



UNIVERSIDAD JOSÉ ANTONIO PÁEZ

SISTEMA SCADA CON SENSORES DE MOVIMIENTO, TEMPERATURA Y DE HUMO PARA LA SUPERVISIÓN DE LAS AULAS DE LA UNIVERSIDAD JOSÉ ANTONIO PÁEZ

Autor: Valente Victor
C.I.: 20.792.785

Tutor: MSc. Jetro Ramón López

Urb. Yuma II, calle N° 3. Municipio San Diego
Teléfono: (0241) 8714240 (master)



REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD JOSÉ ANTONIO PÁEZ
FACULTAD DE INGENIERÍA
ESCUELA DE COMPUTACIÓN

**SISTEMA SCADA CON SENSORES DE MOVIMIENTO, TEMPERATURA
Y DE HUMO PARA LA SUPERVISIÓN DE LAS AULAS DE LA
UNIVERSIDAD JOSÉ ANTONIO PÁEZ**

**Trabajo de grado presentado como requisito para optar el título de
INGENIERO DE COMPUTACIÓN**

Autor: Valente Victor
C.I.: 20.792.785
Tutor: MSc. Jetro Ramón López

San Diego, 2019



REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD JOSÉ ANTONIO PÁEZ
FACULTAD DE INGENIERÍA
ESCUELA DE COMPUTACIÓN

ACEPTACIÓN DEL TUTOR

Quien suscribe, MSc. Jetro Ramón López, portador de la cédula de identidad N° V-8.779.723, en mi carácter de tutor del trabajo de grado presentado por el ciudadano Victor Valente; portador de la cédula de identidad N° V-20.792.785, titulado **“SISTEMA SCADA CON SENSORES DE MOVIMIENTO, TEMPERATURA Y DE HUMO PARA LA SUPERVISIÓN DE LAS AULAS DE LA UNIVERSIDAD JOSÉ ANTONIO PÁEZ”**, presentado como requisito parcial para optar al título de ingeniero en computación, considero que dicho trabajo reúne los requisitos y méritos suficientes para ser sometido a la presentación pública y evaluación por parte de jurado examinador que se designe.

MSc. Jetro Ramón López

C.I: 8.779.723



FI-C -002-2019-2CE

Valencia, 19 de Julio de 2019

Ciudadanos:
Victor Valente
C.I: 20.792.785
Presente-

Cumplo con informarle que la Comisión de Trabajo de Grado y Pasantías de la Facultad de Ingeniería en su reunión N° 01-2019 de fecha 19-07-2019 aprobó el proyecto de trabajo de grado titulado **SISTEMA SCADA CON SENSORES DE MOVIMIENTO, TEMPERATURA Y DE HUMO PARA LA SUPERVISIÓN DE LAS AULAS DE LA UNIVERSIDAD JOSÉ ANTONIO PAÉZ**. Presentado por usted como requisitos para optar al título de Ingeniero en Computación.

Se ratifica la designación del Ing. Jetro López C.I:8.779.723 y la Ing. Alicia De Pizzela C.I: 4.598.880 como Tutores Académicos y Metodológicos que los asesoraran en el desarrollo de este proyecto.

Atentamente,

Prof. Luis Lira

Decano de la Facultad de Ingeniería



e.e. Coordinación de Pasantías y Trabajo de Grado (1).

Llc

AGRADECIMIENTOS

A Dios, por su amor, bendiciones y por darme la oportunidad de lograr una de las metas más importantes en mi vida, por darme salud cada día de mi vida, por llenarme de sabiduría y entendimiento a lo largo de mis estudios.

A mis padres, por todo lo que han hecho por mi durante toda mi vida, por su amor, humildad, por su ejemplo de fortaleza y paciencia; juntos hicimos que este logro se hiciera realidad, gracias a los dos por guiarme a ser lo que soy hoy.

A la empresa Scantech que gracias a la paciencia y al equipo de trabajos me sirvieron de apoyo fundamental para realizar este proyecto de grado.

A mis profesores, por enseñarme todo lo que se, por ser mis amigos y ayudarme en cada dificultad.

A mi tutor Jetro López, quien aportó sus conocimientos, gracias por su enseñanza, consejos, paciencia y por su tiempo.

A mi novia por tener una infinita paciencia todos estos meses y por acompañarme en este largo camino para llegar a donde estoy hoy.

A mis amigos por brindar compañía y apoyo en los momentos más difíciles hasta el día de hoy.

A la médico internista Tatiana Maure porque gracias a ella es que tengo salud y puedo seguir construyendo más logros en la vida.

Valente, Victor

DEDICATORIA

A Dios, que me dio la fuerza necesaria para seguir adelante y llevar a cabo todas mis metas, siendo mi guía a lo largo de mi carrera.

A la Universidad José Antonio Páez, por darnos la oportunidad de una formación profesional.

A mis padres, Valente Fernando y Gouveia Maria, quienes me dieron la educación necesaria para ser lo que soy hoy día, por siempre estar en los momentos más difíciles, sus palabras me llenaron de fuerzas para seguir adelante y cumplir mi meta.

A mi familia, novia y amigos, quienes fueron un pilar en diferentes momentos durante toda mi carrera.

A mi tutor de tesis Profesor Jetro López, quien fue guía para llevar a cabo esta tesis.

Valente, Victor

ÍNDICE

	Pág.
CONTENIDO	
RESUMEN	ix
INTRODUCCIÓN	1
INTRODUCCIÓN	1
EL PROBLEMA	3
1.1 Planteamiento del Problema.....	3
1.2 Formulación del Problema	4
1.3 Objetivos de la investigación.....	5
1.3.1 Objetivo General.....	5
1.3.1 Objetivos Específicos	5
1.4 Justificación.....	5
1.5 Alcance de la Investigación.....	6
1.6 Limitaciones	6
MARCO TEÓRICOS	
2.1 Antecedentes.....	8
2.2 Bases teóricas	9
2.2.1 Automatización.....	9
2.2.2 Sistema SCADA	10
2.2.3 Entorno del Sistema SCADA	13
2.2.4 Control Lógico Programable (PLC)	14
2.2.5 Detector de Humo.....	17
2.2.6 Detector de Movimiento	18

