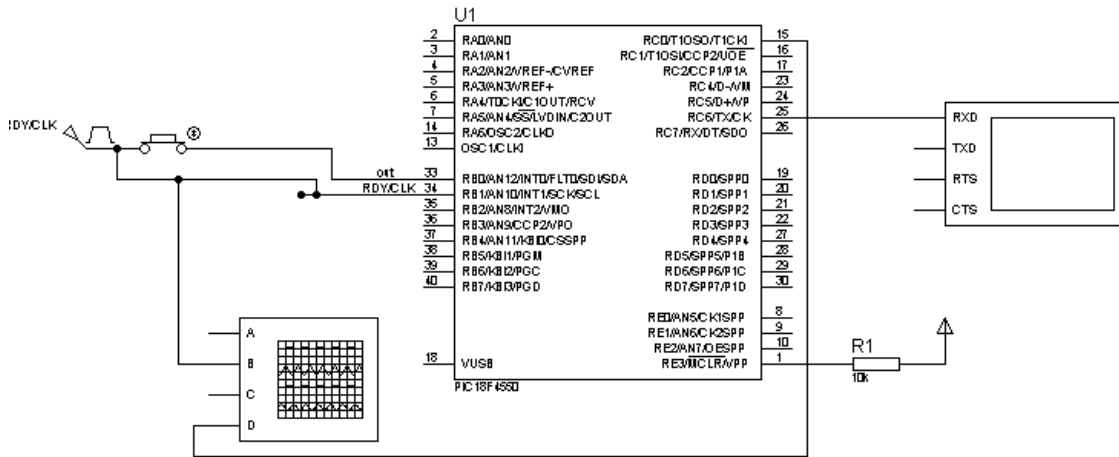
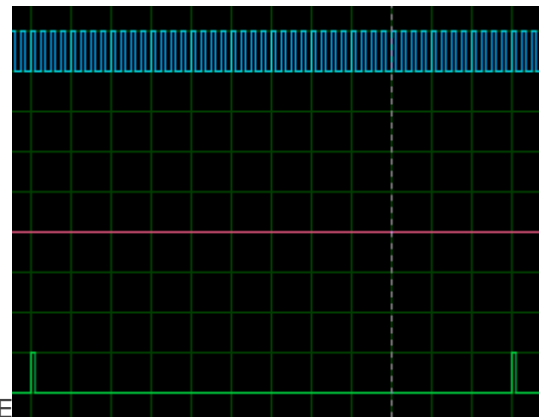


Gráfico 22 : Esquema para la simulación del circuito

Fuente: Proteus

Para simular el dispositivo se ha usado ISIS<sup>8</sup> de PROTEUS en donde se coloca una señal que genera un tren de pulsos, así mismo se ha puesto una botonera en la entrada de out que se encuentra en RB0 que sería por donde ingresan los datos de la tarjeta Rfid . El pin RDY/CLK es una señal de sincronización que está conectada a RB1

Para las pruebas se ha hecho que cuando se lea el flanco<sup>9</sup> de bajada de la señal eléctrica, que representa la interrupción en el pin Rb0, se procede a



<sup>8</sup> Intelligent Schematic Input System (Sistema de E

<sup>9</sup> La transición del nivel bajo al alto (flanco de subida) o del nivel alto al bajo (flanco de bajada) de una señal digital.

Gráfico 23: Simulación de un tren de pulsos del dispositivo RFID

Fuente: El Autor

guardar los bits de información de la tag pasiva, luego hemos hecho que se ponga en alto un bit del puerto C como se muestra en la imagen.

Se puede observar en la imagen como se ha simulado un tren de pulsos que actuaran como la señal que envía el dispositivo Rfid.

Para este caso lo que hemos hecho es aumentar la frecuencia de la señal de

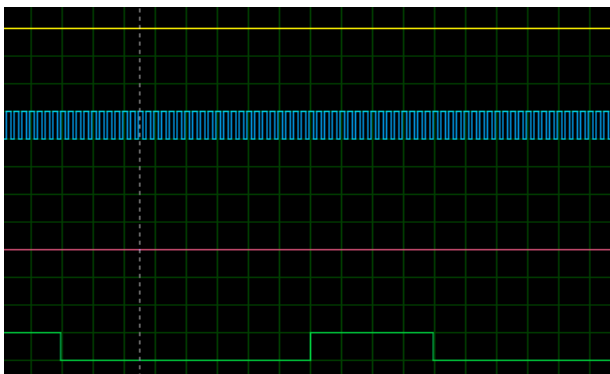


Gráfico 24: Simulación Verificando el Bit de Paridad

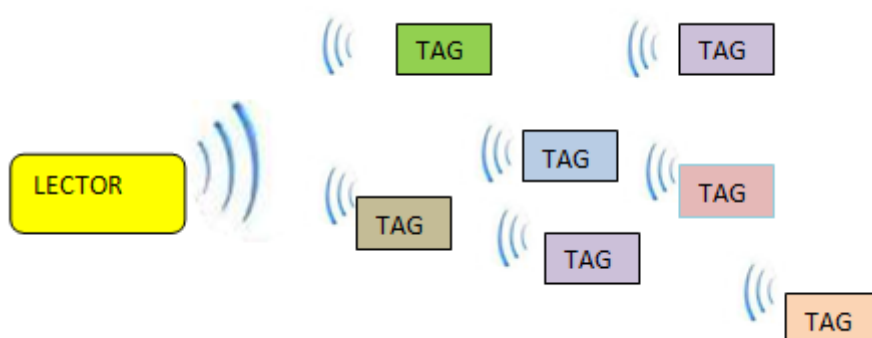
Fuente: El Autor

entrada y también apagar la señal del puerto C para saber que está leyendo los bits de paridad, que luego deberán ser verificados con algún algoritmo específico de paridad.

Las conclusiones que podemos sacar de éstas pruebas de simulación es que el microcontrolador sirve para ésta implementación y además ya se está leyendo los bits de información que en este caso han sido simulados con una señal de reloj.

### **Monitoreo Inalámbrico**

A partir de la tecnología Rfid Activa podemos monitorear los diferentes objetos haciendo un barrido de objeto por objeto para esto nos basamos en el siguiente esquema:



### Gráfico 25: Monitoreo Inalámbrico

Fuente: El Autor

Los objetivos tienen un dispositivo que sería una etiqueta o tag Activa así como también hay un lector Activo.

Sobre el esquema no hay mucho que decir, el lector siempre está preguntando por la presencia de los tags activos y estos responden, si acaso uno no responde el software que interactúa con el hardware sabrá que hacer.

#### ***Diseño de tags activos y lector***

A partir de nuestro esquema primario para el sistema de control de acceso podemos implementar adaptándole a este PIC un lector de dispositivos activos (que tienen su propia fuente de energía), esto con un módulo inalámbrico que haga un testeo constante y “encuentre” a los dispositivos que también diseñaríamos a partir de los siguientes elementos:

***RFM12:*** Transceptor inalámbrico de alto rendimiento, de fácil montaje . Ideal para enviar y recibir datos de forma inalámbrica desde una ubicación remota de hasta 300 metros al aire libre. Alta precisión basado en PLL, con excelentes velocidades de transmisión de datos. Interfaz SPI . Compatible con la mayoría de microcontroladores, PIC. Su bajo consumo de energía lo hacen ideal para el monitoreo de sensores y sistemas de seguridad y su costo en el mercado bordea los 8 dólares



Gráfico 26: Transceptor inalámbrico RFM12

Fuente: Shenzhen Shanhai

### Características:

- ✓ Frecuencia - 433MHz.
- ✓ PLL Based.
- ✓ Voltaje de operación: 2.2 - 5.4 Vdc.
- ✓ Baud rate por encima de 115.2K bps.
- ✓ Rango en campo abierto de hasta 300m.
- ✓ Registros de transmisión y recepción.
- ✓ Trabaja con la mayoría de microcontroladores PIC, SPI Interface.
- ✓ Dimensiones: Largo 16.1, Ancho 15.9mm, Alto 4.2mm.

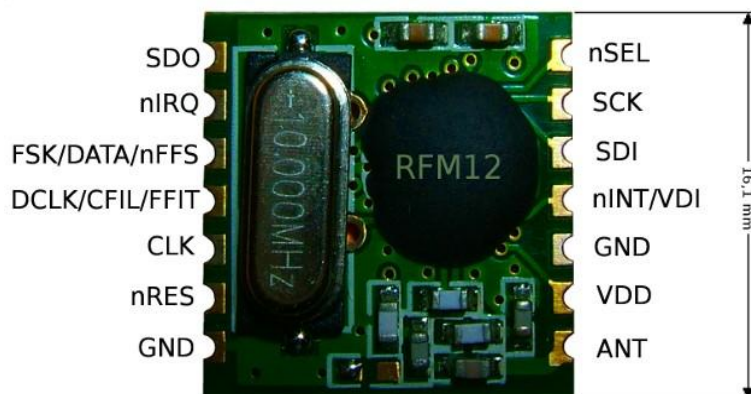


Gráfico 27: Configuración de Pines del Transceptor inalámbrico RFM12

Fuente: Picmania

### **PIC16LF1822:**

Este microcontrolador de bajo consumo de energía (serie L) con un tamaño muy reducido tiene la interfaz SPI<sup>10</sup>, necesaria para poder conectarse al módulo inalámbrico, además tiene un voltaje de operación reducido también llegando hasta operar con 1.8V, su costo en el mercado no supera los 3 dólares.

**Baterías de 3V:** Éstas baterías son usadas cuando el tamaño es un problema, debido a que nuestro tag activo debe ser energizado por una batería y considerando que debe ser un dispositivo lo más pequeño posible éstas baterías son las ideales.

Con estos tres dispositivos se puede construir un tag activo donde tan sólo se conecta el módulo RF y el pic de bajo consumo de energía programándolo para que responda cuando el lector así lo requiera.

**RFM12B:** Es una excelente opción de bajo costo para la comunicación inalámbrica, es una banda ISM<sup>11</sup> módulo de transceptor FSK<sup>12</sup> a cabo con un único PLL<sup>13</sup>. Estos módulos operan en la banda de 434MHz y cumplir plenamente con las normas FCC y ETSI.

Este módulo cuenta con una amplia gama de tensión de alimentación de 2,2 3.8VDC. Una interfaz SPI se utiliza para enviar los datos y configurar el módulo RFM12. Los comandos de configuración, que se describen en la hoja RF12 IC vinculada a continuación, se puede utilizar para fijar el tipo de datos, la banda de frecuencias, despertador, la transferencia de datos, recibir datos de la FIFO de 16-bit, y mucho más.

---

<sup>10</sup> Protocolo SPI ( Serial Peripheral Interface) Bus Serial de Interfaz

<sup>11</sup> ISM: "Industrial Scientific Medical" Bandas de radiofrecuencia electromagnética reservadas internacionalmente para uso no comercial en áreas de trabajo industriales, científicas y médicas.

<sup>12</sup> FSK: Frequency-shift keying Modulación por desplazamiento de frecuencia

<sup>13</sup> PLL: Phase-Locked Loop bucles de enganche de fase

El módulo viene en un paquete de SMD<sup>14</sup> de 14 pines, con alfileres espaciadas por 2 mm. Compruebe más abajo para una junta de arranque.

### **Características:**

- ✓ De bajo costo y de alto rendimiento.
- ✓ SPI compatible con la interfaz.
- ✓ Velocidad de datos de alta densidad (hasta 115,2 kbps en modo digital).
- ✓ Activación del temporizador.
- ✓ 2.2V-3.8V de alimentación.
- ✓ Salidas analógicas y digitales RSSI.
- ✓ Diferencial de entrada de antena.
- ✓ Ajuste automático de antena.
- ✓ 16-bits de datos FIFO RX.
- ✓ PLL y cero si la tecnología.
- ✓ Rápido tiempo de bloqueo de PLL.
- ✓ PLL de alta resolución con el paso 2.5KHz.
- ✓ Programable TX desviación de frecuencia (de 15 a 240 kHz).
- ✓ Ancho de banda del receptor programable (de 67 a 400 kHz).
- ✓ Señal analógica y digital indicador de potencia.
- ✓ Filtrado de datos interna y la recuperación de reloj.
- ✓ Reloj y la salida de señal de reinicio de uso externo MCU.
- ✓ Cristal de 10 MHz para la sincronización PLL.
- ✓ Dimensiones: 15,9 x 16,1 mm (0,626 x 0,634 in).

## **Guía de programación RF12B**

### **Breve descripción**

RF12B es un transceptor FSK de bajo costo IC bruja integrado todas las funciones de RF en un sólo chip. Sólo se necesita un MCU, un cristal, un condensador y una antena para desacoplar construir un sistema de FSK transceptor de alta fiabilidad. La frecuencia de operación puede cubrir 400 a 1000MHz.

RF12B es compatible con una interfaz de comandos para configurar la frecuencia, la desviación, la potencia de salida y también los datos de tasa. No

---

<sup>14</sup> SMD: *Surface Mount Device* Dispositivos de montaje superficial



es necesario ningún ajuste de hardware cuando se utiliza en el salto de frecuencia aplicaciones.

RF12B puede ser utilizado en aplicaciones tales como los juguetes de control remoto, alarma inalámbrica, los datos de sensores inalámbricos, teclado / mouse inalámbrico, la automatización del hogar e inalámbricas colección.

### **Aplicaciones típicas para el RF12B**

- ✓ Control a distancia.
- ✓ Seguridad y alarma de la casa.
- ✓ Teclado / ratón y otros periféricos de la PC inalámbricos.
- ✓ Controles de juguete.
- ✓ Entrada remota sin llave.
- ✓ Control de la presión de neumáticos.
- ✓ Telemetría.
- ✓ Personal / paciente registro de datos.
- ✓ Lectura remota automática de medidores.

### **1.5.2 Herramientas de Desarrollo**

#### **Software**

#### **PHP**

PHP (acrónimo de *PHP: Hypertext Preprocessor*) es un lenguaje de código abierto muy popular especialmente adecuado para desarrollo web y que puede ser incrustado en HTML.

Lo que distingue a PHP de algo lado-cliente como Javascript, es que el código es ejecutado en el servidor, generando HTML y enviándolo al cliente. El cliente recibirá los resultados de ejecutar el script, sin ninguna posibilidad de determinar que código ha producido el resultado recibido. El servidor web puede ser incluso

configurado para que procese todos los archivos HTML con PHP y entonces no hay manera que los usuarios puedan saber que tienes debajo de la manga.

Lo mejor de usar PHP es que es extremadamente simple para el principiante, pero a su vez, ofrece muchas características avanzadas para los programadores profesionales.

## **NetBeans 7**

Es un IDE una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. Está escrito en Java - pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el IDE NetBeans. El IDE NetBeans es un producto libre y gratuito sin restricciones de uso.

El NetBeans IDE es un IDE de código abierto escrito completamente en Java usando la plataforma NetBeans. El NetBeans IDE soporta el desarrollo de todos los tipos de aplicación Java (J2SE, web, EJB y aplicaciones móviles).

La versión del IDE de Netbeans que se utilizara será la versión 7.0.

## **Proteus**

Es una compilación de programas de diseño y simulación electrónica, desarrollado por LabcenterElectronics que consta de dos programas principales: Ares e Isis, y los módulos VSM y Electra.

ISIS.- Intelligent Schematic Input System (Sistema de Enrutado de Esquemas Inteligente), permite diseñar el plano eléctrico del circuito que se desea realizar con componentes muy variados, desde simples resistencias, hasta alguno que otro microprocesador o microcontrolador, incluyendo fuentes de alimentación, generadores de señales y muchos otros componentes con prestaciones diferentes. Los diseños realizados en Isis pueden ser simulados en tiempo real, mediante el módulo VSM, asociado directamente con ISIS.

## **Apache Web Service**

Servidor web de distribución libre y de código abierto, es el más popular del mundo desde abril de 1996, con una penetración actual del 50% del total de servidores web del mundo (agosto de 2007).

La principal competencia de Apache es el IIS (Microsoft Internet Information Services) de Microsoft.

Apache fue la primera alternativa viable para el servidor web de Netscape Communications, actualmente conocido como Sun Java System Web Server.

Apache es desarrollado y mantenido por una comunidad abierta de desarrolladores, bajo el auspicio de la Apache Software Foundation.

La aplicación permite ejecutarse en múltiples sistemas operativos como Windows, Novell NetWare, Mac OS X y los sistemas basados en Unix.

## **MY SQL**

Es un sistema de gestión de bases de datos relacional, multi-hilo y multi-usuario. Se ofrece bajo la GNU GPL para cualquier uso compatible con ésta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso.

## **Java**

Java fue diseñado como un lenguaje orientado a objetos. Los objetos se agrupan en estructuras encapsuladas tanto sus datos como los métodos que manipulan esos datos. La tendencia del futuro, a la que Java se suma, apunta

hacia la programación orientada a objetos, especialmente en entornos cada vez más complejos y basados en red.

Proporciona una colección de clases para su uso en aplicaciones de red, que permiten abrir sockets y establecer y aceptar conexiones con servidores o clientes remotos, facilitando así la creación de aplicaciones distribuidas.

Java es un lenguaje que es compilado e interpretado a la vez, el compilador transforma el código fuente en una especie de lenguaje máquina que luego es interpretado por la Java Virtual Machine (JVM).

## Hardware

### 1.5.3 Acceso

#### Lector RFid

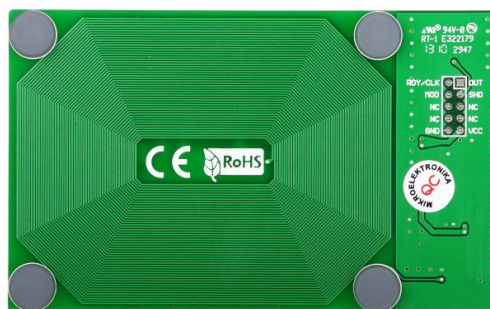


Gráfico 28: Lector RFid

Fuente: Mikroe.