2.2.6.1 Tipos de sensores de movimiento.....	19
2.3 Bases Legales	20
2.3.1 Constitución de la República Bolivariana de Venezuela.....	20
2.3.2 Ley Orgánica De Ciencia, Tecnología e Innovación (LOCTI)	21
2.4 Definición de términos	22
MARCO METODOLÓGICO.....	24
3.1. Tipo de Investigación	24
3.2 Diseño de la Investigación.....	25
3.3 Nivel de la Investigación	25
3.4 Población y Muestra.....	26
3.4.1. Población	26
3.4.2. Muestra	26
3.5 Técnicas e Instrumentos de recolección de datos.....	26
3.5.1. Técnicas de recolección de datos.....	26
3.5.2. Instrumentos de recolección de datos	27
3.6 Fases de la Investigación.....	28
RESULTADOS.....	30
4.1 Fase I: Análisis de los puntos de mayor riesgo en la universidad, examinando los niveles superiores del edificio 4 para realizar los requerimientos funcionales y no funcionales del sistema.....	30
4.1.1 Estudio de los Niveles Superiores	30
4.1.2 Capturar en imágenes los puntos a atacar para demostrar de manera más específica los puntos ciegos.....	32
4.1.3 Requerimientos del Sistema	37

4.2 Fase II: Diseño de la interfaz del sistema SCADA utilizando las herramientas computacionales	38
4.2.1 Actualizar el entorno de desarrollo de visual studio para diseñar utilizando el lenguaje de C#	38
4.2.2 Obtención de las DLL correspondientes al para desarrollar el diseño de la interfaz	44
4.2.3 Diagramas correspondientes para realizar el sistema	59
4.2.4 Diseño del wirframe de la interfaz donde el usuario se sienta cómodo y todo el aprendizaje sea entendible de la manera más rápida.	61
4.3 Fase III: “Desarrollo del sistema SCADA para la automatización de los sensores y obtener toda la información en tiempo real”	62
4.3.1 Investigación mediante la web, tomando referencias e información acerca de los tipos Software disponible.	62
4.3.2 Tipo de software a utilizar teniendo en cuenta características favorables, que a su vez concuerdan con la idea principal del automatizado.	71
4.3.3 Creación de los sensores a través del Programa Proteus 8.0 y realizar la conexión con el sistema Scada	77
4.3.4. Desarrollo del sistema SCADA	81
CONCLUSIONES.....	89
RECOMENDACIONES.....	91
REFERENCIAS BIBLIOGRÁFICAS.....	94

ÍNDICE DE FIGURAS

FIGURAS	Pág.
1- Interfaz del sistema Scada para el control de Procesos.....	20
2- Diseño del sensor de movimiento de amplificador para PIR con 4558.....	30

3- 5to Piso del edificio de estudios múltiples.....	30
4- 4to Piso del edificio de estudios múltiples.....	31
5- Laboratorio de Computación del edificio de estudios múltiples.....	32
6- Laboratorio de Electrónica del edificio de estudios múltiples.....	33
7- Salón de clases del edificio de estudios múltiples nivel 4.....	34
8 - Banco Nacional de Crédito del edificio de estudios múltiples.....	35
9- Entorno de desarrollo Visual Studio 2019.....	39
10- Interfaz de SQL Server Management Studio.....	41
11- Interfaz del Visual Studio Code.....	43
12- Medidor de la librería de AGauge.dll.....	46
15- Dispositivo Electrónico Arduino.....	58
16- Lenguaje Arduino.....	60
17- Interfaz de Proteus 8 Profesional.....	63
18- Interfaz de VirtualBreadboard (VBB).....	65
19- Interfaz de Virtual Serial Port.....	67
22- Sensor de Temperatura LM35 conectado a un arduino.....	76
23- Interfaz Scada Medidor de Temperatura.....	77
24- Sensor de Humo (Flame Sensor) conectado a un Arduino.....	78



REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD JOSÉ ANTONIO PÁEZ
FACULTAD DE INGENIERÍA
ESCUELA DE COMPUTACIÓN

SISTEMA SCADA CON SENSORES DE MOVIMIENTO, TEMPERATURA Y DE HUMO PARA LA SUPERVISIÓN DE LAS AULAS DE LA UNIVERSIDAD JOSÉ ANTONIO PÁEZ

Autor: Victor Valente

Tutor: MSc. Jetro Ramón López.

Fecha: Agosto 2019

El presente proyecto de investigación, surge de la problemática de la falta de sistemas automatizados de seguridad dentro de la Universidad José Antonio Páez, la falta de personal humano dentro de la institución, aunado a las limitaciones de recursos tecnológicos que presenta, ha desarrollado una limitante que evita garantizar el bienestar de la población estudiantil. En consecuencia el presente trabajo lleva como objetivo desarrollar un sistema Scada, con sensores de movimiento, temperatura y humo para la supervisión de las aulas, realizado en el edificio de estudios múltiples. El sistema detecta cualquier irregularidad con respecto a: movimientos, temperaturas y sensores de humo, todo esto se realiza dentro y fuera de los horarios de clase. De esta manera, atacar de forma más efectiva las situaciones que puedan presentarse.

Descriptores: Sensores, Control, Sistema, Adquisición de datos

INTRODUCCIÓN

En la investigación realizada, se presenta un sistema Scada con sensores de movimiento temperatura y humo para la universidad José Antonio Páez, junto con el análisis de la información que este comprende para la elaboración del mismo.

Actualmente la universidad no cuenta con sistemas automatizados para un mejor control en cada una de sus áreas, y no cuentan con suficiente personal para cubrir cada espacio en las horas que esta institución mantiene las puertas abierta lo que ocasiona una falla en la parte de la seguridad y permite que se realicen acontecimientos no deseados dentro del campus universitario.

Los sistemas Scada no cuentan con una interfaz amigable y muchas son confusas para el usuario y terminan generando más problemas que soluciones, la interfaz de este sistema pretende ser más amigable y atractiva para cualquier personal que sea encargado de la parte de este sistema.

A su vez para realizar este sistema, se inicia un estudio de las áreas más vulnerables en la parte de la seguridad de la universidad, para luego realizar el sistema basado en cubrir esas áreas para crear un ambiente de mayor seguridad, una vez realizado el estudio se comienza con el diseño de la interfaz del sistema, buscando que sea sencillo, descriptivo y efectivo para que los usuarios se sientan cómodos y no creen problemas por complejidad del sistema, de esta forma se plantea lograr que de una manera automatizada la Universidad José Antonio Páez Ubicada en Venezuela Carabobo Valencia supervise mediante sensores, movimiento, temperatura y humo, todos los laboratorios del edificio de estudios múltiples.

De tal manera, la presente investigación se encuentra estructurada de la siguiente forma:

- Ø **Capítulo I El Problema:** Planteamiento del problema, Formulación del Problema, Objetivo General, Objetivos Específicos, Justificación del Problema, Alcance, Limitaciones

- Ø **Capítulo II Marco Teórico:** Antecedentes de la Investigación, Bases Teóricas, Bases Legales, Definición de Término
- Ø **Capítulo III Marco Metodológico:** Tipo de Investigación, Nivel de la Investigación, Diseño de la Investigación, Población y Muestra, Fases de la Investigación.
- Ø **Capítulo IV Resultado:** FaseI: Análisis de los puntos de mayor riesgo en la universidad, examinando los niveles superiores del edificio de estudios múltiples para realizar los requerimientos funcionales y no funcionales del sistema, FaseII: Diseño de la interfaz del sistema SCADA utilizando las herramientas computacionales, FaseIII: Desarrollo del sistema SCADA para la automatización de los sensores y obtener toda la información en tiempo real

CAPÍTULO I

EL PROBLEMA

1.1 Planteamiento del Problema

Uno de los mayores problemas que se presenta en la actualidad es la falta de adaptación al internet de las cosas, ya que en muchas localidades aún siguen trabajando con sistemas antiguos y análogos que se conecta a equipos como un DVR (grabador de vídeo digital) que almacena la información sin procesar mayores datos o en muchos casos sin notificar en tiempo real lo que sucede, lo que ocasiona que no se tenga un control específico en los procesos de seguridad, muchas veces la falta de conocimiento o el miedo no permite que se desarrollen más edificio automatizado, lo que lleva a no mantener una ubicación con una mayor supervisión para manejar la seguridad y el ambiente interno de un área determinada de una forma más precisa.

En la República Bolivariana De Venezuela en la actualidad se vive un problema socioeconómico, lo que impide que se pueda invertir en la adquisición de equipos de seguridad y las empresas que te ofrecen estos productos no poseen los sistema que te permita supervisar y adquirir la información en tiempo real, tales como son los movimientos o los cambios de temperatura que ocurran en un área determinada, esto por la falta de recursos ya sea productos electrónicos o en la dificultad de acceso a la información lo que ocasiona que muchos edificios no posean lo necesario para el control de su seguridad y la temperatura, ocasionando un atraso en la vigilancia y una facilidad para cualquier individuo realizar algún acto delictivo.

En la Universidad José Antonio Páez ubicada en La Esmeralda San Diego Edo. Carabobo, no poseen un sistema de seguridad con sensores de movimiento y que pueda emitir un mensaje en tiempo real, esto se debe a los altos costo de los equipos y a la falta de confiabilidad que garantiza las empresas nacionales, esta falta de métodos de vigilancia provoca algunas personas tenga el deseo de realizar actos indebidos o

delictivos, el edificio de estudios múltiples de la institución posee una gran dimensión lo que hace más difícil al personal de seguridad revisar todas las áreas en todo momento y las acciones antes mencionadas pueden ser fácilmente realizadas por la falta de vigilancia en el edificio, también puede ocurrir accidentes laborales que no sean atendido a tiempo.

Como ocurrió en un caso que alguien se quedó atrapado en un salón y no hubo forma de enterarse de lo que aconteció y si no se tiene los equipos electrónicos para realizar una llamada de emergencia, el usuario podía quedar atrapado hasta que alguien usara el aula, por otra parte no poseen un control de temperatura en los salones, ya que la falta de mantenimiento en los aires acondicionados no permite que detecten a que grados Celsius está el ambiente, generando que estos electrodoméstico trabajen más en algunas aulas donde no se detecte tanto movimiento durante el día y con menos sensación de calor, desperdiciando los recursos de la institución elevando la merma por la conservación de los sistemas no usados.

1.2 Formulación del Problema

Tomando en cuenta la problemática anteriormente puntualizada, se plantea la creación de un sistema Scada (SUPERVISIÓN, CONTROL Y ADQUISICIÓN DE DATOS) con sensores de movimiento, temperatura y de humo, para la adquisición de la información en tiempo real lo que permite ver el movimiento en las aulas, saber que salones son los que más poseen movimientos durante el semestre, a que temperatura los alumnos y los profesores se encuentran utilizando las instalaciones, De acuerdo a lo anteriormente expuesto se plantea la siguiente interrogante ¿Cómo se puede monitorear las aulas de la Universidad José Antonio Páez mediante sensores, movimiento, temperatura y humo?

1.3 Objetivos de la investigación

1.3.1 Objetivo General

Desarrollar un sistema Scada (SUPERVISIÓN, CONTROL Y ADQUISICIÓN DE DATOS) con sensores de movimiento, temperatura y humo, utilizando equipos electrónicos como los PLC, videos y almacenamiento de los mismos para el monitoreo en las Aulas de la Universidad José Antonio Páez

1.3.1 Objetivos Específicos

- Analizar los puntos de mayor riesgo en la universidad, examinando los niveles superiores del edificio 4 para poder realizar los requerimientos funcionales y no funcionales del sistema.
- Diseñar la interfaz del sistema SCADA utilizando las herramientas computacionales.
- Desarrollar el sistema SCADA para la automatización de los sensores y obtener toda la información en tiempo real

1.4 Justificación

El desarrollo del sistema SCADA en la Universidad José Antonio Páez, permite el monitoreo de las aulas a través de los sensores que se plantean desarrollar permitiendo la supervisión de los movimientos dentro de los salones de clase así saber cuáles son más frecuentados durante el semestre, para ayudar a la institución a saber qué espacio no están siendo utilizado y dedicarle el mantenimiento y las condiciones a otras partes, también permite evitar que gente con intenciones maliciosas tenga la facilidad de realizar estos tipos de actos dentro de la facultad.

También permite el monitoreo de las temperaturas para mantener al personal y a los estudiantes en un ambiente laboral y educativo reconfortante, y saber cuáles equipos deben de ser utilizados durante el día para no desperdiciar los recursos de otros, generando un bajo consumo eléctrico.