*Especificaciones técnicas*

<i><b>Aplicaciones</b></i>	Junta se puede utilizar para leer el código de etiquetas RFID 125 KHZ o tarjetas de identificación.
<i><b>Características Principales</b></i>	<ul style="list-style-type: none"> <li>• EM4095 transmisor y receptor 125 Khz.</li> <li>• La señal está codificada utilizando el estándar de codificación de Manchester.</li> <li>• Algoritmos anticolidión permiten la lectura simultánea de un gran número de objetos etiquetados, garantizando al mismo tiempo que cada etiqueta se lee sólo una vez.</li> </ul>
<i><b>Principales Ventajas</b></i>	<ul style="list-style-type: none"> <li>• Diseño compacto adecuado para la integración de la junta en el dispositivo final.</li> <li>• Fácil conectividad con el conector IDC10 estándar.</li> <li>• Junta puede ser utilizado tanto como un lector y como un transmisor.</li> <li>• Ready-to-use ejemplos ahorran tiempo de desarrollo.</li> </ul>

**Tabla 2 Especificación Técnica Lector RFid**

**Fuente: Mikroe**

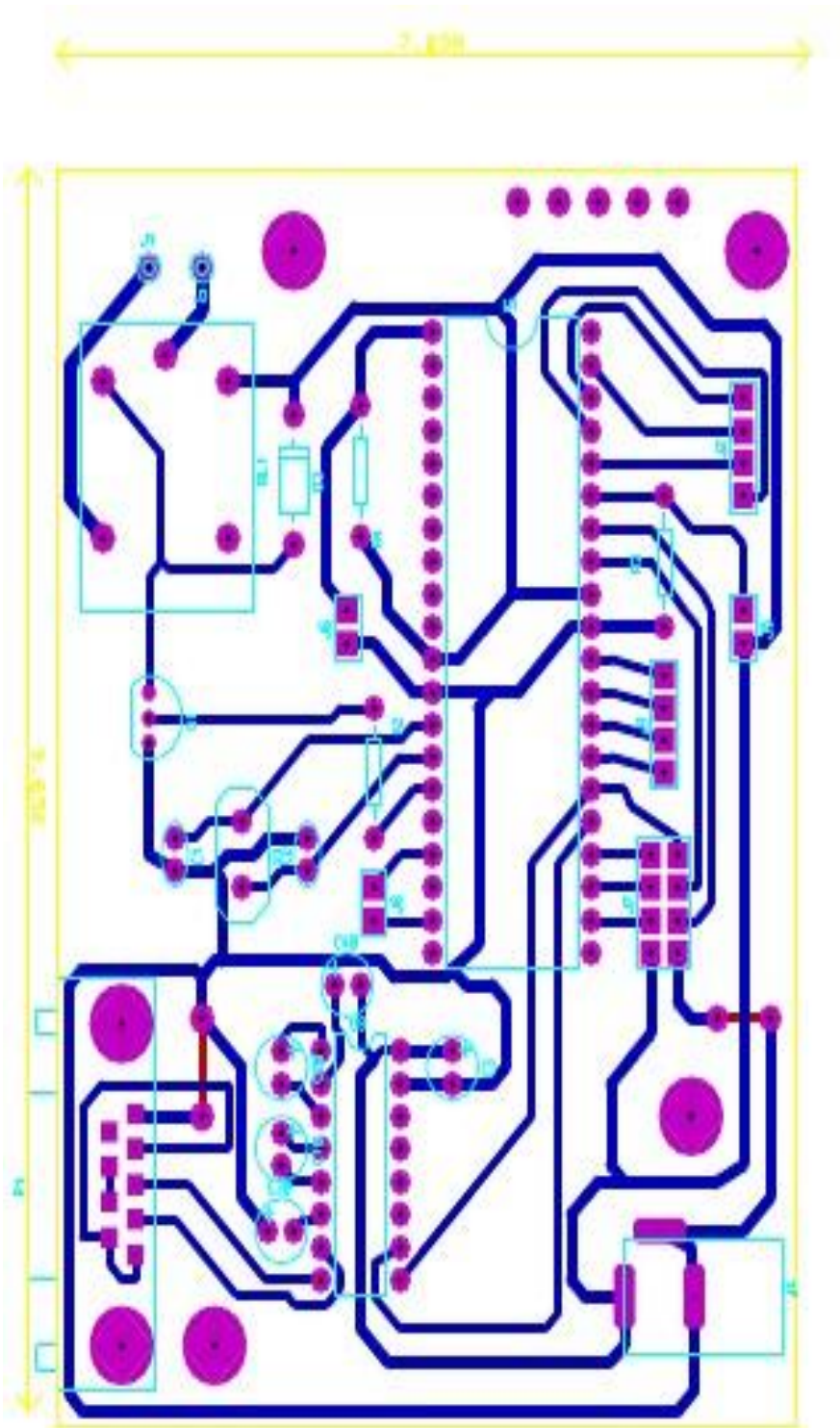
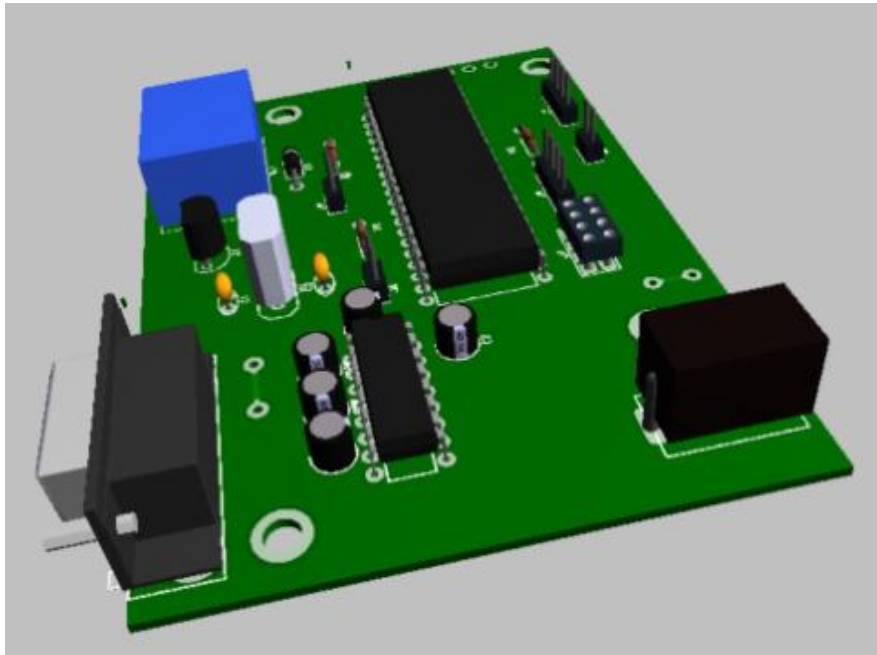


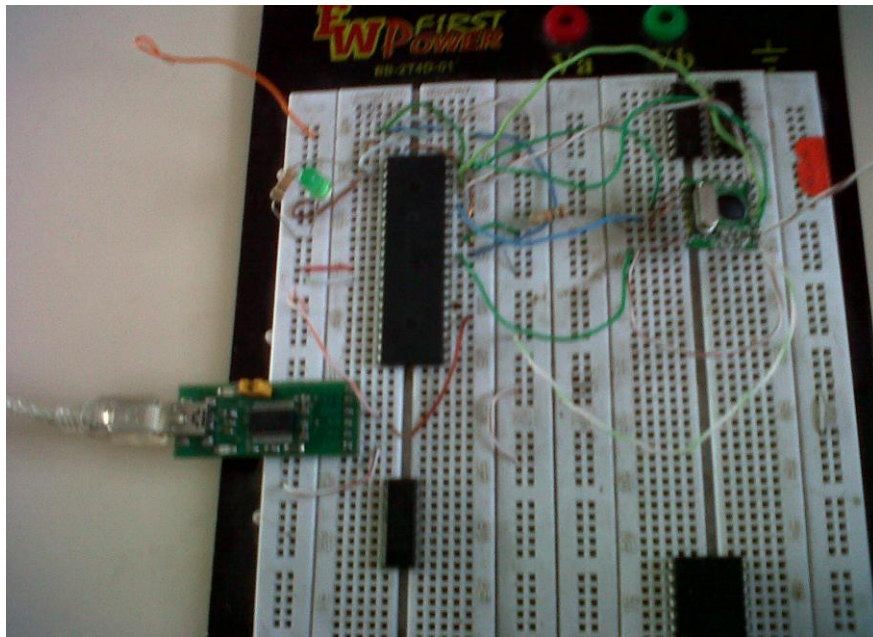
Gráfico 29: Gráfico del Circuito

Fuente: Proteus



**Gráfico 30: Placa del circuito del Lector**

**Fuente: Proteus**



**Gráfico 31: Circuito del Lector**

**Fuente: El Autor**

## PIC 18F4550



PIC18F4550-I/P (DIP40/20MHz)

Gráfico 32 PIC18F4550

Fuente: Mikroe

CARACTERISTICAS	PIC18F4450
Frecuencia de Operación	Hasta 48MHz
Memoria de Programa (bytes)	32.768
Memoria RAM de Datos (bytes)	2.048
Memoria EEPROM Datos (bytes)	256
Interrupciones	20
Líneas de E/S	35
Temporizadores	4
Módulos de Comparación/Captura/PWM (CCP)	1
Módulos de Comparación/Captura/PWM mejorado (ECCP)	1
Canales de Comunicación Serie	MSSP,EUSART
Canal USB	1
Puerto Paralelo de Transmisión de Datos (SPP)	1
Canales de Conversión A/D de 10 bits	13 Canales
Comparadores analógicos	2
Juego de instrucciones	75 (83 ext.)
Encapsulados	PDIP 40 pines QFN 40 pines TQFP 40 pines

Tabla 3 Especificaciones Técnicas PIC18F4550

Fuente: Documentos de Scrib



## PROGRAMACIÓN DEL PIC 18F4550

```
sbit OUT at RB0_bit;
sbit RDY_CLK at RB1_bit;
sbit SHD at RB4_bit;           //shd    b4
sbit MOD at RB3_bit;           //mod    b3

sbit DOOR at RC2_bit;           // c2
sbit DOOR_OPEN at RC0_bit;       // c0
sbit BUZZER_sound at RD0_bit;    // d0
sbit DOOR_OPEN_Direction at TRISC0_bit;
sbit DOOR_Direction at TRISC2_bit;
sbit OUT_Direction at TRISB0_bit;
sbit RDY_CLK_Direction at TRISB1_bit;
sbit SHD_Direction at TRISB4_bit;
sbit MOD_Direction at TRISB3_bit;

unsigned int time=0,time2=0;

unsigned short sync_flag,

one_seq,

data_in,

cnt,

cnt1, cnt2;

unsigned short data_index;

short b=0;

char i;

char _data[256];

char data_valid[64];

char bad_synch;

void Interrupt() {
```

```

if (INT1IF_bit && INT1IE_bit) {
cnt++;          // count interrupts on INT1 pin (RB1)
INT1IF_bit = 0;
}

// This is external INTO interrupt (for sync start)
// - once we get falling edge on RB0 we are disabling INTO interrupt
else if (INT0IF_bit && INT0IE_bit) {
cnt = 0;
sync_flag = 1;
INT0IF_bit = 0;
INT0IE_bit = 0;
INT1IF_bit = 0;
INT1IE_bit = 1;
}

else if (INT2IF_bit && INT2IE_bit) {
TRISC=0X80;
DOOR_OPEN=1;
BUZZER_sound=1;
Delay_ms(1000);
BUZZER_sound=0;
Delay_ms(1000);
DOOR_OPEN=0;
INT2IF_bit = 0;
}
}

char CRC_Check(char *bit_array) {
char row_count, row_bit, column_count;
char row_sum, column_sum;
char row_check[5];

```

```

char column_check[11];

row_count = 9;

while (row_count < 59) {

column_count = 0;

while (column_count < 5) {

row_check[column_count] = bit_array[row_count+column_count];

column_count++;

}

row_bit = 0;           // contar fila de bits

row_sum = 0;

while (row_bit < 4) {

row_sum = row_sum + row_check[row_bit];

row_bit++;

}

if (row_sum.B0 != row_check[4].B0) {

return 0;

}

row_count = row_count + 5;

}

// end row parity check

// column parity check

column_count = 9;     // contar columna

while (column_count < 13) {

row_bit = 0;         // contar columna de bits =)

row_count = 0;       // contar filas

while (row_bit < 11) {

column_check[row_bit] = bit_array[column_count+row_count];

row_bit++;

row_count = row_count + 5;

```

```

}
row_bit = 0;
column_sum = 0;
while (row_bit < 10) {
column_sum = column_sum + column_check[row_bit];
row_bit++;
}
if (column_sum.B0 != column_check[10].B0) {
return 0;
}
column_count++;
}
// end column parity check
if (bit_array[63] == 1) {
return 0;
}
return 1;
}
// main program
void main() {
ADCON1 = 0x0F;
CMCON = 7;
//OSCCON = 0b01110111; //8mhz
OUT_Direction = 1;
RDY_CLK_Direction = 1;
SHD_Direction = 0;
MOD_Direction = 0;
DOOR_Direction = 1;

```

```
DOOR_OPEN_Direction = 1 ;
SHD = 0;
MOD = 0;
UART1_Init(19200);
Delay_ms(100);
sync_flag = 0;
one_seq = 0;
data_in = 0;
data_index = 0;
cnt = 0;
cnt1 = 0;
cnt2 = 0;
// setup interrupts
INTEDG0_bit = 0;
INTEDG1_bit = 1;
INTEDG2_bit = 1;
INT0IF_bit = 0;
INT1IF_bit = 0;
INT2IF_bit = 0;
INT0IE_bit = 0;
INT1IE_bit = 0;
GIE_bit = 1;
while (1) {
b=0;
bad_synch = 0;
cnt = 0;
sync_flag = 0;
INT1IF_bit = 0;
```

```

INT1IE_bit = 0;
INT0IF_bit = 0;
INT0IE_bit = 1;

INT2IE_bit = 1;
while (sync_flag == 0) {           //lazu infinitus
asm nop
}
while (cnt != 16) {
asm nop
}
cnt = 0;
_data[0] = OUT & 1;
for (data_index = 1; data_index != 0; data_index++) {
while (cnt != 32) {
asm nop
}
cnt = 0;
_data[data_index] = OUT & 1;
if(data_index & 1)
if (!(_data[data_index] ^ _data[data_index-1]))
{
bad_synch = 1;
break;           //mala synchronisation
}
}
INT1IE_bit = 0;
if (bad_synch)

```

```

continue;

cnt1 = 0;

one_seq = 0;

for(cnt1 = 0; cnt1 <= 127; cnt1++) {
if (_data[cnt1 << 1] == 1) {
one_seq++;
}
else {
one_seq = 0;
}
if (one_seq == 9) {
break;
}
}

if ((one_seq == 9) && (cnt1 < 73)) {
data_valid[0] = 1;
data_valid[1] = 1;
data_valid[2] = 1;
data_valid[3] = 1;
data_valid[4] = 1;
data_valid[5] = 1;
data_valid[6] = 1;
data_valid[7] = 1;
data_valid[8] = 1;
for(cnt2 = 9; cnt2 <= 63; cnt2++) {
cnt1++;

data_valid[cnt2] = _data[cnt1 << 1];}
if (CRC_Check(data_valid) == 1) {

```

```

UART1_Write(0x20);
UART1_Write_Text("I:");
for (i = 0; i <= 64; i++){ //Envía el código
if (data_valid[i] == 0) { //
Uart1_Write('0'); //
} else { Uart1_Write('1'); //
} UART1_Write(13);
UART1_Write(10);
do {
if (UART1_Data_Ready()) {
if(UART1_Read()==0x4f)
{ if(UART1_Read()==0x4b)
{DOOR_OPEN=1;
BUZZER_sound=1;
Delay_ms(1000);
BUZZER_sound=0;
Delay_ms(500);
DOOR_OPEN=0;
}}} //listo?
time++;
}while(time!=10000);
time=0;
time2=0;
}}
}}

```

**Tag RFid Pasivas**





RFID Card 125KHz

Gráfico 33 Tag Rfid Pasivas

Fuente: Mikroe

<i>Especificaciones técnicas</i>	
Tarjeta de alta calidad	Frecuencia de la portadora 125
ID de 32 bits que es única para cada tarjeta.	Tarjeta blanca suave.
Suave y ligeramente flexible	Codificación de Manchester
64-bit de la secuencia de datos [Encabezado + ID + datos + paridad].	Dimensiones: 2,13 x 3,35 x 0,03 "(54 x 85,5 x 0,8 mm).

Tabla 4 Especificaciones Técnicas Rfid Pasivas

Fuente: Kineteka System.

**Cerradura electromagnética 150 kg. (300 libras) 12 / 24 volt.**



**Gráfico 34 Cerradura electromagnética**

**Fuente: JP Seguridad y Sistemas**

<i>Especificaciones técnicas</i>		
Cerradura Serie 300.	Electromagnetica	Fuerza de Retención 300 libras / 150 Kg.
Alimentación Soportada: 12V / DC 24V.	DC	Sensor de Supervisión NO & NC.
Corriente de operación: 380mA (DC 12V)/ 190mA (DC 24)		Aplicaciones: Compatible con balanceo interior o exterior
Temperatura de Operación: 35°C ± 5 ° C.		Peso: 1Kg.

**Tabla 5 : Especificaciones Técnicas Cerradura Electromagnética**

**Fuente: JP Seguridad y Sistemas**

## Cable Convertidor USB a RS-232



Gráfico 35: Cable convertidor USB a RS-232

Fuente: Electrónica Estudio

<i>Especificaciones técnicas</i>	
Soporta RS-232 serial interfaz.	Soporta 500 kbps velocidad de transferencia de datos.
Plug & Play compatible.	USB controladores de dispositivos de acogida disponibles
Compatible con Windows 98SE, ME, 2000, XP, Vista y Mac10.x 32/64 bits, Windows 7.	USB 1.1 & 2.0 compliant, Interfaz serial RS232 DB9 macho.
Ideal para proyectos con aplicaciones e interfaces entre PC y dispositivo que soporte serial C. Ideal para GPS, Modems y dispositivos seriales.	Soporta tasas de transferencia de datos por encima de los 1 Mbps.

Tabla 6 : Especificaciones Técnicas Cable convertidor USB a RS-232

Fuente: Electrónica Estudio

## 1.5.4 MONITOREO

## Esquema del dispositivo del Monitoreo

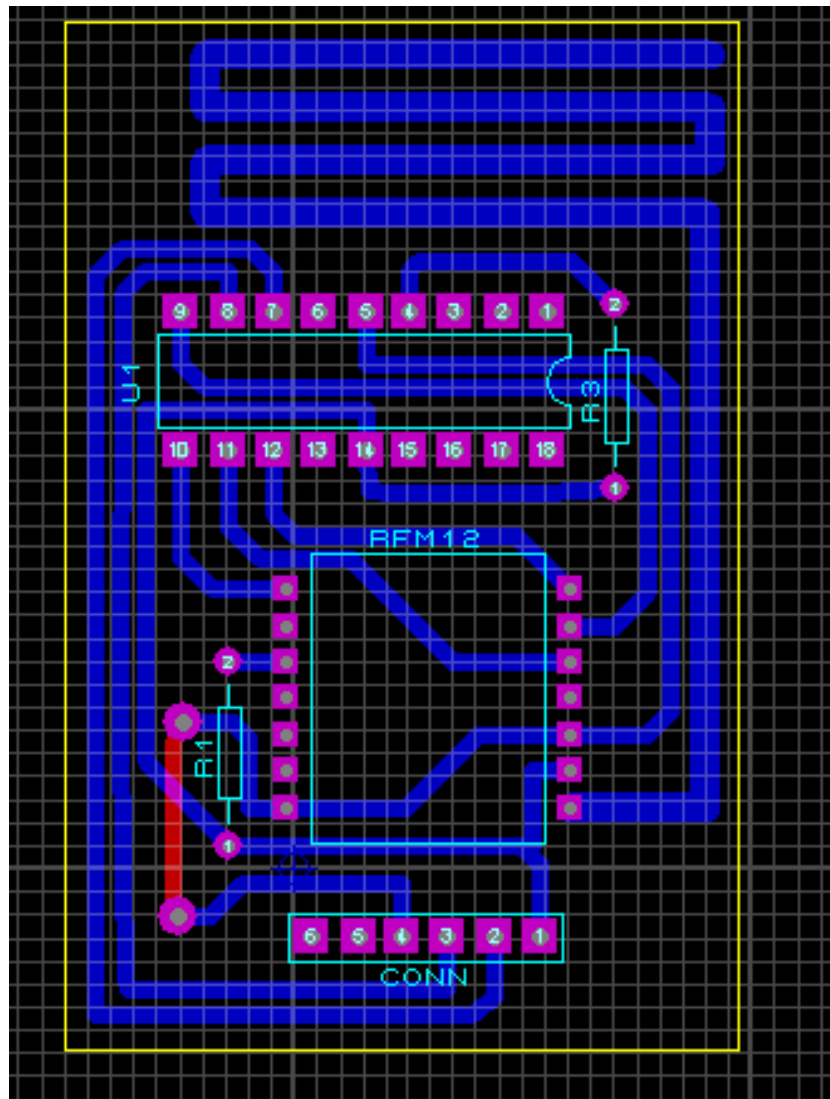


Gráfico 36: Esquema del Dispositivo del Monitoreo

Fuente: El Autor

## Esquema PCB de dispositivo

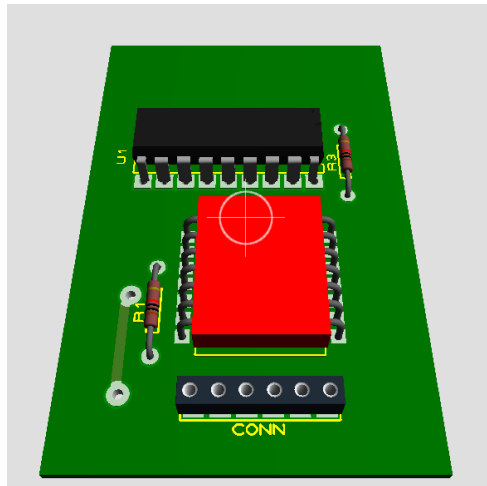


Gráfico 37: Esquema del Dispositivo del Monitoreo

Fuente: El Autor

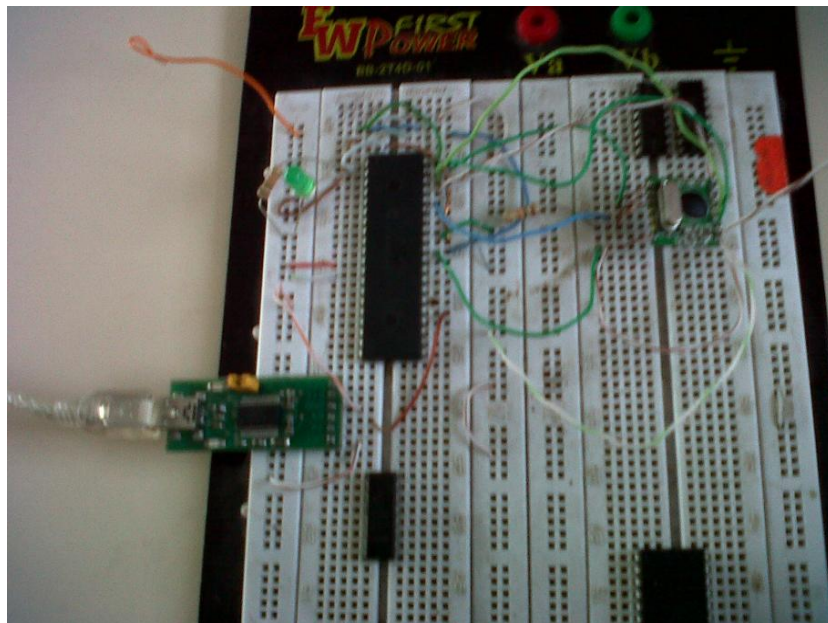


Gráfico 38: Placa del Monitoreo

Fuente: El Autor

## ***Código del Monitoreo***

```
typedef unsigned char uchar;

typedef unsigned int uint;

sbit SDI at RB7_bit;          // c2

sbit SDO at RB6_bit;          // c0

sbit SCK at RB5_bit;          // d0

sbit LED at RA0_bit;          // d0

sbit nSEL at RB4_bit;

sbit SDI_Direction at TRISB7_bit;

sbit SDO_Direction at TRISB6_bit;

sbit SCK_Direction at TRISB5_bit;

sbit nSEL_Direction at TRISB4_bit;

sbit LED_Direction at TRISA0_bit;

void Init_RF12(void);

void Write0( void );

void Write1( void );

void WriteCMD( uint CMD );

void DelayUs(uint us);

void DelayMs(uint ms);

void WriteFSKbyte( uchar DATA );

uint ChkSum;

void main()

{
    ADCON1 = 0x0F; // AD converter off
```

```

    CMCON = 7;

    OSCCON = 0b01110111;

UART1_Init(9600);      // Initialize UART module at 9600 bps

    PORTB=0;

    PORTC=0;

    Delay_ms(100);

    Init_RF12();

    Delay_ms(100);

    while(1)

    {

        ChkSum=0;

WriteCMD(0x8228); //|Power Management| ENABLE TRANSMITTER
"OPEN PA"

        DelayUs( 4);

        WriteCMD(0x8238); //|Power Management| Enable Synthesizer,
ENABLE TRANSMITTER

        NOP();

        NOP();

        WriteFSKbyte( 0xAA );

        WriteFSKbyte( 0xAA );

        WriteFSKbyte( 0xAA );

        WriteFSKbyte( 0x2D );

        WriteFSKbyte( 0xD4 ); // "PREAMBULE"

        WriteFSKbyte( 0x33 );//DATA0

```

```
ChkSum+=0x30;

Delayus(2);

WriteFSKbyte( 0x3a );//DATA1

ChkSum+=0x31;

WriteFSKbyte( 0x31 );

    ChkSum+=0x32;

    WriteFSKbyte( 0x31 );

    ChkSum+=0x33;

    WriteFSKbyte( 0x30 );

    ChkSum+=0x34;

    WriteFSKbyte( 0x31 );

    ChkSum+=0x35;

    WriteFSKbyte( 0x31 );

    ChkSum+=0x36;

    WriteFSKbyte( 0x30 );

    ChkSum+=0x37;

    WriteFSKbyte( 0x31 );

    ChkSum+=0x38;

    WriteFSKbyte( 0x30 );

    ChkSum+=0x39;

    WriteFSKbyte( 0x30 );

    ChkSum+=0x3A;
```



```
WriteFSKbyte( 0x30 );  
  
ChkSum+=0x3B;  
  
WriteFSKbyte( 0x30 );  
  
ChkSum+=0x3C;  
  
WriteFSKbyte( 0x31 );  
  
ChkSum+=0x3D;  
  
WriteFSKbyte( 0x30 );  
  
ChkSum+=0x3E;  
  
WriteFSKbyte( 0x31 );//DATA15  
  
ChkSum+=0x3F;  
  
ChkSum&=0x0FF;  
  
WriteFSKbyte( ChkSum );  
  
WriteFSKbyte( 0xAA );  
  
WriteFSKbyte( 0xAA );  
  
WriteCMD( 0x8208 ); //|Power Management| //CLOSE PA  
  
WriteCMD( 0x8200 ); //|Power Management|
```

DISABLE CRISTAL OSC Receive end, enter sleep

```
LED=1;
```

```
Delay_ms(100);
```

```
LED=0;
```

```
Delay_ms(950);
```

```
}
```

```
}
```

```

void Init_RF12(void)
{
SDI_Direction = 0 ;

SDO_Direction = 1 ;

SCK_Direction = 0 ;

nSEL_Direction= 0 ;

LED_Direction = 0 ;

LED=0;

    nSEL=1;

    SDI=1;

    SCK=0;

    WriteCMD(0x80D8);///Configuration Settings| enable
register,433MHz,12.5pF,TX & FIFO RX

    Uart1_Write(0x01);

    WriteCMD(0x8208);///Power Management| Turn on crystal,!PA

    Delay_ms(2000);

    WriteCMD(0xA640);///Frequency Setting| Center Frequency =
434.0000 MHz, Fc = 430 + F x 0.0025 MHz; F=1600

    Delay_ms(2000);

    WriteCMD(0xC647);///Data Rate| Data Rate = 4.789 kbps;R=71

    Delay_ms(2000);

    WriteCMD(0XCC77);///PLL Settings???

    Delay_ms(2000);

    WriteCMD(0x94A0);///Receiver Control|

```

VDI,FAST,134kHz,LNA GAIN MAX 0dBm,DRSSI = -103dBm

Delay\_ms(2000);

WriteCMD(0xC2AC);//|Data Filter & Clock Recovery| Filter Type  
DIGITAL, Quality Threshold 4, Recovery Mode AUTO

Recovery Speed SLOW

WriteCMD(0xCA80);// FIFO and Reset Mode|

WriteCMD(0xCA83);//|FIFO and Reset Mode|

FIFO INT Level 8, FIFO Fill Start Sync, FIFO Fill Enabled ON, Reset  
Sensitivity LOW

WriteCMD(0xC49B);//|Automatic Frequency Control|

WriteCMD(0x9850);//|TX Control|

Frequency Shift +,Deviation 90Khz, Power Out 0dB MAX OUT  
"!Imp,9810=30kHz"

WriteCMD(0xE000);//|Wake-Up Timer| NOT USE

WriteCMD(0xC80E);//|Low Duty-Cycle| D=7 NOT USE

WriteCMD(0xC000);//|Low Battery

Detect and  $\mu$ C Clock| 1.0MHz,2.2V

//Synchronization Pattern?? 0xCEXX

```
Uart1_Write_Text("Comands Succesful!!");  
} void Write0( void )  
{  
SCK=0;  
NOP();  
SDI=0;
```

```
NOP();NOP();NOP();NOP();  
NOP();NOP();NOP();NOP();  
NOP();NOP();NOP();NOP();  
NOP();NOP();NOP();SCK=1;  
NOP();  
}
```

```
void Write1( void )
```

```
{  
    SCK=0;  
    NOP();  SDI=1;  
    NOP();  NOP();  
    NOP();  NOP();  
    NOP();  NOP();  
    NOP();  NOP();  
    NOP();  NOP();
```

```
    NOP();  NOP();  
    NOP();  NOP();  
    NOP();  SCK=1;  
    NOP(); }
```

```
void WriteCMD( uint CMD )
```

```
{  
    uchar n=16;
```

```

    SCK=0;

    nSEL=0;

while(n--)
{   if(CMD&0x8000)

    {Write1();

    }   else

{Write0();

    }   CMD=CMD<<1;

    }   SCK=0;

    nSEL=1;

} void WriteFSKbyte( uchar DATA )

{   uchar RGIT=0;

    uint temp=0xB800; // B8+byte a transmitit

    temp|=DATA;

    Loop:

    SCK=0;

    nSEL=0;

    SDI=0;

    SCK=1;

    if(SDO) //Polling SDO

    {   RGIT=1;   }   else

    {   RGIT=0;   }   SCK=0;

```

```

SDI=1;

nSEL=1;

    if(RGIT==0)

        {   goto Loop;

        }   else

        {   RGIT=0;

            WriteCMD(temp);

        }} void DelayUs(uint us){

uint i;

    while(us--)

        { i=2;

            while(i--)

                { NOP();

                } }}

```

## Módulo RFM12



Gráfico 39: Módulo RFM12

Fuente: Clifford Metaparts

<i>Especificaciones técnicas</i>	
Ajuste automático de antena.	Ancho de banda del receptor programable (67 a 400 kHz).
Bajo costo, alto rendimiento y precio.	RX reconocimiento de patrones sincrónicos.
Los datos internos de filtración y de recuperación de reloj.	Alta velocidad de datos (hasta 115,2 kbps con demodulador interno, con filtro de velocidad de datos más alta RC).
Tiempo de bloqueo PLL Fast.	Desviación de frecuencia TX programable (15-240 KHz).
Dos 8 bits TX registros de datos.	Temporizador despertador.
Bajo consumo de energía.	Interfaz SPI.
Corriente espera menos de 0.3Ua.	Reloj y salida de señal de reset para uso externo MCU.
Entrada de antena de diferencial	PLL de alta resolución con 2,5 paso KHz.
2.2V - 5.4V fuente de alimentación.	Cristal 10MHz para la sincronización PLL.

Tabla 7 Especificaciones Técnicas Módulo RFM12

Fuente: Clifford Metaparts

## PIC16F628A

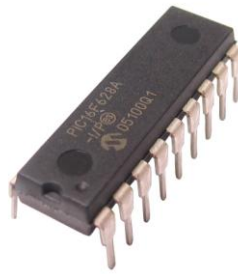


Gráfico 40: PIC16F628A

Fuente: Aula Pic

ESPECIFICACIONES TÉCNICAS	
Memoria de programa	1 KBytes)
Memoria <b>SRAM</b>	224 Bytes
Memoria <b>EEPROM</b>	128 Bytes
Pines de <b>E/S</b>	16
Entradas analógicas ( <b>ADC</b> )	No
Salidas <b>PWM</b>	2
<b>SPI</b>	No
<b>I2C</b>	No
<b>USART</b>	Si
<b>Temporizadores de 8Bits</b>	2
<b>Temporizadores de 16Bits</b>	1
<b>Comparadores</b>	2
<b>Oscilador</b>	Frecuencia máxima: 20MHz Oscilador interno de 4MHz.
Número de pines	18
<b>Encapsulado</b>	<b>PDIP, SOIC, SSOP, QFN</b>

Tabla 8: Especificaciones PIC16F628A

Fuente: Aula Pic



## Módulo UART USB

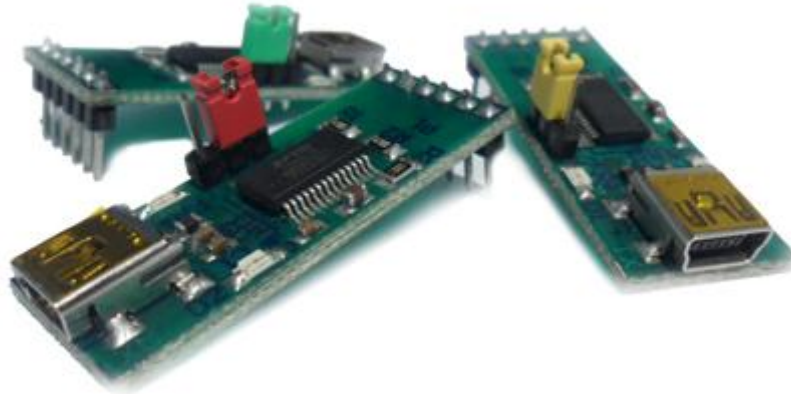


Gráfico 41: Módulo UART USB

Fuente: Ideas &Tecnologías

Es un módulo de comunicación serial por puerto USB para hacer interface con un computador o PC.

## Resistencia de 10 K

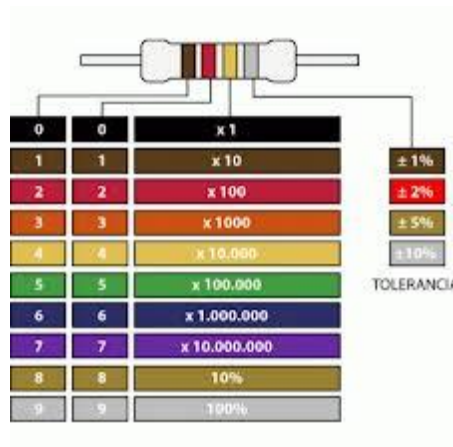


Gráfico 42: Resistencia de 10 K

Fuente: Sual Labs

Descripción: Resistencia de 1/4 Watt, +/- 5% de tolerancia.

Características:

- Resistencia de carbón 250mW.
- Tolerancia :± 5%.
- Máxima potencia nominal: 250mW.
- Voltaje nominal: 250V.
- Coeficiente de temperatura:± 450ppm/°C.
- RoHS.

### 1.5.5 Marco conceptual.

**Clase:** Es una construcción que se utiliza como un modelo (o plantilla) para crear objetos de ese tipo. El modelo describe el estado y el comportamiento que todos los objetos de la clase comparten.

**MVC:** Modelo Vista Controlador es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

**Query:** Es una consulta o instrucción a la bases de datos, la cual puede retornar o no datos almacenados en ella o modificar los ya almacenados.

**Servidor:** Es una computadora que, formando parte de una red, provee servicios a otras computadoras denominadas clientes.

**Trigger:** Es una instrucción almacenada en la base de datos que puede ejecutarse cuando se realiza una acción sobre la base de datos.

**DOM:** Es esencialmente una interfaz de programación de aplicaciones que proporciona un conjunto estándar de objetos para representar documentos HTML y XML, un modelo estándar sobre cómo pueden combinarse dichos objetos, y una interfaz estándar para acceder a ellos y manipularlos.

**API:** Una interfaz de programación de aplicaciones o API (del inglés Application Programming Interface) es el conjunto de funciones y procedimientos que ofrece

cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

**SQL:** El lenguaje de consulta estructurado o SQL es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones en éstas.

**PIC:** Son una familia de microcontroladores tipo RISC fabricados por Microchip Technology Inc. y derivados del PIC1650, originalmente desarrollado por la división de microelectrónica de General Instrument.

**Ethernet:** Es un estándar de redes de área local para computadores con acceso al medio por contienda CSMA/CD. ("Acceso Múltiple por Detección de Portadora con Detección de Colisiones"), es una técnica usada en redes Ethernet para mejorar sus prestaciones.

**Base de Datos:** Una base de datos o banco de datos (en ocasiones abreviada con la sigla BD o con la abreviatura b. d.) es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.

**Encapsulamiento:** En programación modular, y más específicamente en programación orientada a objetos, se denomina encapsulamiento al ocultamiento del estado, es decir, de los datos miembro, de un objeto de manera que sólo se puede cambiar mediante las operaciones definidas para ese objeto.

**Sistema de Información:** Es un conjunto de elementos orientados al tratamiento y administración de datos e información, organizados y listos para su posterior uso, generados para cubrir una necesidad (objetivo).

## 1.6 Formulación de la hipótesis y variables.

### 1.6.1 Hipótesis general.

La Universidad Politécnica Salesiana no cuenta con un sistema de seguridad ni de control avanzado para los Laboratorios y sus equipos. Lo cual conlleva una

pérdida de tiempo y de recursos temporales y materiales en los laboratorios. Esto se podría mejorar con la implementación de un sistema automatizado de control de acceso a los Laboratorios y de los equipos.

### **1.6.2 Hipótesis Particulares.**

- ❖ Con la implementación de un Sistema de control de acceso a los laboratorios de Telemática se podrá optimizar el acceso del personal autorizado.
  
- ❖ Con el sistema de monitoreo continuo de los equipos de laboratorio, se eliminarán las pérdidas de todos los suministros, reduciendo costos de mantenimiento y reposición de equipos.

### **1.6.3 MATRIZ CAUSA – EFECTO.**

<b>FORMULACIÓN DEL PROBLEMA</b>	<b>OBJETIVO GENERAL</b>	<b>HIPÓTESIS GENERAL</b>
---------------------------------	-------------------------	--------------------------

¿Cuáles son las causas por los que los docentes al buscar el encargado de abrir los laboratorios pierden tiempo y a su vez incrementa la inseguridad de los equipos de trabajo?	Identificar las causas por los que los docentes al buscar el encargado de abrir los laboratorios pierden tiempo y a su vez incrementa la inseguridad de los equipos de trabajo.	La Universidad Politécnica Salesiana no cuenta con un sistema de seguridad ni de control avanzado para los Laboratorios y sus equipos. Lo cual conlleva una pérdida de tiempo y de recursos temporales y materiales en los laboratorios. Esto se podría mejorar con la implementación de un sistema automatizado de control de acceso a los Laboratorios y de los equipos.
<b>SISTEMATIZACIÓN DEL PROBLEMA</b>	<b>OBJETIVOS ESPECÍFICOS</b>	<b>HIPÓTESIS ESPECÍFICAS</b>
¿Cuál es el proceso que se debe de tomar para que los docentes no pierdan tiempo?	Conocer el proceso que se debe de aplicar para que los docentes no pierdan tiempo	Con la implementación de un Sistema de control de acceso a los laboratorios de Telemática se podrá optimizar el acceso del personal autorizado.
¿Qué ocasiona que los docentes pierden el tiempo en sus actividades y no aprovechen sus horas de clases completas?	Identificar que ocasiona que los docentes pierdan tiempo en sus actividades y no aprovechen sus horas de clases completas	Con el sistema de monitoreo continuo de los equipos de laboratorio, se eliminarán las pérdidas de todos los suministros, reduciendo costos de mantenimiento y reposición de equipos.
¿Qué incidencia tiene la inseguridad en los laboratorios si estos quedan abiertos durante horas?	Determinar la incidencia que tiene la inseguridad en los laboratorios si estos quedan abiertos durante horas	
¿Qué consecuencia tiene la inseguridad en los laboratorios?	Determinar las consecuencias que tiene la inseguridad en los laboratorios	

**Tabla 9 Matriz Causa - Efecto**

**Fuente: El Autor**

#### **1.6.4 Variables.**

#### **1.6.4.1 Variables independientes.**

Nivel de acceso, seguridad y control de los activos del laboratorio de telemática.

Sistema de control de acceso y monitoreo

#### **1.6.4.2 Variables dependientes.**

- Falta de tiempo
- Falta de recursos
- Seguridad y acceso oportuno

### **1.7 Aspectos metodológicos de la investigación.**

#### **1.7.1 Tipo de estudio.**

- ❖ **Investigación Exploratoria:** Se realizara investigaciones y así poder resolver los problemas existentes mencionados mediante las hipótesis que se plantearon en la sección anterior.
- ❖ **Investigación de Campo:** Se aplica este tipo de investigación porque mediante ella podemos informarnos la opinión de la problemática mediante entrevistas o cuestionarios a los docentes con el fin de dar seguridad a los laboratorios.
- ❖ **Investigación Aplicada:** Mediante ella podemos poner en práctica los modelos de la tecnología de identificación por radiofrecuencia es decir toda la tecnología que se aprendió durante los años de estudio.
- ❖ **Investigación Pura:** Se aplica cuando queremos cumplir el objetivo de poner en práctica los conocimientos teóricos de las leyes y las teorías. ocuparse de su aplicación práctica. Es una parte eminentemente teórica de todas las ciencias.

## **1.7.2 Método de investigación.**

### **1.7.2.1 Método Observativo.**

Mediante este método pudimos fijarnos la problemática que vive la universidad puesto que la seguridad de los equipos es cada día necesaria para las tareas diarias, ya que mediante ésta técnica se tendría un correcto control de ellos y así ayudamos a dar menos responsabilidad a los delegados.

### **1.7.2.2 Método Analítico.**

Se efectuó la comprobación de todo el déficit de seguridad de los laboratorios de la Universidad en especial el mejor equipado y más completo de la Carrera de Ingeniería de Sistemas que es el Laboratorio de Telemática y de ésta forma establecer el sistema eficaz y sólido.

### **1.7.2.3 Método Experimental.**

Por el método experimental vamos a comprobar o demostrar las hipótesis planteadas ya que no sólo con mirar lo sucedido resolverá el dilema del control de acceso.

## **1.7.3 Fuentes y técnicas para la recolección de información.**

Para la recolección de los datos se tiene una gran diversidad de métodos como los son las encuestas, entrevistas, cuestionarios, etc. que van a dar la nota final al trabajo de implementación y nos dará la certeza de la meta fijada.

### ***Encuesta***

Según Stanton, Etzel y Walker, una encuesta consiste en reunir datos entrevistando a la gente<sup>15</sup>.

Para Richard L. Sandhusen, las encuestas obtienen información sistemáticamente de los encuestados a través de preguntas, ya sea personales, telefónicas o por correo.<sup>16</sup>

En síntesis, y teniendo en cuenta las anteriores definiciones, se planteo la siguiente definición de encuesta:

La encuesta es un instrumento de la investigación de mercados que consiste en obtener información de las personas encuestadas mediante el uso de cuestionarios diseñados en forma previa para la obtención de información específica.