1.5 Alcance de la Investigación

La aplicación Tiene como objetivo la parte de la supervisión de los espacios de la universidad José Antonio Páez, desde la seguridad hasta el monitoreo de la temperatura, su enfoque va dirigido principalmente al edificio 4 de las instalaciones, ya que por sus dimensiones el personal que se encarga de la seguridad de la Páez no puede abarcar la vigilancia total del edificio, el piso 4 y 5 de la universidad son los más vulnerables ya que los encargados de la seguridad se mantiene en los niveles inferiores, después de todo se encuentran las clínicas y es un área que requiere de mayor control y mayor personal de vigilancia.

1.6 Limitaciones

Todos los casos de estudio no poseen las mismas limitaciones, cada una de estas prestaran diferentes particularidades, es el tiempo un factor limitante al desarrollo del trabajo, puesto que este no pudo haber sido suficiente para la mayor profundización en el periodo evaluado. Así mismo, puede haber limitaciones en cuanto a los recursos especialmente financieros para poder desarrollar una investigación más profunda.

CAPÍTULO II

MARCO TEÓRICO

A continuación, se presentan varios proyectos o trabajos integradores efectuados en los últimos años, que tomaron aportes valiosos para la investigación que pueda brindar cada uno de ellos.

2.1 Antecedentes

Según Hernández, Sampieri (2008) señala que un Marco Teórico es un compendio escrito de artículos, libros y otros documentos que describen el estado pasado y actual de conocimiento sobre el problema de estudio. Nos ayuda a comentar como nuestra investigación agrega valor a la literatura existente.

Castillo, L y Fuenmayor, R (2019) en su proyecto de investigación titulado **“Sistema scada (supervisión, control y adquisición de datos) para el control automático del encendido de los aires acondicionados de la universidad José Antonio Páez”**, presentado ante la Universidad José Antonio Páez ubicada en Venezuela-Carabobo Municipio San Diego, para optar por el título en Ingeniería Electrónica e Ingeniería en Computación. Este proyecto tiene como objetivo automatizar el encendido de los aires acondicionados solo de las aulas de clase de los edificios 1, 4 y el sótano de arquitectura. Actualmente, el proceso de encendido/apagado de los aires acondicionados de las aulas de la universidad José Antonio Páez, no está automatizado, son encendidos al comenzar la mañana y son apagados al finalizar el día, estando encendidos más de 12 horas diarias. Es por esto que se propone la implementación de un sistema SCADA para controlar el encendido automático de los aires acondicionados. Este proyecto de investigación tiene relación con este trabajo porque compartimos el tipo de investigación, nivel y diseño

Igualmente Ferretti C., Ángel L. (2012) realizaron una investigación que lleva por título “**Actualización del sistema Scada del manipulador de tuberías del taladro de perforación HH-200 de la empresa petreven**”, presentado ante la Universidad Central de Venezuela, ubicada en Venezuela-Caracas Municipio Libertador, para optar al título de Ingeniero Electricista. Este proyecto tiene como objetivo Actualizar el sistema SCADA del manipulador de tuberías del taladro de perforación HH-200 de la empresa PETREVEN en el Edo. Monagas.

Este proyecto surgió ante un problema de comunicación del software RsView de la empresa Rockwell Automation, que ocasionaba pérdidas esporádicas de la comunicación entre el controlador lógico programable (PLC) y la computadora de visualización de proceso. Ante esta situación, aunado a que no se poseía el CD de instalación del software RsView para su reinstalación y que el sistema sobre el que estaba instalado, Windows NT, es muy antiguo, lo que dificultaba actualizaciones y compatibilidad con nueva tecnología, se buscó migrar el sistema a otro paquete de software compatible con el sistema. Esta investigación tiene relación con el proyecto que se plantea, porque es una actualización de un sistema ya propuesto para la universidad con las mejoras necesarias.

Por ultimo Rojas, X y Correa, G (2015) en su proyecto de investigación que se titula “**Diseño e implementación de un sistema de control y supervisión HMI para maquina barnizadora de la empresa el telégrafo**”, presentado ante la Universidad Politécnica Salesiana ubicada en Ecuador-Quito para optar por el título de ingeniería en electrónica, este proyecto tiene como objetivo, diseñar e implementar un sistema de control y supervisión HMI para maquina barnizadora de la empresa EL TELÉGRAFO, en la empresa se cuenta con una maquina industrial de barnizado y secado UV que no está operativa, pero en buen estado del sistema mecánico. Está máquina tiene un sistema de control bastante obsoleto y manuales técnicos deficientes lo que provoca dificultad en la revisión del sistema eléctrico. Por eso se propuso la creación de una interfaz HMI (KTP-600), y un controlador lógico PLC, este proyecto tiene relación

con esta propuesta ya se plantea realizar una interfaz y conectarla a un PLC virtual para demostrar la funcionalidad del sistema.

2.2 Bases teóricas

2.2.1 Automatización

El concepto de automatización según Gutiérrez, J.A.(1994) corresponde a la necesidad de minimizar la intervención humana en los procesos de gobierno directo en la producción, vale decir, ahorrar esfuerzo laboral

La automatización es un sistema donde se transfieren tareas de producción, realizadas habitualmente por operadores humanos a un conjunto de elementos tecnológicos. Un sistema automatizado consta de dos partes principales:

- **Parte de Mando:** suele ser un autómata programable (tecnología programada), aunque hasta hace bien poco se utilizaban relés electromagnéticos, tarjetas electrónicas o módulos lógicos neumáticos (tecnología cableada). En un sistema de fabricación automatizado el autómata programable está en el centro del sistema. Este debe ser capaz de comunicarse con todos los constituyentes de sistema automatizado.
- **Parte Operativa:** es la parte que actúa directamente sobre la máquina. Son los elementos que hacen que la máquina se mueva y realice la operación deseada. Los elementos que forman la parte operativa son los accionadores de las máquinas como motores, cilindros, compresores y los captadores como fotodiodos, finales de carrera.

2.2.1.1 Tipos de Automatización

Operación riesgosa.

Baja calidad.

Sólo para procesos simples.