Complementando lo anterior, cabe señalar que el Diccionario de Marketing de Cultural S.A. define el término encuestación como el método de recogida de información cuantitativa que consiste en interrogar a los miembros de una muestra, sobre la base de un cuestionario perfectamente estructurado.<sup>17</sup>

### **1.8 Resultados e impactos esperados.**

Lo que se espera con este proyecto es dar mayor resguardo a los equipos y más agilidad al acceso por parte de los docentes a los laboratorios, generando mayor puntualidad a las clases, mejor desempeño en el uso de los laboratorios, llevar un registro de control en caso de alguna irregularidad en cualquier momento.

## **CAPITULO 2**

---

<sup>15</sup> Del libro: Fundamentos de Marketing, 13a. Edición, de Stanton, Etzel y Walker, Mc Graw Hill, 2004, Pág. 212.

<sup>16</sup> Del libro: Mercadotecnia, Primera Edición, de Sandhusen L. Richard, Compañía Editorial Continental, 2002, Pág. 229.

<sup>17</sup> Del libro: Diccionario de Marketing, de Cultural S.A., Edición 1999, Pág. 113.



## **2. Análisis presentación de resultados y diagnóstico.**

### **2.1 Análisis de la Situación Actual**

#### **Universidad Politécnica Salesiana**

La Universidad Politécnica Salesiana es una universidad ecuatoriana perteneciente a la Congregación Salesiana. Fue fundada en 1994 en la ciudad de Cuenca y tiene sedes en Guayaquil y Quito.

Se Caracteriza por educar a sus estudiantes con excelencia humana y académica.

#### **Reseña Histórica**

La presencia salesiana en el Ecuador es una realidad social desde enero de 1888, como respuesta al convenio firmado por Don Bosco y el representante del Gobierno del Ecuador en Turín (Italia) en 1887, por el que se confía a los salesianos el Protectorado Católico de Artes y Oficios de Quito, para que impartan educación moral y científica a los hijos del pueblo y para el desarrollo de la industria nacional mediante una enseñanza sistemática de la artesanía".

Muy pronto, la obra evangélica-educativa de los salesianos se extendió a otras ciudades del Ecuador, destacándose la fundación de las Misiones en el Oriente Ecuatoriano como Gualaquiza (1893), Indanza (1914), Méndez (1915), Macas (1924), Sucúa (1931) y Limón (1936). En lo educativo también se fundan obras como las de Quito (1888) con los talleres de artes y oficios en el Protectorado Católico; en Riobamba (1881) se funda la escuela primaria, talleres y el oratorio festivo; en Cuenca (1893) empiezan los talleres y el oratorio festivo.

En Quito, en el barrio La Tola (1896) se abren los talleres de mecánica y carpintería, la escuela primaria y la Iglesia dedicada a María Auxiliadora; Guayaquil (1904) vio la primera fundación con el Instituto Domingo Santistevan para niños huérfanos con el patrocinio de la Junta de Beneficencia. En el Barrio Centenario de ésta misma ciudad se fundó el Colegio Cristóbal Colón (1911) para la educación humanística de la juventud guayaquileña; en Manabí (1927) los salesianos reciben la Parroquia Rocafuerte, en la que se abre igualmente una escuela primaria y un oratorio festivo.

Desde 1888 las obras educativas y apostólicas se han ido multiplicando por el Ecuador, insertándose en los diversos grupos sociales con el fin de responder a las necesidades de los jóvenes, especialmente de los más pobres a través de una educación de calidad basada en el Sistema Preventivo e inspirada en los valores del Evangelio, con el fin de formar "honrados ciudadanos y buenos cristianos".

Hoy, los Salesianos de Ecuador son alrededor de 200 hermanos, distribuidos en 27 comunidades en Costa, Sierra y Amazonía.<sup>18</sup>

### **Misión**

La formación de honrados ciudadanos y buenos cristianos, con excelencia humana y académica. El desafío de nuestra propuesta educativa liberadora es formar actores sociales y políticos con una visión crítica de la realidad, socialmente responsables, con voluntad transformadora y dirigida de manera preferencial a los pobres.<sup>19</sup>

### **Visión**

La Universidad Politécnica Salesiana, inspirada en la fe cristiana, aspira constituirse en una institución educativa de referencia en la búsqueda de la verdad, el desarrollo de la cultura, de la ciencia y tecnología, mediante la aplicación de un estilo educativo centrado en el aprendizaje, docencia, investigación y vinculación con la colectividad, por lo que se compromete, decididamente, en la construcción de una sociedad democrática, justa, equitativa, solidaria, con responsabilidad ambiental, participativa y de paz.<sup>20</sup>

## **Estructura Organizacional**

---

<sup>18</sup> Fuente: <http://www.ups.edu.ec/resena-historica>

<sup>19</sup> Fuente: <http://www.ups.edu.ec/razon-de-ser>

<sup>20</sup> Fuente: <http://www.ups.edu.ec/razon-de-ser>



Gráfico 43: Organigrama de la Universidad Politécnica Salesiana

Fuente: <http://www.ups.edu.ec/organigrama>

## 2.2 Herramientas de análisis.

La recolección de datos es un proceso muy importante para la investigación ya que permite la recolección de elementos para obtener datos de la realidad, para esto se crean instrumentos que debe contemplar en su construcción la forma (técnica para cumplir con la tarea) y contenido (especificación de datos que necesitamos conseguir). El instrumento permite sintetizar toda la labor previa de la investigación y por esto la importancia que tiene para poder obtener datos verdaderos.

### 2.2.1. Encuestas

Se realizó una encuesta a 10 docentes que laboraban en el laboratorio de telemática para recabar información en forma verbal, a través de preguntas para conocer la incidencia de los hurtos en los laboratorios y la pérdida de tiempo en

el acceso a los laboratorios. Se los entrevisto de forma individual y se captó el parecer de todos los docentes sobre el prototipo que se construyó.

### 2.2.1.2 Análisis Estadístico de las encuestas

***Cree usted ¿qué existe inseguridad dentro de los laboratorios del Campus?***

Si	2	66%
No	1	33%

Tabla 10 Tabulación 1

Fuente: El Autor

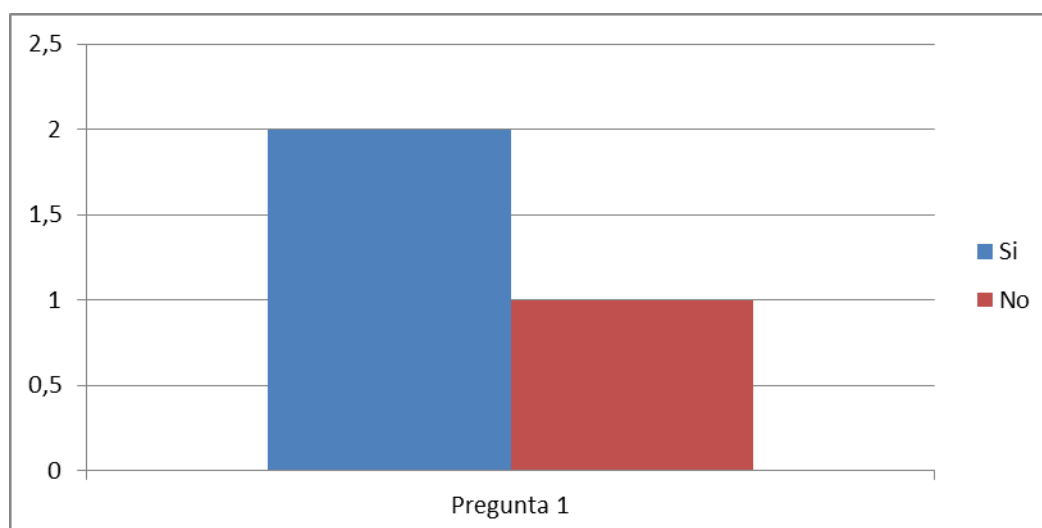


Gráfico 44: Diagrama de Barras pregunta 1

Fuente: El Autor

La mayoría de los profesores que utilizan el laboratorio manifiestan que existe inseguridad en los laboratorios del campus, esto crea la necesidad de tomar medidas en torno a la seguridad de los mismos, tanto en el acceso como en el uso de todos los laboratorios.

***Cree usted ¿qué pierde mucho tiempo en el acceso a los distintos laboratorios de la Universidad al momento de impartir su clase?***

Si	10	100%
----	----	------

No	0	0%
----	---	----

Tabla 11: Tabulación 2

Fuente: El Autor

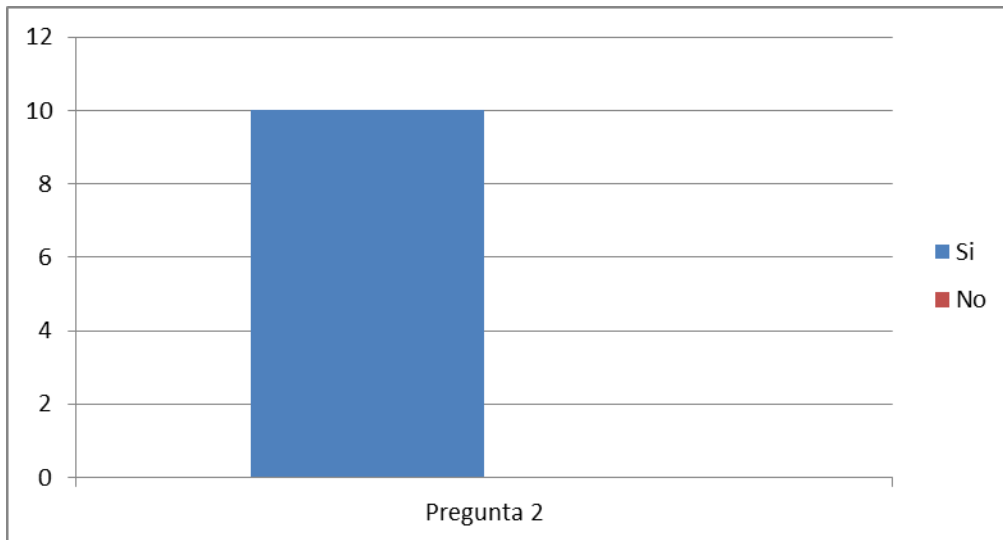


Gráfico 45 Diagrama de Barras pregunta 2

Fuente: El Autor

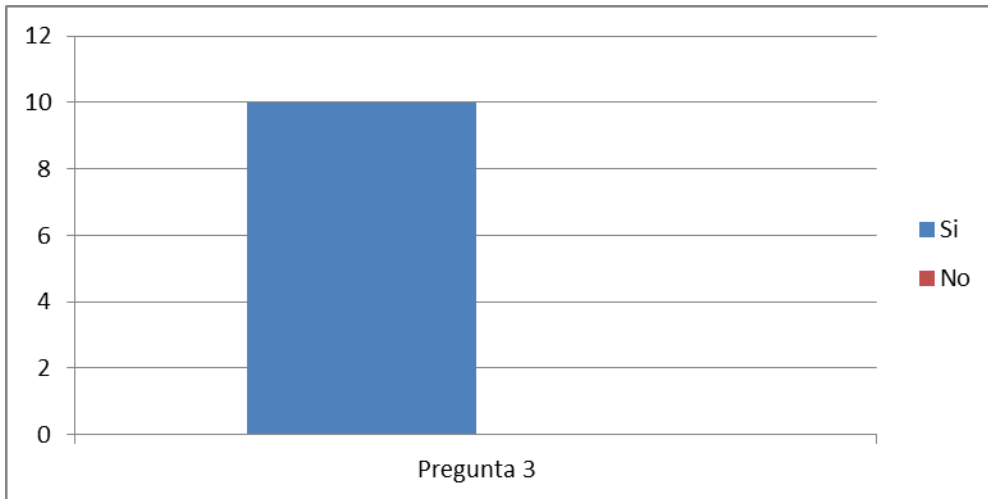
El 100% de los docentes que utilizan los laboratorios coinciden en que se pierde demasiado tiempo en el acceso, el encargado de abrir los laboratorios en ciertas ocasiones no está en su piso o se encuentra realizando otras actividades y este problema afecta al dictar la cátedra.

***Cree usted ¿qué es necesario la implementación del sistema de control de acceso y monitoreo en los laboratorios de la UNIVERSIDAD POLITECNICA SALESIANA?***

Si	10	100%
No	0	0%

Tabla 12: Tabulación 3

Fuente: El Autor



**Gráfico 46: Diagrama de Barras pregunta 2**

**Fuente: El Autor**

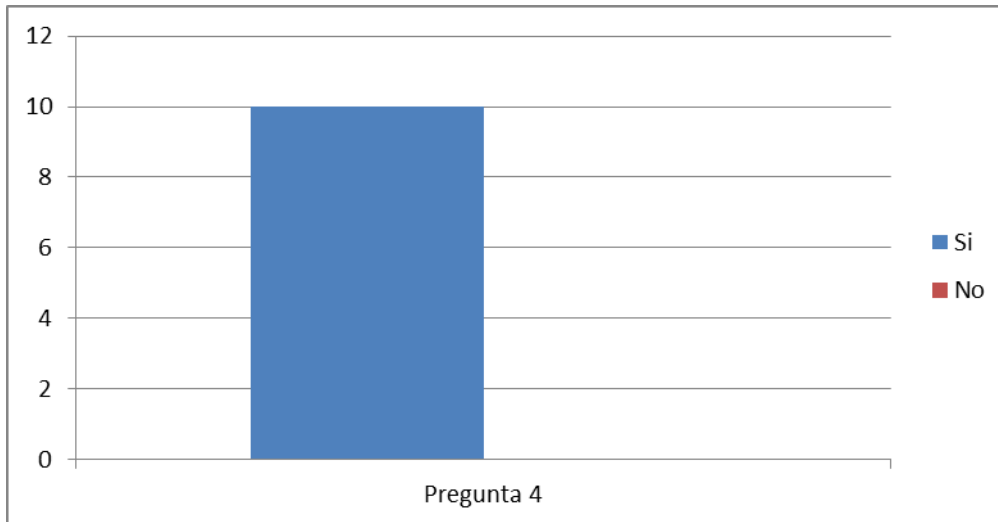
Todos los docentes encuestados coinciden en la necesidad de instalar un sistema de control de acceso a los laboratorios y de monitoreo de los activos de todos los laboratorios de las diferentes facultades.

***Cree usted ¿qué el sistema facilitaría el control del ingreso y acceso a los laboratorios?***

Si	10	100%
No	0	0%

**Tabla 13: Tabulación 4**

**Fuente: El Autor**



**Gráfico 47: Diagrama de Barras pregunta 2**

Fuente: El Autor

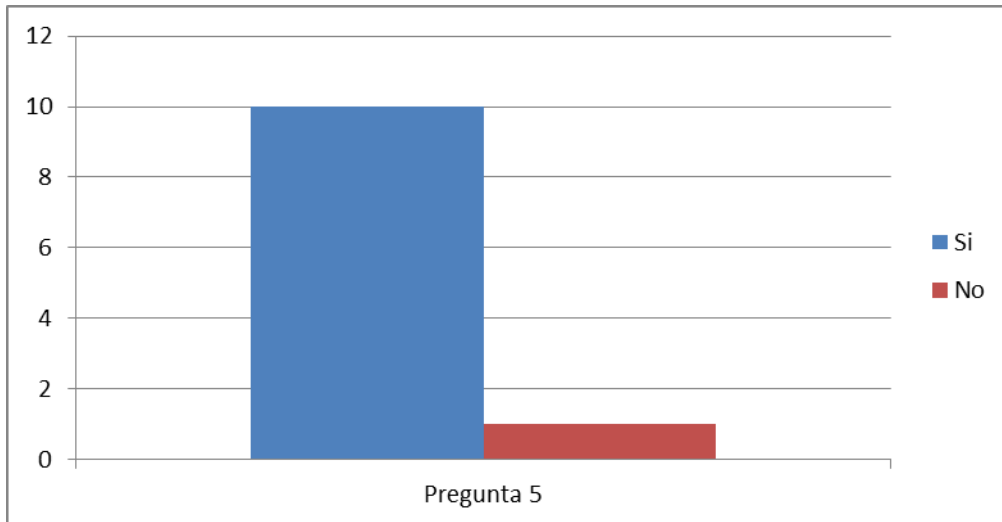
Los docentes manifestaron estar de acuerdo con que el sistema facilitaría el acceso y el control de los activos de los laboratorios.

***Cree usted ¿qué se reduciría el índice de pérdidas de los equipos o materiales de los laboratorios mediante la implementación del sistema?***

Si	10	100%
No	1	10%

**Tabla 14: Tabulación 5**

Fuente: El Autor



**Gráfico 48: Diagrama de Barras pregunta 2**

**Fuente: El Autor**

En su mayoría, los docentes, convergen en que la pérdida de los activos se reduciría con la implementación de este sistema; uno de los encuestados aduce que de todos modos encontrarán la forma de hurtar los equipos a pesar de la implementación de esta medida.

### **CAPÍTULO 3**



### **3. Análisis y diseño del Sistema propuesto**

#### **3.1 Arquitectura del Sistema**

El prototipo que se realizará es un sistema de control de acceso y monitoreo que consiste en controlar el acceso a un lugar determinado, es decir al laboratorio; ya que el fin de este proyecto es de dar acceso a las personas autorizadas y controlar los activos fijos que se encuentren dentro de él.

El sistema consiste en dos módulos:

- Acceso y Monitoreo



**Gráfico 49: Arquitectura del Sistema**

**Fuente: El autor**

### **3.1.1 Descripción General del Sistema**

Este prototipo tiene como fin minimizar las posibilidades de hurto, descargar el trabajo a los encargados de abrir los laboratorios y ayudar a que las clases empiecen a tiempo.

#### **ACCESO**

El Sistema de Control de Acceso de Laboratorio Telemática UPS, tiene por objetivo capturar:

- Ingreso de Personal Autorizado
- Horas hábiles en las cuales el personal autorizado tiene acceso al laboratorio
- Reporte de Ingresos del personal autorizado
- Reporte de equipamiento incluido en el laboratorio
- Reporte de personal autorizado
  - Profesores
  - Personal Administrativo

El sistema se encuentra dividido en 3 partes primordiales:

- Interfaz Gráfica
- Interfaz de Comunicación
- Lector de Tarjetas

Ésta información será ingresada mediante el uso de una tarjeta lectora, la cual conectada a un controlador enviará datos al sistema para que permita identificar si el profesor poseedor de la tarjeta tiene o no permitida la entrada.

#### **Interfaz Gráfica**

La interfaz gráfica permitirá realizar:

#### **Ingreso de Profesores**

La pantalla de Ingreso de profesores es la encargada de realizar el ingreso de los profesores y los códigos de tarjetas asignadas a cada uno de ellos.

### **Ingreso de Disponibilidad de horarios de Ingreso de Profesores**

La pantalla de Ingreso de Disponibilidad de horarios permite seleccionar a los profesores y asignarles un horario en el que puede ingresar al laboratorio.

### **Reporte de Horario de Profesores**

La pantalla de Reporte de Horario de Profesores permite visualizar la información de los horarios permitidos para cada profesor.

### **Reporte de Inventario de Laboratorio**

En la pantalla de inventario encontrará la información de los elementos de los activos que posee en stock el laboratorio; es decir, cuantas computadoras, impresoras, etc hay en existencia.

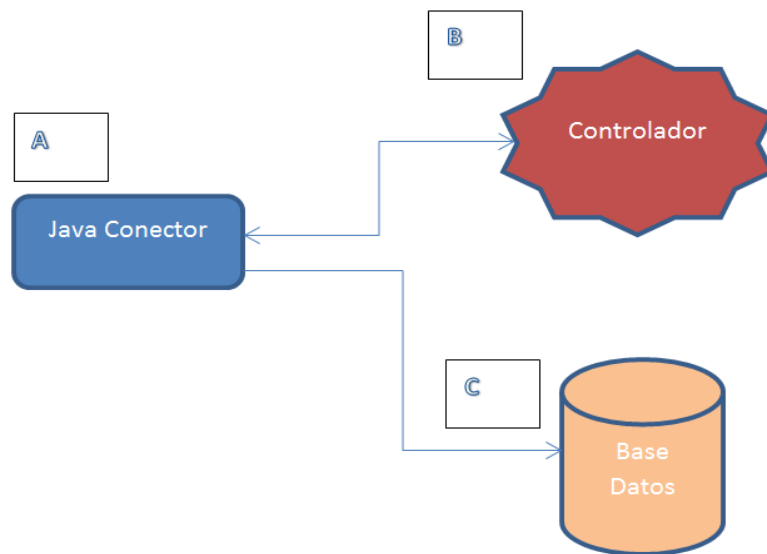
### **Reporte de Accesos al Laboratorio de Telemática**

En la siguiente pantalla se encontrarán los distintos accesos o intentos de acceso al laboratorio. De ésta manera se puede visualizar el trabajo realizado por el proceso Java Conector y el controlador.

### **Interfaz de Comunicación y Lector de Tarjetas**

La interfaz de comunicación está implementada en java con jdk 1.7. Este aplicativo se mantendrá ejecutándose permanentemente en el equipo servidor al cual va a estar conectado el controlador.

En el gráfico a continuación se identifica el proceso de ejecución del mismo.



**Gráfico 50: Proceso de ejecución del Sistema**

**Fuente: El autor**

A. Java Conector se encontrará ejecutándose en espera del parámetro de comunicación por el puerto UDP asignado durante la instalación.

Una vez receptado el comando enviado por el controlador, el Java Conector tomará el comando y lo comparará con el código de tarjeta asignado a cada profesor en la base de datos. Al momento de realizar la comparación, verificará si el profesor tiene habilitado su ingreso al laboratorio. En caso de que el profesor tenga permitido el acceso al laboratorio, se asignará en el comando el bit de asignación en ON, de ésta manera se lo enviará hacia el controlador permitiendo la apertura de la puerta del laboratorio; caso contrario, la puerta no se abrirá.

B. El controlador tiene la programación para receptar el impulso enviado y generado por el acercamiento de la tarjeta del profesor y el lector de la tarjeta. Además, será el encargado de enviar la comunicación por el puerto UDP al cual estará permanentemente conectado el proceso Java Conector. El controlador

tiene la función de recibir el comando enviado por el proceso Java Conector, el cual le indicará si debe o no abrir la puerta para el paso del profesor.

C. La base de datos almacena la información de los números de tarjeta y el horario que tiene permitido el ingreso de cada profesor. Es aquí de donde el proceso Java Conector obtendrá la información para validar el ingreso o no de los profesores al laboratorio.

### 3.2 Modelo de Análisis

#### 3.2.1 Diagrama de Clases

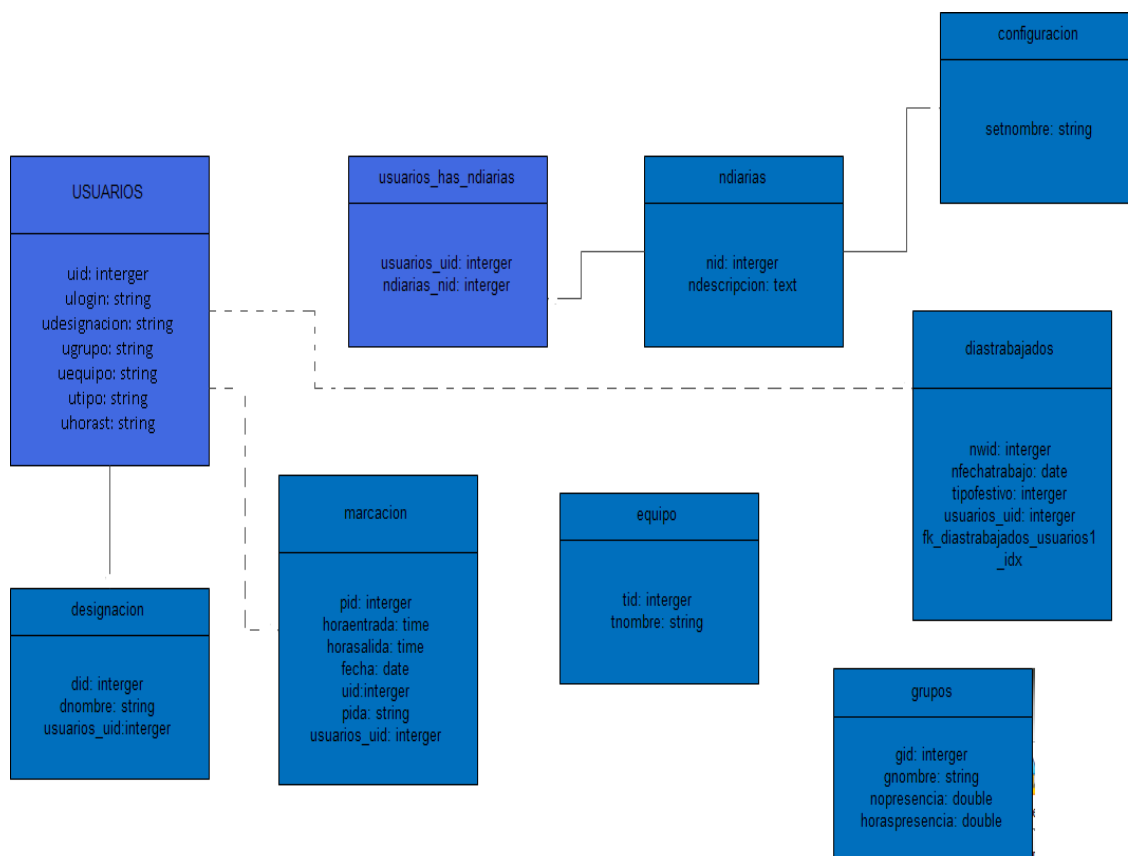


Gráfico 51: Diagrama de Clases

Fuente: El Autor

### 3.2.2 Diagrama de Actividad

#### Función del Administrador

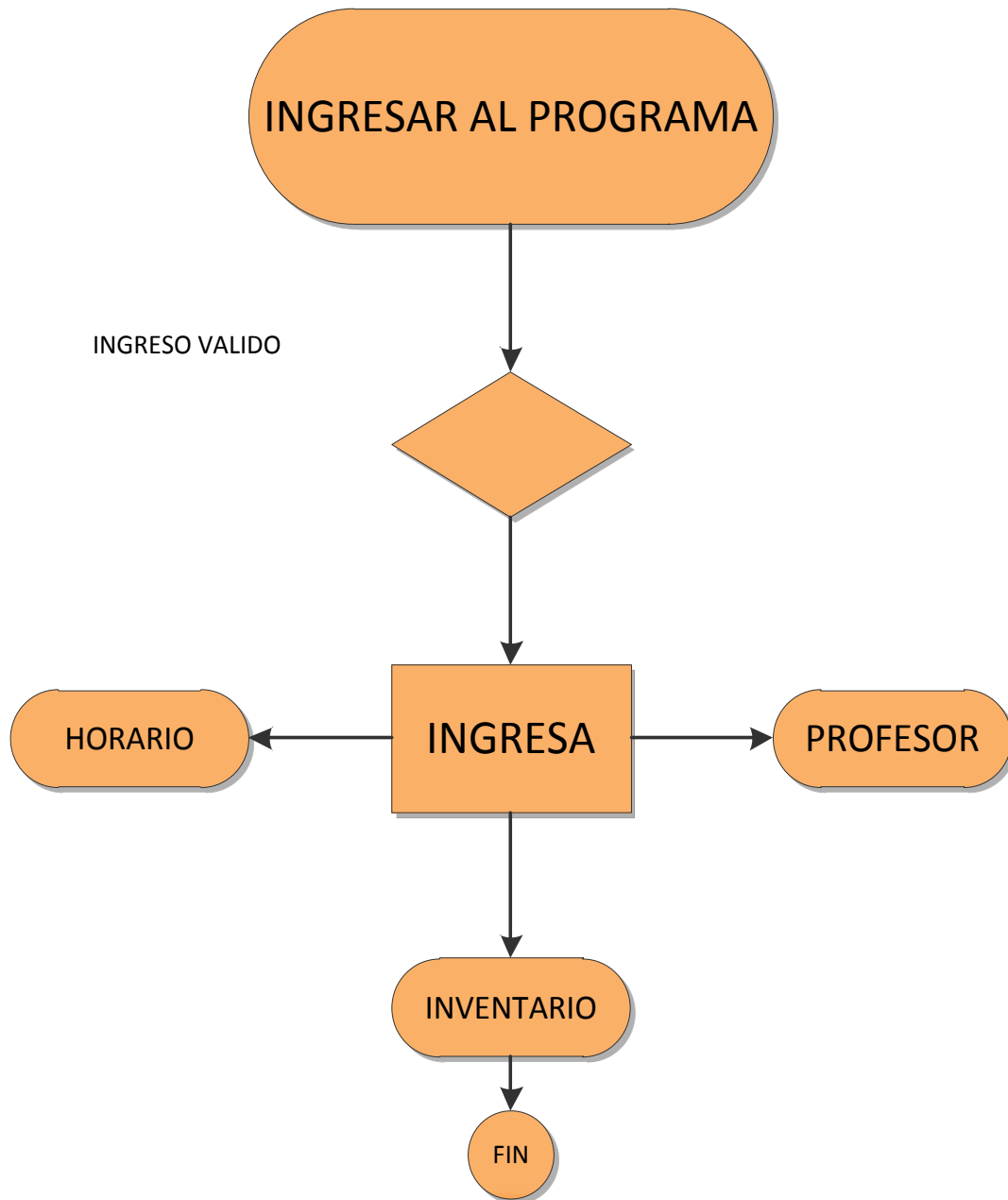


Gráfico 52: Diagrama de Actividad – Función Administrador

Fuente: El Autor

## Función del Supervisor

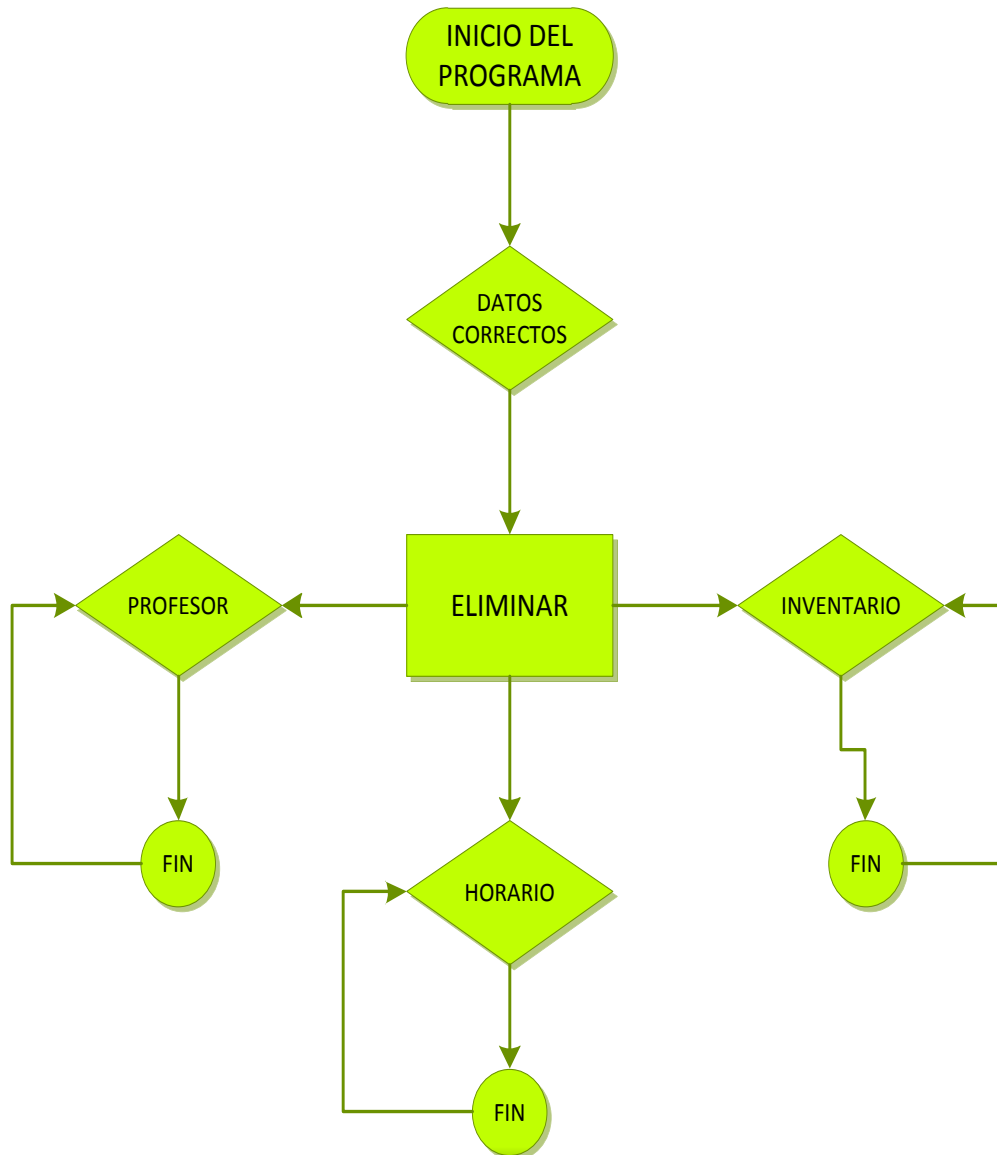


Gráfico 53: Diagrama de Actividad – Función Supervisor

Fuente: El Autor

### 3.2.3 Diagrama de Despliegue del Software

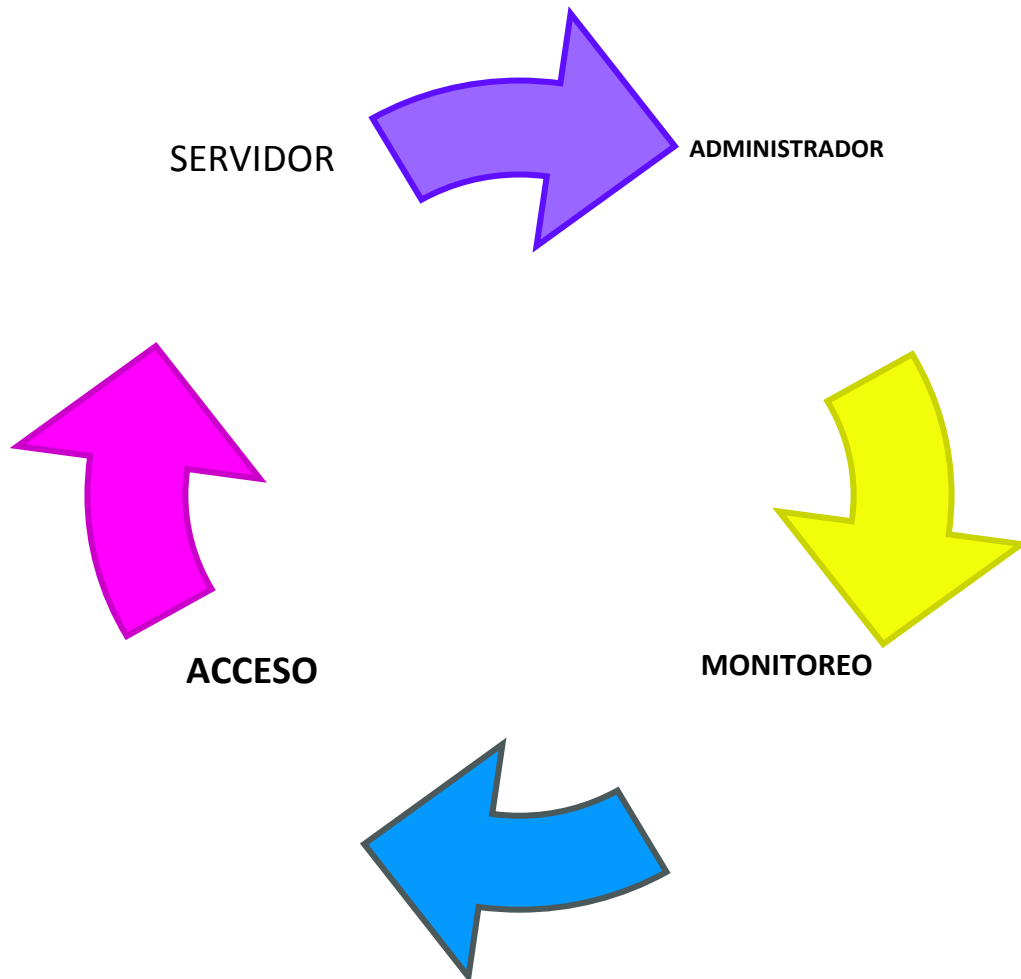


Gráfico 54: Diagrama de Despliegue de Software

Fuente: El Autor



### **3.2.4 Diagramas de Casos de Usos**

Se han determinado los siguientes actores:

#### **1. Administrador**

**Realiza las siguientes acciones:**

Se logonea al sistema, ingresa al menú para:

- ✓ Ingresar el inventario de las computadoras.
- ✓ Los horarios establecidos de los docentes
- ✓ Ingresar los nombres de los profesores con sus respectivas materias.
- ✓ También genera reportes de los profesores ingresados y del inventario de los activos del laboratorio.

#### **2. Supervisor**

**Realiza las siguientes acciones:**

Se logonea al sistema, ingresa al menú para:

- ✓ Actualizar el inventario.
- ✓ Actualizar los horarios de profesores.
- ✓ Añadir nuevas horas.
- ✓ Generar reportes de horarios.
- ✓ Revisar horas de entradas y salidas.
- ✓ Revisar Inventario.
- ✓ Añadir profesores.
- ✓ Actualizar horas.

## Caso de Uso Autenticación de Usuario



ADMINISTRADOR



PERFIL DE



SUPERVISOR



SE INGRESA AL

SISTEMA

USUARIO Y



SE VALIDA EL

INGRESO

Gráfico 55: Caso de Uso - Autenticación de Usuario

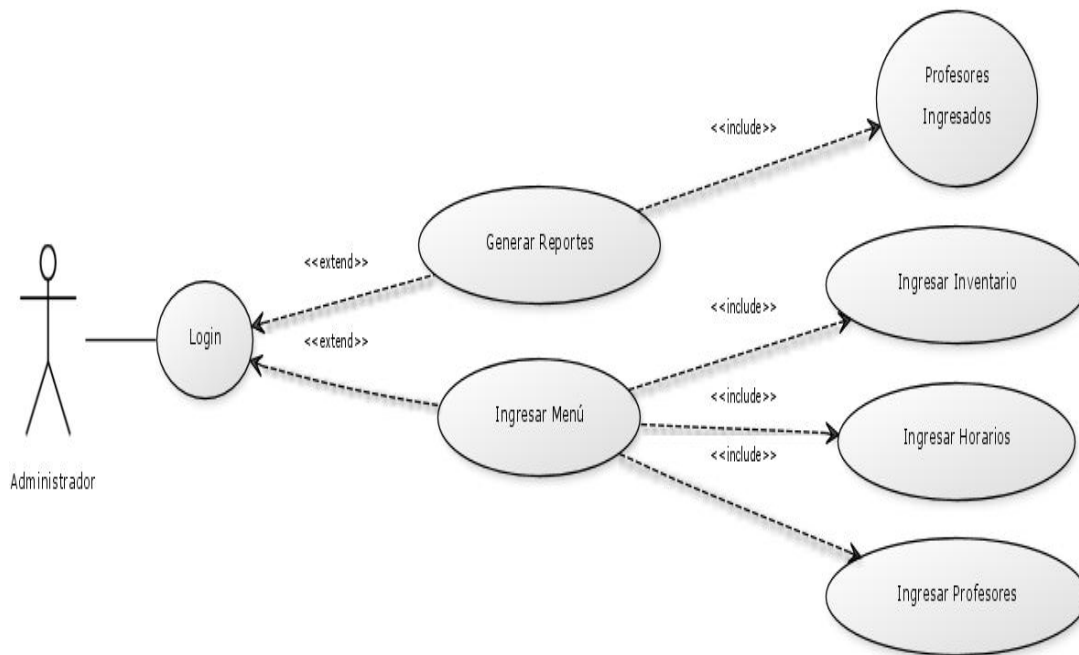
Fuente: El Autor

<b>DESCRIPCIÓN CASO DE USO AUTENTICACIÓN DE USUARIO</b>	
<b>ACTORES</b>	Administrador, Supervisor y Base de Datos.
<b>OBJETIVO</b>	Ingresar al Sistema.
<b>DESCRIPCIÓN</b>	Se ingresa con un User y un Password..
<b>PRECONDICIÓN</b>	Ninguna.
<b>SECUENCIA</b>	El usuario digita su usuario y contraseña. El sistema comprueba la información. El sistema admite al usuario.

Tabla 15 Casos de Uso Módulo Usuario

Fuente: El Autor

## Caso de Uso Módulo Administrador



**Gráfico 56: Casos de Usos Módulo Administrador**

**Fuente: El Autor**

<b>DESCRIPCIÓN CASOS DE USO MÓDULO ADMINISTRADOR</b>	
<b>ACTORES</b>	Administrador, Base de Datos.
<b>OBJETIVO</b>	Ingresar al Sistema como Administrador.
<b>DESCRIPCIÓN</b>	Ingresar su usuario y su contraseña.
<b>PRECONDICION</b>	Tiene que haber ingresado como Administrador.
<b>SECUENCIA</b>	Elije la opción que necesite.