- **Control semiautomático:** se emplea principalmente para facilitar las maniobras de mando y dar flexibilidad a las maniobras de control de aquellas instalaciones en las que el control manual no es posible. Este tipo de control requiere un operador que inicie cualquier cambio en la posición o condición de funcionamiento de la máquina. En la siguiente figura se muestra un control semiautomático.
- **Control Automático:** este control se basa en la realimentación o feedback (medición tomada desde el proceso que entrega información del estado actual de la variable que se desea controlar) cuya característica especial es la de mantener al controlador central informado del estado de las variables para generar acciones correctivas cuando así sea necesario. Este mismo principio se aplica en campos tan diversos como el control de procesos químicos, control de hornos en la fabricación del acero, control de máquinas herramientas, control de variables a nivel médico e incluso en el control de trayectoria de un proyectil militar. El control posee las siguientes características:
 - Intervención humana moderada.
 - Poca capacidad de adaptación
 - Lazos locales de control.
 - Alto mantenimiento.
 - Poca capacidad de supervisión central.

2.2.2 Sistema SCADA

Se da el nombre de SCADA a cualquier software que permita el acceso a datos remotos de un proceso y permita, utilizando las herramientas de comunicación necesarias en cada caso, el control del mismo:

No se trata de un sistema de control, sino de una utilidad software de monitoreo o supervisión, que realiza la tarea de interface entre los niveles de control (PLC) y los de gestión, a un nivel superior. (ver figura 1).

Sistema Scada



Figura I: Interfaz del sistema Scada para el control de Procesos

Fuente: www.ingetes.com.co/sistema-scada-automatizacion-de-procesos/

2.2.2.1 Objetivos del sistema SCADA

Los sistemas SCADAS se conciben principalmente como una herramienta de supervisión y manado. Entre sus objetivos podemos destacar:

- Economía
- Accesibilidad
- Mantenimiento
- Ergonomía
- Gestión
- Flexibilidad
- Conectividad

El sistema a controlar aparece ante el usuario bajo un número más o menos elevado de pantallas con mayor o menor información. Se pueden encontrar planos, fotografías, esquemas eléctricos, gráficos de tendencia, etc.

- Flexibilidad
- Conectividad

2.2.2.2 Prestaciones del Sistema SCADA

El paquete SCADA, en su vertiente de herramientas de interfaces Hombre-máquina, comprende toda una serie de funciones y utilidades encaminadas a establecer una comunicación, lo más clara posible, entre el proceso y el operador:

El paquete SCADA, en su vertiente de herramientas de interfaces Hombre-máquina, comprende toda una serie de funciones y utilidades encaminadas a establecer una comunicación, lo más clara posible, entre el proceso y el operador:

- **La monitorización:** Representación de los datos en tiempo real a los operadores de la planta o proceso.
- **La supervisión:** supervisión, mando y adquisición de datos de un proceso y herramientas de gestión para la toma de decisiones.
- **La adquisición de datos de los procesos en observación:** en un sistema se puede observar mediante herramientas registradoras y obtener así un valor medio guardando los valores obtenidos y evaluándose a posteridad.
- **La visualización de los estados de la señales del sistema:** Reconocimiento de eventos excepcionales acaecidos en la planta y su inmediata puesta en conocimiento a los operarios para efectuar las acciones correctivas pertinentes. Además, los paneles de alarma pueden exigir alguna acción de reconocimiento por parte del operador, de forma que queden registradas las incidencias.
- **El Mando:** posibilidad de que los operadores puedan cambiar consignas u otros datos claves del proceso directamente desde el ordenador.
- **Garantizar la seguridad de los datos:** tanto el envío como la recepción de datos deben de estar suficientemente protegidos de influencias no deseadas, intencionadas o no.

- **Garantizar la seguridad en los accesos:** registrando zonas de programa comprometidas a usuarios no autorizados, registrando todos los accesos y acciones llevadas a cabo por cualquier operador.

Cuando se habla de un sistema SCADA no hay que olvidar que hay algo más que las pantallas que nos informan cómo van los procesos en nuestra instalación. Tras estas se encuentran multitud de elementos de regulación y control, sistemas de comunicaciones y múltiples utilidades de software que pretenden que el sistema funcione de forma eficiente y segura.

2.2.3 Entorno del Sistema SCADA

La automatización de sistemas, desde el estado inicial de aislamiento productivo, ha pasado a formar parte del ámbito del corporativo y se engloba dentro del paquete empresarial con la finalidad de optimizar la productividad y la mejor calidad el esquema que representa los flujos de información dentro de la empresa y representa cómo se realiza la integración de todos los niveles es similar a la conocida pirámide de automatización CIM.

Presente la estructura corporativa dividida en tres niveles, que engloban las funciones principales que se desarrollan en cada uno de ellos:

- **ERP (Enterprise Resource planning):** engloba la parte de gestión ,finanzas ,compras ,ventas, logística
- **MES (Manufacturing Execution System):** comprende la gestión de calidad, documentación, gestión de producción, mantenimiento y optimización.
- **CONTROL:** engloba toda la parte de automatización y control de procesos

Estos niveles engloban los diferentes flujos de información que se dan entre los elementos de cada uno de ellos y el intercambio de información que se da entre los diferentes niveles.

- La finalidad de este organigrama es disponer de la máxima información posible sobre el estado operativo global de la empresa para planificar las acciones de producción.

- Conocer la demanda prevista para planificar la producción a corto, medio y largo plazo y coordinar compras y logística (ERP).
- Conocer las existencias de material disponible para aplicar en el proceso productivo y decidir si hay que planificar nuevas compras (MES) Gestión de almacén
- Coordinar los ciclos de mantenimiento preventivo para conocer la disponibilidad de maquinaria y la capacidad operativa durante el tiempo de producción previsto(MES, mantenimiento)
- Conocer el control operativo de planta (CONTROL, SCADA)

2.2.4 Control Lógico Programable (PLC)

PLC traducido al español significa “Control Lógico Programable, estos dispositivos están diseñados para automatizar procesos, un PLC moderno es básicamente un control secuencial de Relevadores, si lo queremos ver desde el punto de vista más simple, es una tarjeta la cual cuenta con 8 relevadores (los comunes, hay otros de mayor capacidad o exenciones), estos relevadores no son como los comunes y son llamados de estado sólido, por lo cual no hay ese sonido que caracteriza a un relevador común, esta tarjeta puede ser programada para que siga una secuencia, posee dos tipos de conexiones :

- TCP/IP
- Puerto Serial

2.2.4.1 Características de los PLC.

Poseen memoria volátil y no volátil: tanto el programa de aplicación escrito por el usuario como los datos internos del PLC, normalmente es guardado en una RAM (memoria volátil), lo que le permite tener un acceso más veloz a las instrucciones de programa y a los datos internos de registros, contadores, temporizadores, bits internos, etc. También, una vez que se ha depurado el programa de aplicación, los PLC permiten la opción de salvaguardar el programa en memorias tipo EEPROM (no volátiles) para

así recuperar el mismo en caso de un corte muy prolongado de energía que ocasiona una pérdida de datos de la RAM.