**Tabla 17 Casos de Uso Módulo Administrador**

**Fuente: El Autor**

## Caso de Uso Módulo Supervisor

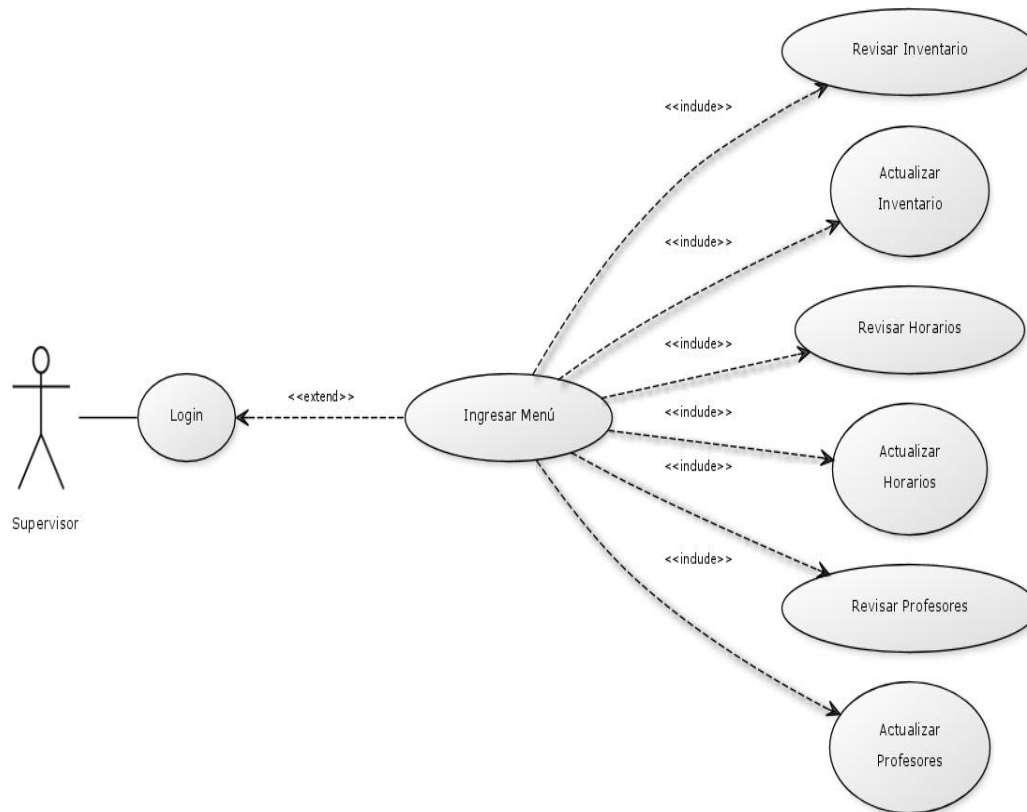


Gráfico 57: Módulo Supervisor

Fuente: El Autor

<b>DESCRIPCIÓN CASOS DE USO MÓDULO SUPERVISOR</b>	
<b>ACTORES</b>	Supervisor, Base de Datos.
<b>OBJETIVO</b>	Ingresar al Sistema como Supervisor.
<b>DESCRIPCION</b>	Ingresar su usuario y su contraseña.
<b>PRECONDICION</b>	Tiene que haber ingresado como Supervisor.
<b>SECUENCIA</b>	Elije la opción que necesite.

Tabla 18 Casos de Uso Módulo Supervisor

Fuente: El Autor

## Casos de Uso Ingreso de Docentes

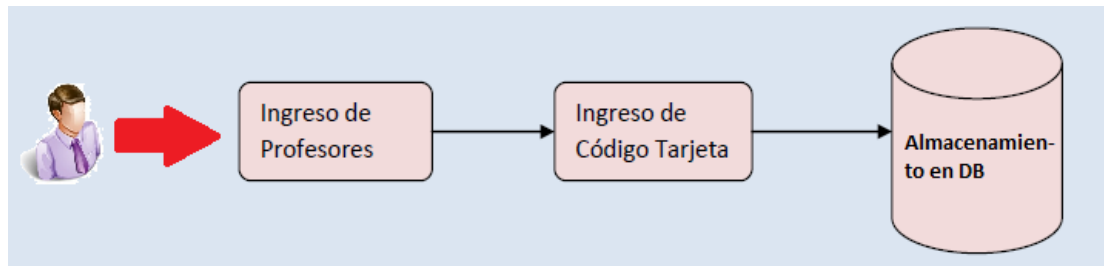


Gráfico 58: Caso de Uso - Ingreso De Docentes

Fuente: El Autor

<b>DESCRIPCIÓN CASOS DE USO INGRESO DE DOCENTES</b>	
<b>ACTORES</b>	Supervisor, Base de Datos.
<b>OBJETIVO</b>	Ingresar al Sistema como Supervisor.
<b>DESCRIPCIÓN</b>	Ingresar su usuario y su contraseña.
<b>PRECONDICIÓN</b>	Tiene que haber ingresado como Supervisor.
<b>SECUENCIA</b>	Ingresar los profesores. Ingresar los códigos de las tarjetas. Se almacena en la base de datos.

Tabla 19 Casos de Usos Ingreso de Docentes

Fuente: El Autor

## Caso de Uso Creación de Registro

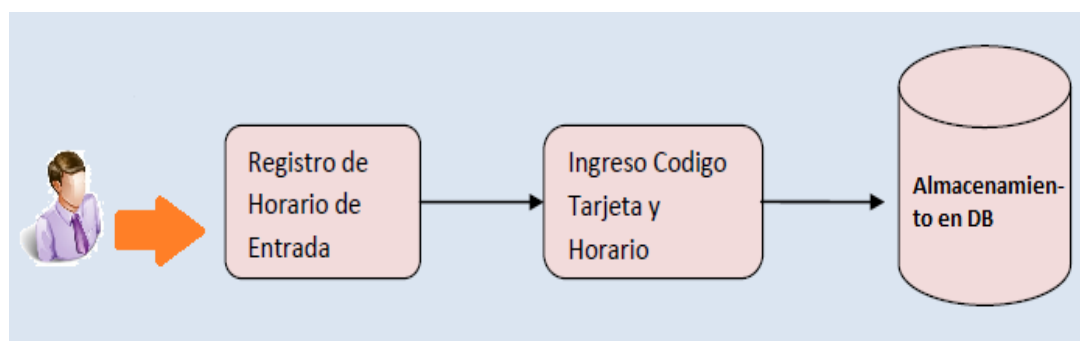


Gráfico 59: Casos de Uso - Creación de Registro de Horarios

Fuente: El Autor

<b>DESCRIPCIÓN CASOS DE USO INGRESO DE DOCENTES</b>	
<b>ACTORES</b>	Supervisor, Base de Datos.
<b>OBJETIVO</b>	Ingresa al Sistema como Supervisor.
<b>DESCRIPCIÓN</b>	Ingresa su usuario y su contraseña.
<b>PRECONDICIÓN</b>	Tiene que haber ingresado como Supervisor.
<b>SECUENCIA</b>	Ingresa los profesores. Ingresa los códigos de las tarjetas. Se almacena en la base de datos.

Tabla 20 Casos de Uso Ingreso de Docentes

Fuente: El Autor

### **Casos de Uso Conexión**

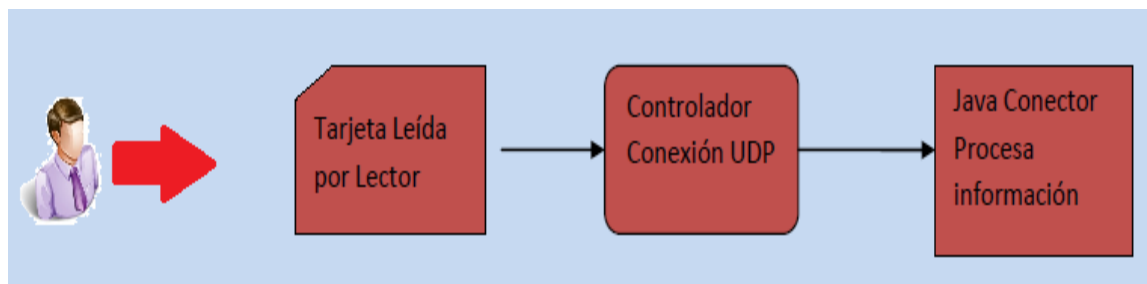


Gráfico 60: Caso de Uso-Conexión

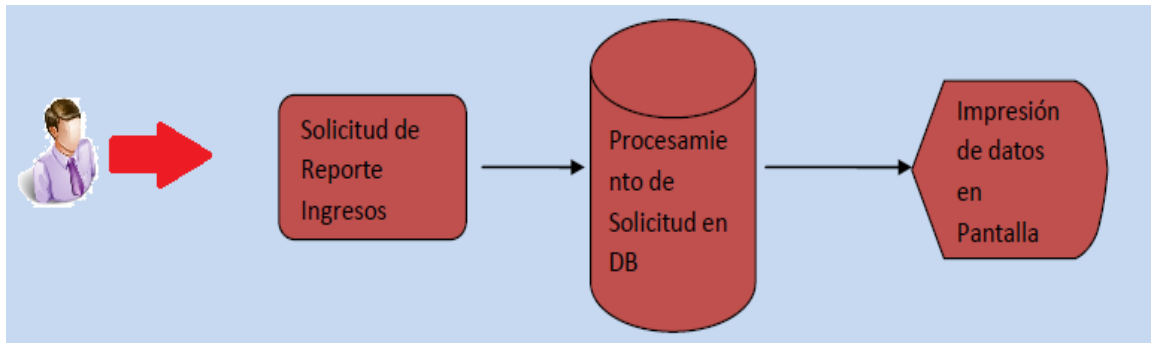
Fuente: El Autor

<b>DESCRIPCIÓN CASOS DE USO CONEXIÓN</b>	
<b>ACTORES</b>	Supervisor, Base de Datos.
<b>OBJETIVO</b>	La tarjeta es leída por el lector.
<b>DESCRIPCIÓN</b>	Conexión UDP.
<b>PRECONDICIÓN</b>	Java Conector procese información.
<b>SECUENCIA</b>	Se almacena en la base de datos.

Tabla 21 Casos de Uso Conexión

Fuente: El Autor

### **Casos de Uso Solicitud de Reportes**



**Gráfico 61: Casos de Usos - Solicitud de Reportes**

Fuente: El Autor

<b>DESCRIPCIÓN CASOS SOLICITUD DE REPORTES</b>	
<b>ACTORES</b>	Supervisor, Base de Datos.
<b>OBJETIVO</b>	Reportes de los datos.
<b>DESCRIPCIÓN</b>	El supervisor ingresa la solicitud de Reporte de Ingresos.
<b>PRECONDICIÓN</b>	El procesamiento de Bases de Datos.
<b>SECUENCIA</b>	El Supervisor ingresa al sistema. El Supervisor emite la petición se reporte. Se procesan los datos. Se imprimen los reportes en pantalla.

**Tabla 22 Casos de Uso Solicitud de Reportes**

Fuente: El Autor

### **3.2.5 Diagramas de Interacción**

El diagrama de interacción, representa la forma en como un Cliente (Actor) u Objetos (Clases) se comunican entre sí en petición a un evento.

Esto implica recorrer toda la secuencia de llamadas, de donde se obtienen las responsabilidades claramente.

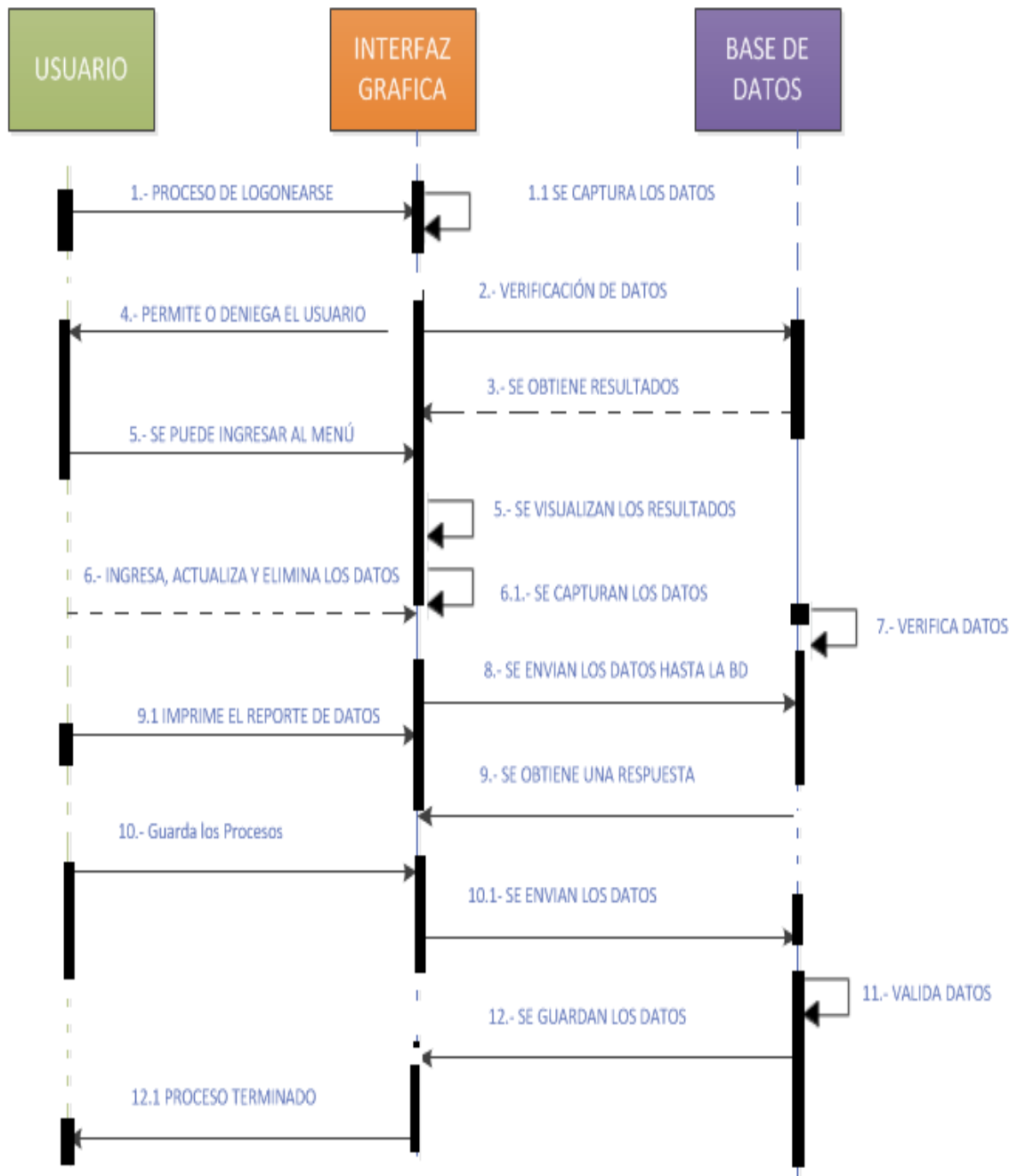


Gráfico 62: Diagrama de Interacción

Fuente: El Autor

### 3.3 Diseño de la Arquitectura



### 3.3.1 Capa de Base de Datos

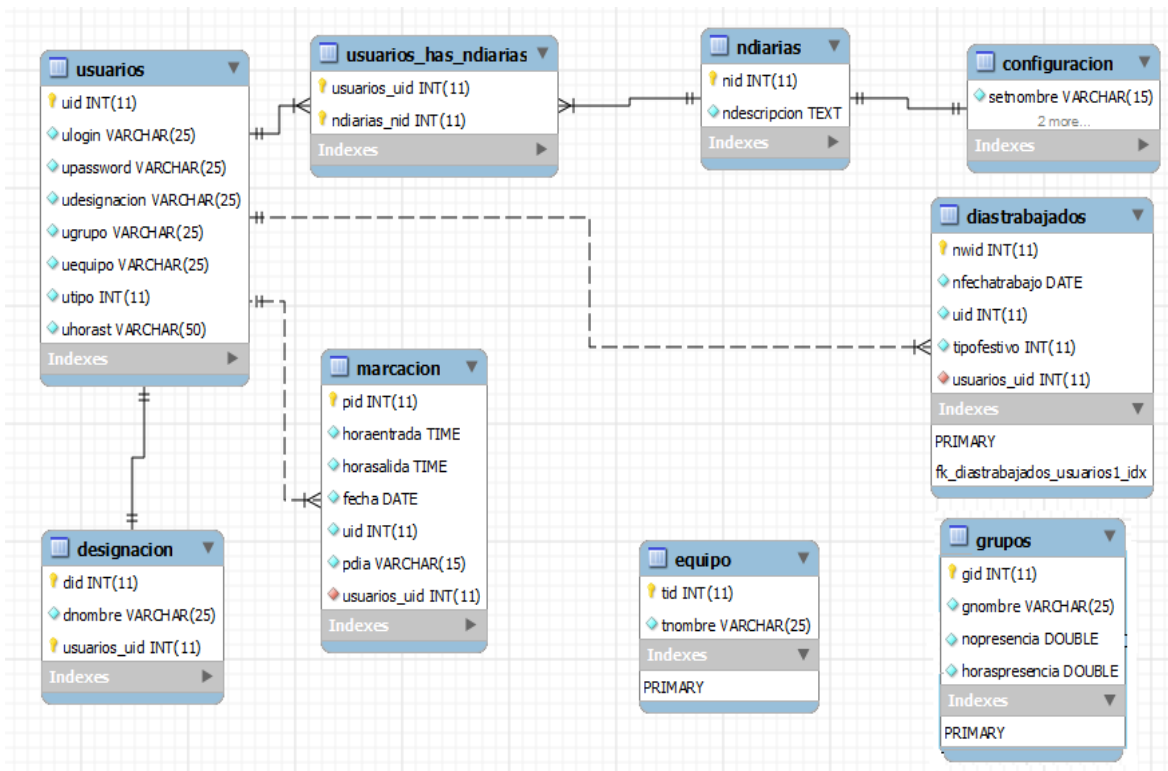


Gráfico 63: Capa de Base de Datos

Fuente: El Autor

#### Definición de las Tablas del Sistema

Se detallan tablas del Sistema SCAL:

- **Usuarios**

Ésta tabla es donde designamos todos los campos del programa.

COLUMNA	TIPO DE DATOS	DETALLE	DESCRIPCIÓN
Uid	INTEGER (11)	Primary Key Not Null.	Id del usuario
ulogin	VARCHAR(25)		El usuario para iniciar sesión
upassword	VARCHAR(25)		Contraseña para iniciar

			sesión
<b>udesignacion</b>	VARCHAR(25)		Demarca los perfiles de usuario, y demarca que tipo de persona tiene acceso al laboratorio
<b>ugrupo</b>	VARCHAR(25)		Para separar las tarjetas
<b>uequipo</b>	VARCHAR(25)		Se almacena el inventario de los activos del laboratorio
<b>utipo</b>	INTEGER (11)		Tipo de equipo de los activos del laboratorio
<b>uhorast</b>	VARCHAR(25)		El tiempo donde el usuario ha pasado la tarjeta por el lector.

Tabla 23 Diccionario de Datos – Tabla Usuarios

Fuente: El Autor

- **Marcaciones Diarias de los Usuario (Usuarios\_has\_nd)**

Ésta tabla describe las marcaciones que se hacen diariamente para luego emitir un reporte.

<b>COLUMNA</b>	<b>TIPO DE DATOS</b>	<b>DETALLE</b>	<b>DESCRIPCIÓN</b>
<b>Usuarios_uid</b>	INTERGER(11)	PRIMARY KEY	Id de usuario
<b>Ndiarias_nid</b>	INTERGER(11)	PRIMARY KEY	Id de marcaciones

Tabla 24 Diccionario de Datos – Marcaciones Diarias de los Usuarios

Fuente: El Autor

- **Designación**

Ésta tabla es para la división entre el tipo de las tarjetas es decir los diferentes ids de usuarios ya sea de los profesores, los conserjes, etc.

COLUMNA	TIPO DE DATOS	DETALLE	DESCRIPCIÓN
Idid	INTERGER (11)	PRIMARY KEY	ID
Dnombre	VARCHAR (25)		División entre las id de tarjetas
Usuarios_uid	INTERGER (11)		Los id de usuarios de las tarjetas

Tabla 25 Diccionario de Datos - Designación

Fuente: El Autor

- **Marcación**

Ésta tabla describe la fecha, la hora de entrada, la hora de salida en que el usuario ha marcado para acceder al laboratorio.

COLUMNA	TIPO DE DATOS	DETALLE	DESCRIPCIÓN
Pid	INTERGER (11)	PRIMARY KEY	Id de la tarjeta
Horaentrada	TIME		La hora de entrada que marco el usuario
Horasalida	TIME		La hora de salida que marco el usuario
Fecha	DATE		La fecha en que el usuario marco

Tabla 26 Diccionario de Datos - Marcación

Fuente: El Autor

- **Ndiarias**

COLUMNA	TIPO DE DATOS	DETALLE	DESCRIPCIÓN
Nid	INTERGER (11)	PRIMARY KEY	ID de las entradas diarias
Ndescripcion	TEXT		La descripción de las entradas por persona

Tabla 27 Diccionario de Datos - Ndiarias

Fuente: El Autor

- **Configuración**

Guarda los datos de configuración del sistema los horarios

COLUMNA	TIPO DE DATOS	DETALLE	DESCRIPCIÓN
Setnombre	VARCHAR (15)		El nombre de la tarjeta

Tabla 28 Diccionario de Datos Reportes

Fuente: El Autor

- **Días Trabajados**

Ésta tabla describe los registros de los días trabajados con fecha, usuario y tipo.

COLUMNA	TIPO DE DATOS	DETALLE	DESCRIPCIÓN
Nwid	INTERGER (11)		ID para realizar un reporte
Nfechatrabajo	DATE		Registro de la fecha en que se marco

Uid	INTERGER (11)		id
Tipofestivo	INTERGER (11)		Elección de un día que no este agendado
Usuario_uid	INTERGER (11)		El id del usuario

Tabla 29 Diccionario de Datos – Días trabajados

Fuente: El Autor

- **Equipo**

Ésta tabla describe el nombre del activo del laboratorio.

COLUMNA	TIPO DE DATOS	DETALLE	DESCRIPCIÓN
Tid	INTERGER (11)	PRIMARY KEY	ID del activo del laboratorio.
tnombre	VARCHAR (25)		Nombre del activo que está en el inventario

Tabla 30 Diccionario de Datos - Equipos

Fuente: El Autor

- **Grupos**

COLUMNA	TIPO DE DATOS	DETALLE	DESCRIPCIÓN
Gid	INTERGER (11)	PRIMARY KEY	ID del activo del laboratorio.
gnombre	VARCHAR (25)		Nombre del activo que está en el inventario
Nopresencia	DOUBLE		

horaspresencia	DOUBLE		
----------------	--------	--	--

Tabla 31 Diccionario de Datos Reportes - Grupos

Fuente: El Autor

### 3.4 Módulo de Inicio de Sesión de Administrador

**Ingreso al sistema:**



Gráfico 64: Ingreso al Sistema

Fuente: El Autor

#### **Código de Inicio de Sesión**

```
namespace BusinessApplication1.LoginUI
{
    using System;
    using System.ComponentModel.DataAnnotations;
    using System.ServiceModel.DomainServices.Client.ApplicationServices;
    using System.Windows;
    using System.Windows.Controls;
    using System.Windows.Input;
```

```
/// <summary>
```

```
/// Formulario que presenta los campos de inicio de sesión y  
controla el proceso de inicio de sesión.
```

```
/// </summary>
```

```
public partial class LoginForm : StackPanel
```

```
{
```

```
    private LoginRegistrationWindow parentWindow;
```

```
    private LoginInfo loginInfo = new LoginInfo();
```

```
    private TextBox userNameTextBox;
```

```
/// <summary>
```

```
/// Crea una nueva instancia de <see cref="LoginForm"/>.
```

```
/// </summary>
```

```
public LoginForm()
```

```
{
```

```
    InitializeComponent();
```

```
        // Establezca el DataContext de este control en la  
instancia de LoginInfo para facilitar los enlaces.
```

```
        this.DataContext = this.loginInfo;
```

```
    }
```

```
/// <summary>
```

```
/// Establece la ventana primaria del <see cref="LoginForm"/> actual.
```

```
/// </summary>
```

```

        /// <param name="window">Ventana que se va a utilizar como
        primaria.</param>

public void SetParentWindow(LoginRegistrationWindow window)

{
    this.parentWindow = window;
}

/// <summary>

    /// Controla el <see cref="DataForm.AutoGeneratingField"/> para
    proporcionar el PasswordAccessor.

/// </summary>

    private void LoginForm_AutoGeneratingField(object sender,
    DataFormAutoGeneratingFieldEventArgs e)

{
    if (e.PropertyName == "UserName")

    {
        this.userNameTextBox = (TextBox)e.Field.Content;
    }
    else if (e.PropertyName == "Password")

    {
        PasswordBox passwordBox = new PasswordBox();

        e.Field.ReplaceTextBox(passwordBox,
        PasswordBox.PasswordProperty);

        this.loginInfo.PasswordAccessor = () => passwordBox.Password;
    }
}

/// <summary>

```

```

    /// Envía la <see cref="LoginOperation"/> al servidor

```

```

    /// </summary>

```

```

private void LoginButton_Click(object sender, EventArgs e)

```

```

{

```



***// Es necesario forzar la validación, ya que no se está utilizando el botón Aceptar estándar del DataForm.***

***// Si no se garantiza que el formulario sea válido, se obtiene una excepción que invoca la operación en caso de que la entidad no sea válida.***

```
        if (this.loginForm.ValidateItem())  
  
        {  
  
            this.loginInfo.CurrentLoginOperation =  
            WebContext.Current.Authentication.Login(this.loginInfo.ToLoginParameters(),  
            this.LoginOperation_Completed, null);  
  
this.parentWindow.AddPendingOperation(this.loginInfo.CurrentLoginOperation);  
  
        }  
  
    }  
  
    /// <summary>  
  
    /// Controlador de finalización de una <see cref="LoginOperation"/>.  
  
    /// Si la operación es correcta, cierra la ventana.  
  
    /// Si tiene un error, muestra una <see cref="ErrorWindow"/>  
  
    y marca el error como controlado.  
  
    /// Si no se canceló pero se produjo un error de inicio de sesión,  
debe haber sido porque las credenciales eran incorrectas,  
así que se agrega un error de validación para notificar al usuario.
```

```
    /// </summary>  
  
    private void LoginOperation_Completed(LoginOperation loginOperation)  
    {  
  
        if (loginOperation.LoginSuccess)
```

```

    {
        this.parentWindow.DialogResult = true;
    }
    else if (loginOperation.HasError)
    {
        ErrorWindow.CreateNew(loginOperation.Error);
        loginOperation.MarkErrorAsHandled();
    }
    else if (!loginOperation.IsCanceled)
    {
        this.loginInfo.ValidationErrors.Add(new
        ValidationResult(ErrorResources.ErrorBadUserNameOrPassword,
        new string[] { "UserName", "Password" }));
    }
}

/// <summary>
/// Cambia al formulario de registro.
/// </summary>
private void RegisterNow_Click(object sender, RoutedEventArgs e)
{
    this.parentWindow.NavigateToRegistration();
}

```

```
/// <summary>
```

```
/// Si hay una operación de inicio de sesión en curso y se puede  
cancelar, hágalo.
```

```
/// De lo contrario, cierre la ventana.
```

```
/// </summary>
```

```
private void CancelButton_Click(object sender, EventArgs e)
```

```
{
```

```
    if (this.loginInfo.CurrentLoginOperation != null &&  
        this.loginInfo.CurrentLoginOperation.CanCancel)
```

```
    {
```

```
        this.loginInfo.CurrentLoginOperation.Cancel();
```

```
    }
```

```
    else
```

```
    {
```

```
        this.parentWindow.DialogResult = false;
```

```
    }
```

```
}
```

```
/// <summary>
```

```
/// Asigna Esc al botón Cancelar y Entrar al botón Aceptar.
```

```
/// </summary>
```

```
private void LoginForm_KeyDown(object sender, KeyEventArgs e)
```

```
{
```

```
    if (e.Key == Key.Escape)
```

```

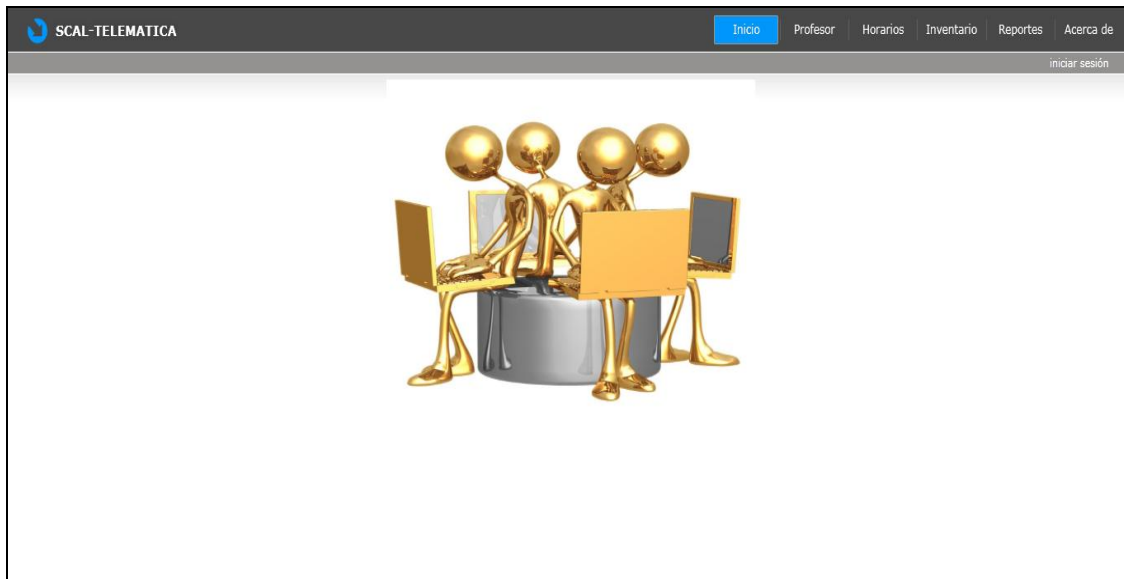
    {
        this.CancelButton_Click(sender, e);
    }
    else if (e.Key == Key.Enter && this.loginButton.IsEnabled)
    {
        this.LoginButton_Click(sender, e);
    }
}

/// <summary>
/// Establece el enfoque en el cuadro de texto de nombre de usuario.
/// </summary>

public void SetInitialFocus()
{
    this.userNameTextBox.Focus(); } } }

```

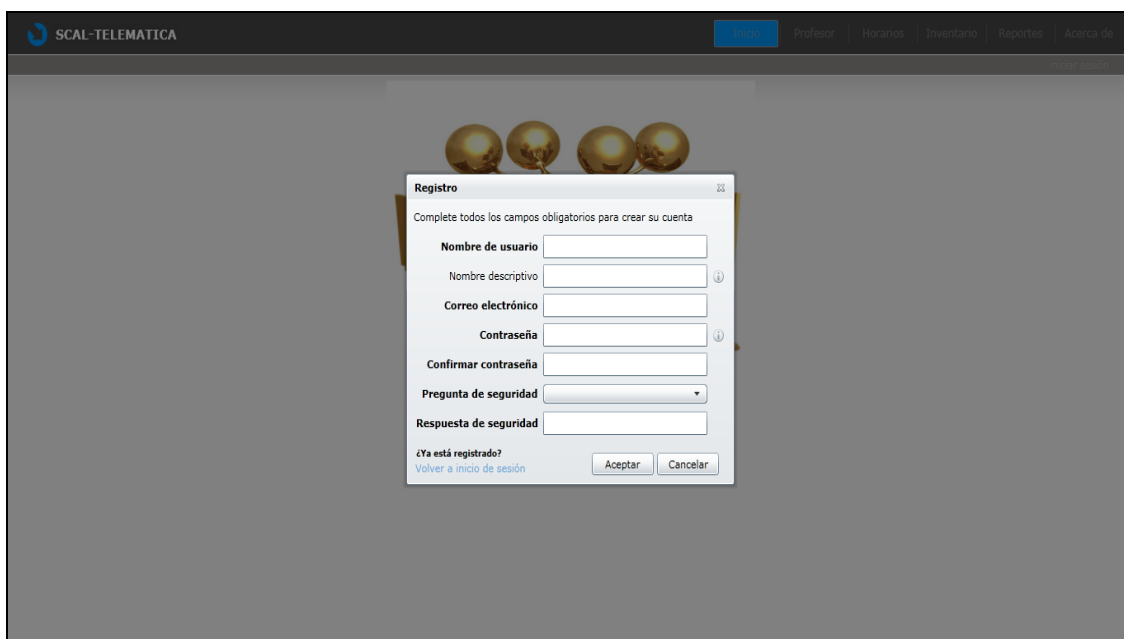
## ***Menú Principal***



**Gráfico 65: Menú Principal - Sistema SCAL**

**Fuente: El Autor**

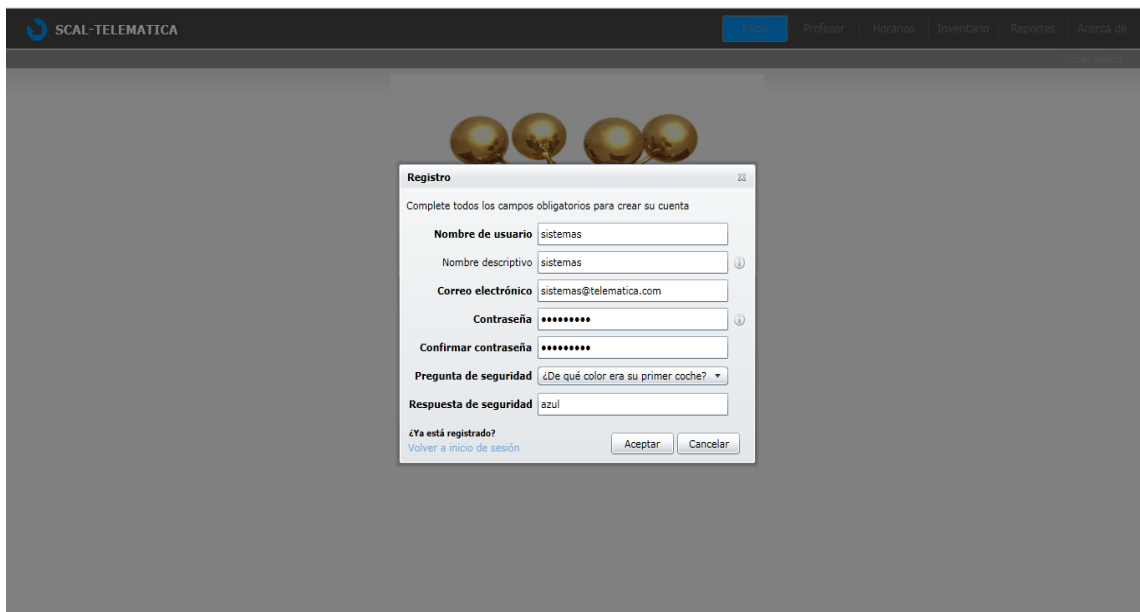
### ***Registro del Usuario Supervisor***



**Gráfico 66: Registro De Usuario - Sistema SCAL**

**Fuente: El Autor**

## Formulario de Registro SCAL

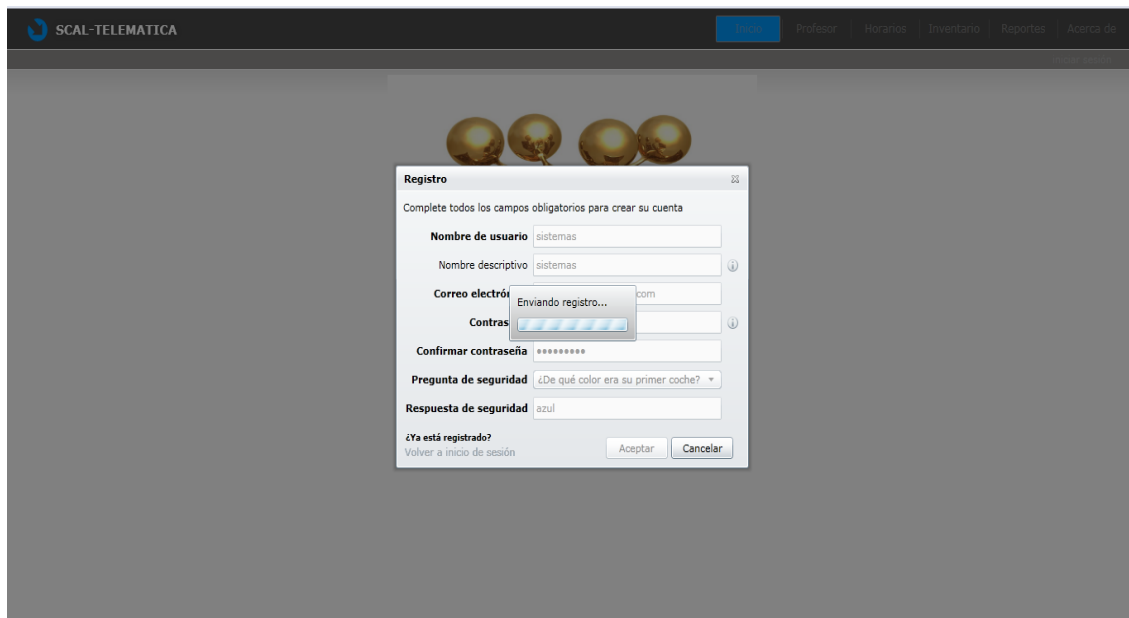


The screenshot shows the SCAL-TELEMATICA registration form. The form is titled "Registro" and includes the following fields: "Nombre de usuario" (sistemas), "Nombre descriptivo" (sistemas), "Correo electrónico" (sistemas@telematica.com), "Contraseña" (masked with dots), "Confirmar contraseña" (masked with dots), "Pregunta de seguridad" (¿De qué color era su primer coche?), and "Respuesta de seguridad" (azul). There are "Aceptar" and "Cancelar" buttons at the bottom right. A link "¿Ya está registrado? Volver a inicio de sesión" is at the bottom left.

Gráfico 67: Formulario de Registro SCAL

Fuente: El Usuario

## Almacenamiento de Datos



This screenshot is identical to the previous one, but with a loading spinner (a grey box with a blue progress bar) overlaid on the "Contraseña" field, indicating that the registration process is in progress.

Gráfico 68: Almacenamiento de Datos - Sistema SCAL

Fuente: El Usuario

## Código del Registro

```

namespace BusinessApplication1.LoginUI
{
using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.ServiceModel.DomainServices.Client;

using System.ServiceModel.DomainServices.Client.ApplicationServices;

using System.Windows;

using System.Windows.Controls;

/// <summary>

/// Clase <see cref="ChildWindow"/> que controla el proceso de registro.

/// </summary>

public partial class LoginRegistrationWindow : ChildWindow
{
    private IList<OperationBase> possiblyPendingOperations = new
    List<OperationBase>();

    /// <summary>

    /// Crea una nueva instancia de <see
    /// cref="LoginRegistrationWindow"/>.

    /// </summary>

    public LoginRegistrationWindow()
    {
        InitializeComponent();
    }
}

```

```

    this.registrationForm.SetParentWindow(this);

    this.loginForm.SetParentWindow(this);

    this.LayoutUpdated += this.GoToInitialState;

    this.LayoutUpdated += this.UpdateTitle;
}

/// <summary>

    /// Inicializa el <see cref="VisualStateManager"/> de este
    componente al establecerlo en estado "AtLogin".

/// </summary>

private void GoToInitialState(object sender, EventArgs eventArgs)
{
    this.LayoutUpdated -= this.GoToInitialState;

    VisualStateManager.GoToState(this, "AtLogin", true);
}

/// <summary>

/// Garantiza que el estado visual y el enfoque sean correctos
    al abrir la ventana.

/// </summary>

protected override void OnOpened()
{
    this.NavigateToLogin();
}

```



```
/// <summary>
```

```
/// Actualiza el título de la ventana según el panel
```

```
(registro o inicio de sesión) que se esté mostrando en ese momento.
```

```
/// </summary>
```

```
private void UpdateTitle(object sender, EventArgs eventArgs)
```

```
{
```

```
    this.Title = (this.registrationForm.Visibility == Visibility.Visible) ?
```

```
        ApplicationStrings.RegistrationWindowTitle :
```

```
        ApplicationStrings.LoginWindowTitle;
```

```
}
```

```
/// <summary>
```

```
/// Notifica a la ventana <see cref="LoginRegistrationWindow"/>
```

```
que sólo se puede cerrar si la <paramref name="operation"/>
```

```
termina o se puede cancelar.
```

```
/// </summary>
```

```
    /// <param name="operation">Operación pendiente que se va a  
    supervisor</param>
```

```
public void AddPendingOperation(OperationBase operation)
```

```
{
```

```
    this.possiblyPendingOperations.Add(operation);
```

```
}
```

```
/// <summary>
```

```
/// Hace que el <see cref="VisualStateManager"/> cambie al estado  
"AtLogin".
```

```
/// </summary>
```

```
public virtual void NavigateToLogin()
```

```
{
```

```
    VisualStateManager.GoToState(this, "AtLogin", true);
```

```
    this.loginForm.SetInitialFocus();
```

```
}
```

```
/// <summary>
```

```
    /// Hace que el <see cref="VisualStateManager"/> cambie al estado  
    "AtRegistration".
```

```
/// </summary>
```

```
public virtual void NavigateToRegistration()
```

```
{
```

```
    VisualStateManager.GoToState(this, "AtRegistration", true);
```

```
    this.registrationForm.SetInitialFocus();
```

```
}
```

```
/// <summary>
```

```
/// Evita que la ventana se cierre mientras hay operaciones en curso
```

```
/// </summary>
```

```
    private void LoginWindow_Closing(object sender, CancelEventArgs  
    eventArgs)
```

```
{
```

```
    foreach (OperationBase operation in this.possiblyPendingOperations)
```

```
{  
  if (!operation.IsComplete)  
  {  
    if (operation.CanCancel)  
    {  
      operation.Cancel();  
    }  
    else  
    {  
      eventArgs.Cancel = true;  
    }  
  }  
}  
}  
}  
}}
```

## **Conexión Cliente UDP**

```
//Lado del Cliente:

import java.io.*;

import java.net.*;

class UDPClient
{
public static void main(String args[]) throws Exception
{
    BufferedReader inFromUser =
        new BufferedReader(new InputStreamReader(System.in));

    DatagramSocket clientSocket = new DatagramSocket();

//Aquí se crea el socket del cliente

    InetAddress IPAddress = InetAddress.getByName("localhost");

//Obtiene la Dirección IP

    byte[] sendData = new byte[1024];

    byte[] receiveData = new byte[1024];

    String sentence = inFromUser.readLine();

    sendData = sentence.getBytes();

//envía información

    DatagramPacket sendPacket = new DatagramPacket(sendData,
        sendData.length, IPAddress, 9876);
```

```

clientSocket.send(sendPacket);

    DatagramPacket receivePacket = new DatagramPacket(receiveData,
receiveData.length);

clientSocket.receive(receivePacket);

String modifiedSentence = new String(receivePacket.getData());

System.out.println("Controlador:" + modifiedSentence);

clientSocket.close();

    }

}

//Lado del Servidor:

import java.io.*;

import java.net.*;

class UDPServer

{

public static void main(String args[]) throws Exception

{

DatagramSocket serverSocket = new DatagramSocket(9876);

//Aquí se crea el socket del servidor, para realizar prueba de
conectividad en caso de pérdida de comunicación

con el puerto UDP

byte[] receiveData = new byte[1024];

byte[] sendData = new byte[1024];

while(true)

{

```

```

DatagramPacket receivePacket = new DatagramPacket(receiveData,
receiveData.length);

    serverSocket.receive(receivePacket);

    String sentence = new String( receivePacket.getData());

    System.out.println("RECIBIDO: " + sentence);

    InetAddress IPAddress = receivePacket.getAddress();

    int port = receivePacket.getPort();

    String capitalizedSentence = sentence.toUpperCase();

    sendData = capitalizedSentence.getBytes();

    DatagramPacket sendPacket =

        new DatagramPacket(sendData, sendData.length, IPAddress,
port);

    serverSocket.send(sendPacket);

        }

    }

}

```

### 3.4 Descripción del Monitoreo Inalámbrico

El monitoreo inalámbrico fue realizado mediante LabVIEW<sup>21</sup> ideal para cualquier sistema de medidas y control y el corazón de la plataforma de diseño.