Capacidad modular de entradas-salidas: esto permite la combinación de distintos niveles y tipos de señal de entrada, así como también el manejo de salidas para distintos tipos de carga. Igualmente si la aplicación crece, y se requiere mayor número de entradas / salidas, casi sin ningún problema los PLC pueden adecuarse al nuevo requerimiento.

Auto diagnóstico: el PLC monitorea el funcionamiento de su CPU, Memoria y circuito de interfaces de entrada y de salida, e igualmente funcionamiento del promedio de led en su cara frontal el estado respectivo. Obviamente esta capacidad es de gran utilidad para efectos de mantenimiento y corrección de fallas.

Programación de la lógica de control: esto permite la fácil adaptación a los cambios en la lógica de operación de las máquinas y procesos.

Capacidad para generar reportes y comunicarse con otros sistemas: con esta facilidad se pueden integrar interfaces de explotación Hombre-Máquina, sacándole al sistema mayor cantidad de información. Igualmente los PLC pueden participar en redes de datos comunicándose con otros PLC para formar sistemas de control distribuidos, o integrándose a las redes administrativas de la producción.

2.2.4.2 Comunicación entre PLC e Interfaz Humano Maquina (HMI)

A lo largo de los años, hemos visto el gran auge tecnológico, esto también estuvo presente en las industrias, lo podemos observar en la evolución de los dispositivos utilizados para llevar el control en ellas y también en el tipo de comunicación que utilizan para lograr esto, a continuación se explicaran los tres protocolos más utilizados en la industria para hacer la comunicación entre los controladores lógicos programables (PLC) y las interfaces hombre maquina (HMI) en la industria.

Comunicación Interfaz Multipunto (MPI)

La comunicación tipo MPI (Multi Point Interface) es la interface de comunicación más sencilla que existe entre los autómatas programables, y la más

económica, ya que no requiere a diferencia de las otras comunicaciones una tarjeta de comunicación adicional. La comunicación MPI se realiza a través del puerto integrado del PLC. Si queremos realizar la conexión entre un computador con un software de programación y el PLC, se deberá realizar a través del puerto MPI pero 19 por medio de un adaptador. Este adaptador convertirá la señal RS-232 al interface MPI. Con este medio de comunicación se puede transferir o acceder a cualquier zona del mapa de memoria del PLC con el que deseamos comunicarnos, y tan sólo es necesario indicar dónde queremos que se nos dejen los datos leídos o cuales es la fuente de datos a enviar.

Estándar de comunicaciones PROFIBUS-PA

Es una red especialmente diseñada para la automatización de procesos, además, permite que sensores y actuadores puedan ser conectados a un bus común en áreas especiales de seguridad. Básicamente, es la aplicación de profibus-DP 20 compatible en comunicación con una tecnología que permite aplicaciones para la automatización de procesos recintos expuesto al peligro de explosiones.

Protocolo de comunicación MODBUS

Modbus es un protocolo de comunicación industrial, que normalmente podemos encontrar en fábricas y edificios inteligentes. El objetivo del protocolo es la transmisión de información entre distintos equipos electrónicos conectados a un mismo bus. Muchos dispositivos de campo lo usan para poder comunicarse con PLC's y SCADA's. Las redes Modbus utilizan una arquitectura maestro – esclavo. El maestro inicia las comunicaciones preguntando datos a un esclavo que le responde siempre en función de la pregunta hecha por el maestro. Cada dispositivo de una red posee una dirección única. Modbus es un protocolo de aplicación, lo que significa que puede implementarse sobre diferentes capas físicas. Es por ello que podemos encontrar versiones TCP/IP, o también serie como RTU y ASCII. Por lo tanto, podremos usar Modbus tanto con cables de red como con cables serie, siempre que los dispositivos con los que interactuemos estén preparados para ello.

Comunicación Profinet

Es un sistema que ofrece todo el potencial que ofrece Ethernet, pero utiliza medidas de seguridad, incluidas las de control de acceso y autenticación, seguridad en la conectividad y administración, a fin de asegurar y garantizar la confidencialidad e integridad de la red y ofrecer datos libres de interferencias. La comunicación Ethernet es un estándar de redes de computadores de área local, o sea de corta extensión. La diferencia entre las comunicaciones anteriormente nombradas radica en la velocidad de transmisión, por una rapidez de transmisión mil veces mayor la opción escogida es Ethernet.

2.2.5 Detector de Humo

Los detectores de humo son dispositivos pensados, en el ámbito edificatorio, para detectar la presencia de un incendio en el interior de un edificio. La palabra humo hace referencia a la existencia de productos derivados de una combustión ineficiente con partículas en suspensión (cenizas, entre otros). En el ámbito técnico, es preferible hablar de detectores de incendios, puesto que en el caso de una combustión en la que se generen gases únicamente no puede hablarse de humo.

Los detectores de humo o de incendios pueden ser analógicos (indican nivel de la sustancia o magnitud medida, por ejemplo, concentración de un elemento generado durante el incendio) o digitales (detectan presencia o variación de humo, pero sin poder cuantificar el nivel). Si se atiende al uso en edificios, existen 3 tipos de detectores de humo o de incendios:

- **Detectores convencionales:** Detectan humo, fuego, calor o cualquier combinación de estos. Se instalan, habitualmente, en lazo e indican si hay una alarma en las diferentes zonas de detección.
- **Detectores convencionales direccionables:** Estos sistemas son capaces de detectar, también, la existencia de calor, humo o fuego (o combinación de los mismos) pero indican si existe alarma por cada elemento de detección, no por

cada zona. Por tanto, son unos sistemas más precisos que permiten una mejor acción de los servicios de intervención.

2.2.5.1 Tipos de detectores de humos y de incendios

Los detectores más utilizados son de cuatro tipos y se basan en diferentes principios físicos y de funcionamiento:

- **Detectores de humos iónicos.** Funcionan por un principio de ionización del aire en caso de que entre humo en la cámara de detección; los elementos detectados pueden ser invisibles a simple vista. Se instalan generalmente en la industria química, ya que son más sensibles a fuegos de crecimiento rápido o de humo no visible. Es importante mencionar que no funcionan si el aire tiene una velocidad superior a unos 0,5 m/s.
- **Detectores ópticos.** El humo que entra en el equipo hace que se interrumpa la transmisión de un rayo de luz que hay en su interior, lo cual genera una alarma. Son sistemas para detección puntual y existen dos tipos: los detectores ópticos analógicos y los digitales. Estos detectores se colocan en los conductos de extracción de aire (ventilación) del local para que detecten la presencia de humo en grandes estancias o en el edificio en general. Se utilizan, habitualmente, en viviendas, centros comerciales, supermercados, cárceles o bodegas, entre otros.

2.2.6 Detector de Movimiento

Un detector de movimiento, o sensor de presencia, es un dispositivo electrónico equipado de sensores que responden a un movimiento físico, estos transmiten señales de radio de alta frecuencia y dan aviso sí reconocen algún tipo de desplazamiento en el área de cobertura. Identifica si se ha dado movimiento en un área determinada, para a partir de allí emitir una señal con la que se alerte de la situación y se tome una decisión. Las señales que se obtienen son en el orden de corriente directa, con impulsos de menos de 10Hz, por ello es mejor filtrar las etapas para una respuesta mejor a menos de 10Hz y así evitar falsas alarmas (ver figura 2).

2.2.6.1 Tipos de sensores de movimiento

- **Sensores de vibración:** detectan el movimiento a través de la vibración. Es muy común en automóviles ya que alertan a los dueños de golpes o forcejeo en coche.
- **Sensores infrarrojos:** Estos usan luces infrarrojas para detectar los cambios de calor, como cuando una persona se mueve en un determinado espacio, esta luz lo detecta con ayuda del sensor infrarrojo. Si una persona se mueve dentro del rango asignado, la alarma se activa. Este tipo de sensores es el más común ya que es económico, pequeño y confiable.
- **Sensores foto-eléctricos:** Es una combinación de láser y rayos foto-eléctricos. El láser es disparado a la otra esquina de la habitación y si es interrumpido se activa la alarma.
- **Sensores acústicos:** Pueden detectar la energía de cualquier sonido, incluso vidrios que se han roto. En caso de detectar algún ruido la alarma se activará inmediatamente.
- **Detector de movimiento por video:** Tiene un sistema de cámaras que se activará si los sensores identifican un movimiento en el área de vigilancia para empezar con una grabación
- **Detector de movimiento inalámbrico:** Se comunica sin requerir de alambres o cables en un sistema de alarma. No precisa de instalaciones o complicaciones complejas

Diagrama de conexión del sensor PIR con 4558

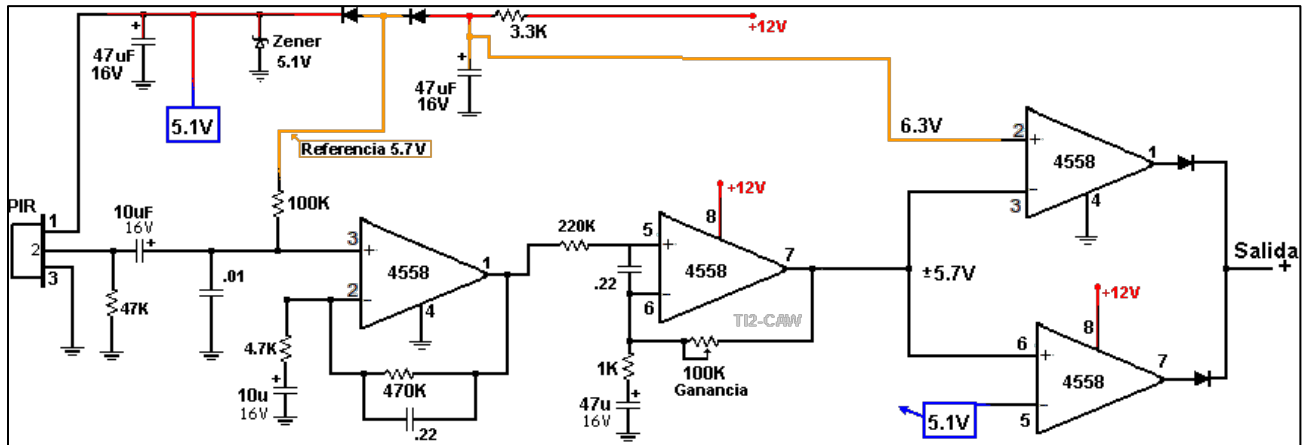


Figura 2: Diseño del sensor de movimiento de amplificador para PIR con 4558

Fuente: <https://www.proyectoelectronico.com/>

2.3 Bases Legales

Según Villafranca D. (2002) “Las bases legales no son más que se leyes que sustentan de forma legal el desarrollo del proyecto” explica que las bases legales “son leyes, reglamentos y normas necesarias en algunas investigaciones cuyo tema así lo amerite”. A continuación, se presentan los argumentos legales que servirán de sustento jurídico al presente proyecto:

2.3.1 Constitución de la República Bolivariana de Venezuela

De la Constitución de la República Bolivariana de Venezuela, se han citado los siguientes artículos:

"...Artículo 98.- La creación cultural es libre. Esta libertad comprende el derecho a la inversión, producción y divulgación de la obra creativa, científica, tecnológica y humanística, incluyendo la protección legal de los derechos del autor o de la autora sobre sus obras. El Estado reconocerá y protegerá la propiedad intelectual sobre las obras científicas, literarias y artísticas, invenciones, innovaciones, denominaciones, patentes, marcas y lemas de acuerdo con las condiciones y excepciones que establezcan la ley y los tratados internacionales suscritos y ratificados por la República en esta materia".

Este artículo ha sido considerado como base legal ya que la implementación del sistema SCADA, está comprendida dentro la "libertad cultural" mencionada en el mismo, considerándose además como avance tecnológico para la institución, ya que por primera vez se ha implementado una propuesta de esta índole. Además, el Estado proporcionará protección legal de los derechos de autor de este proyecto socio innovador.