Interfaz Visual del Programa



**Gráfico69: Interfaz Visual del Programa**

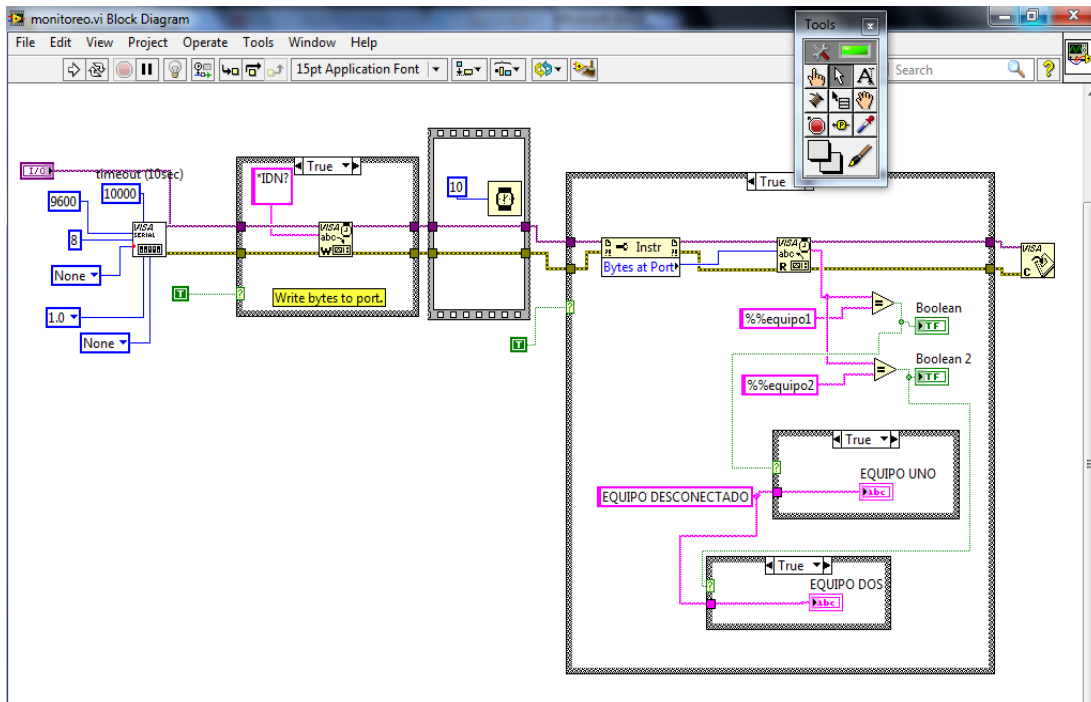
**Fuente: El autor**

El programa de monitoreo se divide en dos:

- Interfaz serial
- Procesamiento de señal

---

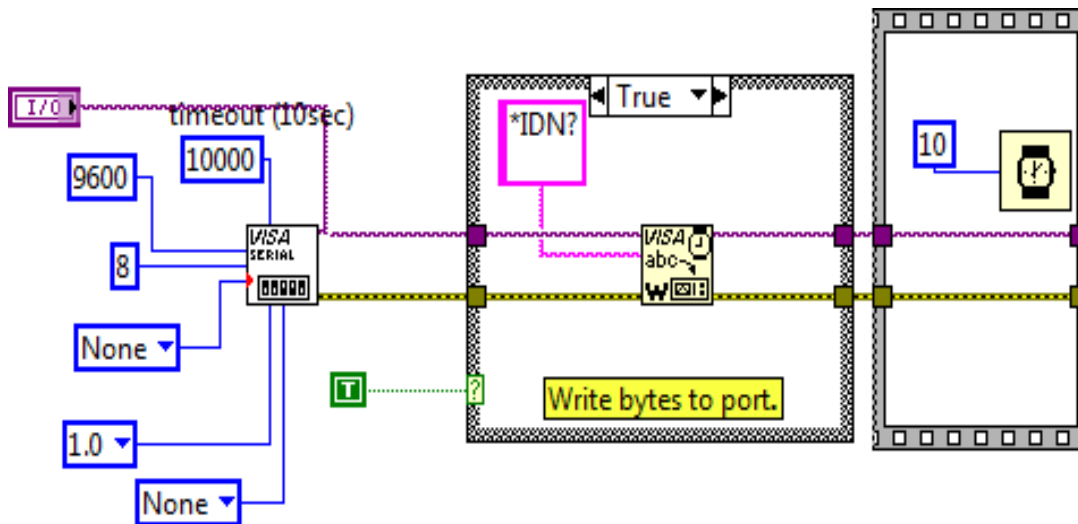
<sup>21</sup> LabVIEW es un entorno de desarrollo altamente productivo que los ingenieros y científicos utilizan para la programación gráfica y la integración de hardware sin precedentes, para diseñar y desplegar rápidamente sistemas de medidas y control.



**Gráfico 70 : Diagrama de Bloques**

**Fuente: El autor**

En este bloque de programación gráfica se crea una interfaz serial de 9600 bps con 8



**Gráfico 71: Codificación del monitoreo**

**Fuente: El autor**



Por último se procesa la señal

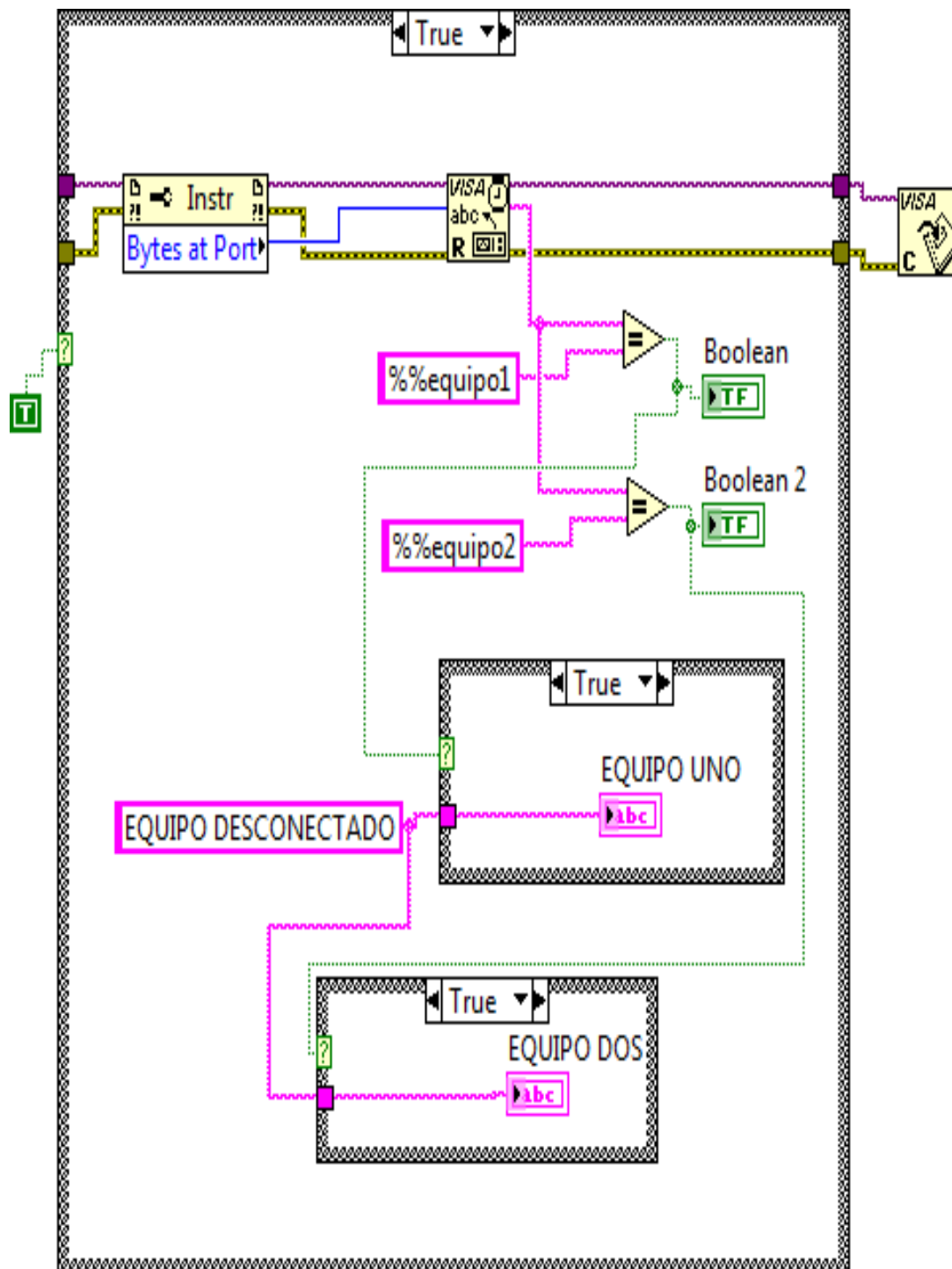


Gráfico 72: Instrucciones del monitoreo

Fuente: El autor

#### 4. CONCLUSIONES

El sistema SCAL, se recomienda para las empresas, los supermercados y en este caso en particular a los laboratorios de las Universidades, en donde existe la necesidad de salvaguardar los activos fijos, así también controlar el acceso de las personas autorizadas en horarios establecidos.

Cabe señalar que la tecnología RFID, tiene una amplia gama de aplicaciones que a futuro servirán para ir mejorando la idea.

Este sistema puede ser implementado no solamente en el laboratorio de Telemática sino también en las diferentes Facultades de la Universidad Politécnica Salesiana, aun cuando parecería que el proyecto no es escalable porque la tecnología está en constante innovación y por su costo.

La rentabilidad del proyecto se verá reflejada en el costo beneficio, que al invertir en la implementación del sistema, optimizaremos el tiempo, aumentaremos la seguridad en el acceso y el control de las personas mediante el continuo monitoreo, eliminando las pérdidas de los equipos.

Debo indicar que este sistema fue implementado con recursos propios, demostrando la efectividad en las hipótesis y los objetivos desarrollados.

## **5. RECOMENDACIONES**

Se recomienda implementar este prototipo llamado SCAL, que servirá en el control y seguridad de los laboratorios de la Universidad Politécnica Salesiana, así también ayudará eficientemente en el acceso a los mismos, sin necesidad de la intervención de personas no autorizadas.

Se sugiere, mejorar este proyecto con un sistema biométrico y también que se lo acople al sistema de seguridad de la universidad.

Es recomendable que las etiquetas electrónicas no se encuentren en un lugar visible, para evitar que personas desatencivas los desprendan, garantizando así el monitoreo seguro.

Se propone también, que para tener mayor información. Sobre las bondades que ofrece el sistema SCAL, se realice un estudio de factibilidad para la implementación del proyecto.

## 6. BIBLIOGRAFIA

- Alice Y H Tsai, (1990) *Sistemas de Base de Datos: Administración y Uso*: Prentice Hall Hispanoamericana, S.A.
- Cultural S.A.(1999) *Diccionario de Marketing*.
- Chugani Mahesh L., Samant Abhay R., Cerna Michael. (1998). *LabVIEW Signal Processing*: Pearson Education.
- DuBois, Paul. (2013) *MySQL*. (5ª ed.): Addison-Wesley.
- Heinrich, Keil. (1987). *Microcomputadores: Fundamentos, Módulos y conjuntos modulares, Recursos para el desarrollo de sistemas controlados por MC herramientas para el desarrollo*: Siemens.
- Kroenke, David M. (2003) *Procesamiento de Base de Datos: Fundamentos, Diseño e Implementación*.
- Sandhusen L. Richard. (2002) *Mercadotecnia*. (1ª ed.): Compañía Editorial Continental.
- Stanton, Etzel y Walker. (2004) *Fundamentos de Marketing*. (13a ed.): Mc Graw Hill.
- Tojeiro Calaza Germán. (2008). *Proteus: Simulación de circuitos electrónicos y microcontroladores a través de ejemplos*: Marcombo, S.A.
- Usategui, José María Angulo, Martínez Ignacio Angulo. (2003). *Microcontroladores PIC: diseño práctico de aplicaciones: primera parte, el PIC 16F84, lenguajes PBASIC y Ensamblador*. (3ª. Ed.): McGraw-Hill Interamericana.
- Usategui, José María Angulo, Martínez Ignacio Angulo. (2006). *Microcontroladores PIC: diseño práctico de aplicaciones: segunda parte, el PIC 16F87X, PIC18FXXXX*.(2ª. Ed.): McGraw-Hill Interamericana.
- Yan Zhang, Laurence T. Yang, Jiming Chen. (2010). *RFID and Sensor Networks: Architectures, Protocols, Security, and Integrations Wireless Networks and Mobile Communications*: CRC Press.
- Alvarado Sánchez, Jorge Alberto. (2008). *Sistema de Control de Acceso con RFID*. Maestro en Ciencias En la Especialidad de

Ingeniería Eléctrica Opción Computación, Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, México.

- Jaramillo Rodas, Diana. Llor Reyes, Gabriela. (2009). *Sistema de Control Personal*. Escuela Superior Politécnica Del Litoral, Guayaquil, Ecuador.
- Mikrotiks LTD (2000, Noviembre 20). Mikrotik. Recuperado el 3 de marzo del 2003, de [www.mikrotik.com/](http://www.mikrotik.com/).
- Ingeniería electrónica y proyectos PicMicro. Electrónica Estudio Recuperado el 27 de septiembre del 2012 de <http://www.electronicaestudio.com/microcontrolador.htm>.
- El Blog de Nextia Fenix (2010). Nextia Fenix. Recuperado el 19 de diciembre del 2011, de <http://www.blog.nextiafenix.com>.
- Documentos de Scribd (2013). El relé. Recuperado el 19 de agosto del 2010, de <http://es.scribd.com/doc/36142626/rele>.
- R – Luis. (2013, Agosto 07). Sistemas Microcontrolados. Recuperado el 22 de agosto del 2010, de <http://r-luis.xbot.es/pic1/pic01.html>.
- RedRaven (2010, Agosto 07). Mis Primeros Pasos con el PIC 18F4550. Recuperado el 22 de agosto del 2010, de [http://picmania.garcia-cuervo.net/invitados\\_primer18f4550.php](http://picmania.garcia-cuervo.net/invitados_primer18f4550.php).
- MikroElektronika. (1998 – 2013) PIC Microcontrollers. Recuperado el 23 de agosto del 2010, de <http://www.mikroe.com/chapters/v>.
- Tecnología Ltd. de Shenzhen Shanghai. (2011 - 2013) Módulo de transceptor FSK inalámbrico RFM12. Recuperado el 3 de septiembre del 2010, de [http://spanish.gsm-gprs-modem.com/china-rfm12\\_wireless\\_fsk\\_transceiver\\_module-134833.html](http://spanish.gsm-gprs-modem.com/china-rfm12_wireless_fsk_transceiver_module-134833.html).
- The PHP Group. (2001- 2013) PHP. Recuperado el 5 de septiembre del 2010, de <http://www.php.net/>.
- Oracle (2013). Recuperado el 25 de septiembre del 2010, de <https://netbeans.org/>
- The Labcenter Electronics. (2013). Recuperado el 4 de diciembre del 2010, de <http://www.labcenter.com/index.cfm>

- The Apache Software Foundation. (2012). Recuperado el 7 de diciembre del 2010, de <http://httpd.apache.org/>
- Oracle (2013). Recuperado el 11 de diciembre del 2010, de <http://www.oracle.com/technetwork/java/index.html>
- MikroElektronika. (1998 – 2013) PIC Microcontrollers. Recuperado el 23 de agosto del 2010, de <http://www.mikroe.com/add-on-boards/various/rfid-reader/>
- Documentos de Scribd (2013). PIC18F4550. Recuperado el 19 de agosto del 2010, de <http://es.scribd.com/doc/2940003/PIC18F4550>
- MikroElektronika. (1998 – 2013) PIC Microcontrollers. Recuperado el 23 de agosto del 2010, de <http://www.mikroe.com/>
- Ingeniería electrónica y proyectos PicMicro. Electrónica Estudio Recuperado el 27 de septiembre del 2012 de [http://www.electronicaestudio.com/usb\\_rs232.htm](http://www.electronicaestudio.com/usb_rs232.htm)
- Hope Microelectronics co., Ltd (2009). RF Modules. Recuperado el 07 de septiembre del 2010, de [http://svn.clifford.at/metaparts/trunk/datasheets/DS\\_647a2fa8f4e97af8dc853299dc010412.pdf](http://svn.clifford.at/metaparts/trunk/datasheets/DS_647a2fa8f4e97af8dc853299dc010412.pdf)
- Algarreyrochtorr. (2011 abril). Funcionamiento del PIC16f628a. Recuperado el 22 de septiembre del 2010, de <http://proyecto-aula-pic16f628a.blogspot.com/2011/04/pic16f628a.html>
- IDETEC.CIA.LTDA. Módulo USB. Recuperado el 10 de octubre del 2010, de <http://www.ideastechnology.com>.
- Sual Labs (2013). Resistencia de carbón 10K Ohms, de 1/4 Watt. Recuperado el 15 de octubre del 2010, de [http://www.suallabs.com/index.php?route=product/product&product\\_id=101](http://www.suallabs.com/index.php?route=product/product&product_id=101)
- Universidad Politécnica Salesiana. Recuperado el 10 de enero del 2012, de <http://www.ups.edu.ec/>

- Parallax Inc. (2013). Recuperado el 28 de noviembre del 2010, de <http://www.parallax.com>

# ANEXOS

## **Modelo de entrevista realizada**

Cree usted ¿qué existe inseguridad dentro de los laboratorios del Campus?

Si



No

Cree usted ¿qué pierde mucho tiempo en el acceso a los distintos laboratorios de la Universidad al momento de impartir su clase?

Si

No

Cree usted ¿qué es necesario la implementación del sistema de control de acceso y monitoreo en los laboratorios de la UNIVERSIDAD POLITECNICA SALESIANA?

Si

No

Cree usted ¿qué el sistema facilitaría el control del ingreso y acceso a los laboratorios?

Si

No

Cree usted ¿qué se reduciría el índice de pérdidas de los equipos o materiales de los laboratorios mediante la implementación del sistema?

Si

No

---

ENTREVISTADOR

---

DOCENTE DE LA UPS