2.3.2 Ley Orgánica De Ciencia, Tecnología e Innovación (LOCTI)

De la Ley Orgánica De Ciencia, Tecnología e Innovación, se pueden citar los siguientes artículos:

Artículo 1. "...desarrollar los principios orientadores que en materia de ciencia, tecnología e innovación y sus aplicaciones, establece la Constitución de la República Bolivariana de Venezuela, organizar el Sistema Nacional de Ciencia, Tecnología e Innovación, definir los lineamientos que orientarán las políticas y estrategias para la actividad científica, tecnológica, de innovación y sus aplicaciones, con la implantación de mecanismos institucionales y operativos para la promoción, estímulo y fomento de la investigación científica..."

Es obligación de esta ley, implantar mecanismos institucionales y operativos para promover y estimular la investigación científica, la apropiación social del conocimiento y la transferencia e innovación tecnológica, en este sentido la investigación se considera una innovación tecnológica para la U.P.T.N.M. (Universidad Politécnica Territorial del Norte de Monagas)

Sujetos de esta ley

Artículo 3. "...los sujetos que forman parte del Sistema son:

1. El Ministerio de Ciencia y Tecnología, sus organismos adscritos y las entidades tuteadas por éstos, o aquellas en las que tengan participación.
2. Las instituciones de educación superior y de formación técnica, academias nacionales, colegios profesionales, sociedades científicas, laboratorios y centros de investigación y desarrollo, tanto público como privado.

3. Los organismos del sector privado, empresas, proveedores de servicios, insumos y bienes de capital, redes de información y asistencia que sean incorporados al Sistema.
4. Las unidades de investigación y desarrollo, así como las unidades de tecnologías de información y comunicación de todos los organismos públicos.
5. Las personas públicas o privadas que realicen actividades de ciencia, tecnología, innovación y sus aplicaciones."

De acuerdo con el artículo mencionado, la implementación del sistema SCADA está sujeto a esta ley, ya que es un proyecto socio innovador para obtener el título de Ingeniero en Instrumentación y Control (formación universitaria).

2.4 Definición de términos

- **Automatizar:** Automático o mecanizado nuestros movimientos y acciones, el trabajo y la producción en general.
- **Base de datos:** Conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.
- **Controladores:** Dispositivos electrónicos con fin de lograr que una máquina o dispositivo funcione mediante mandos.
- **DLL:** Una biblioteca de enlace dinámico es el término con el que se refiere a los archivos con código ejecutable que se cargan bajo demanda de un programa por parte del sistema operativo
- **Interfaz:** Es el mecanismo o herramienta que posibilita esta comunicación mediante la representación de un conjunto de objetos, iconos y elementos 30 gráficos que vienen a funcionar como metáforas o símbolos de las acciones o tareas que el usuario puede realizar en la computadora.
- **PLC:** Controladores lógicos programables.
- **Proceso:** Conjunto de fases sucesivas de un fenómeno o hecho complejo.

- **Programación:** la programación refiere a la acción de crear programas o aplicaciones, a través del desarrollo de un código fuente, el cual se basa en el conjunto de instrucciones que sigue el ordenador para ejecutar un programa.
- **Sistema:** Conjunto de cosas ordenadas y relacionadas entre sí. Método o grupo de órganos que regulan una función.
- **Software:** está compuesto por un conjunto de programas que son diseñados para cumplir una determinada función dentro de un sistema, ya sean estos realizados por parte de los usuarios o por las mismas corporaciones dedicadas a la informática.

CAPÍTULO III

MARCO METODOLÓGICO

Tamayo y Tamayo (2003) define al marco metodológico como “Un proceso que, mediante el método científico, procura obtener información relevante para entender, verificar, corregir o aplicar el conocimiento”, dicho conocimiento se adquiere para relacionarlo con las hipótesis presentadas ante los problemas planteados. (p.37)

3.1. Tipo de Investigación

Con respecto al tipo de investigación, Tamayo, M (2003) (p.24) expresa sobre la investigación descriptiva “Comprende la descripción, registro, análisis e interpretación de la naturaleza actual, y la composición o procesos de los fenómenos. El enfoque se hace sobre conclusiones dominantes o sobre cómo una persona, grupo o cosa se conduce o funciona en el presente. La investigación descriptiva trabaja sobre realidades de hecho, y su característica fundamental es la de presentarnos una interpretación correcta.”

Según el autor Arias, F. (2012): “La investigación descriptiva consiste en la caracterización de un hecho, fenómeno, individuo o grupo, con el fin de establecer su estructura o comportamiento. Los resultados de este tipo de investigación se ubican en un nivel intermedio en cuanto a la profundidad de los conocimientos se refiere”. (pag.24).

En relación con lo expresado anteriormente, se dice que la presente investigación puede calificarse como documental descriptiva, pues constituye un estudio sistemático de investigaciones previas ya comprobadas, y a su vez, se realiza bajo la metodología XP, esta metodología es ágil y muy organizada a la hora de desarrollar proyectos en corto tiempo, y cuyo esfuerzo se destina a la implantación de propuestas, que pueden materializarse y brindar soluciones a problemas que se plantean en la sociedad

3.2 Diseño de la Investigación

El diseño de la investigación es el conjunto de directrices que toma el investigador con el fin de observar, analizar y plantear una solución de ser posible a la problemática objeto de la investigación. Según el autor (Santa P. y Feliberto M (2010) P21), define: El diseño experimental es aquel según el cual el investigador manipula una variable experimental no comprobada, bajo condiciones estrictamente controladas. Su objetivo es describir de qué modo y porque causa se produce o puede producirse un fenómeno. Busca predecir el futuro, elaborar pronósticos que una vez confirmados, se convierten en leyes y generalizaciones tendentes a incrementar el cúmulo de conocimientos pedagógicos y el mejoramiento de la acción educativa. (pag.86). Según el autor [Santa P y Feliberto M (2010)(P.35)], define: La Investigación de campo consiste en la recolección de datos directamente de la realidad donde ocurren los hechos, sin manipular o controlar las variables. Estudia los fenómenos sociales en su ambiente natural. El investigador no manipula variables debido a que esto hace perder el ambiente de naturalidad en el cual se manifiesta.(pag.88)

3.3 Nivel de la Investigación

El nivel de investigación se refiere según Arias, F (2012) “al grado de profundidad con que se aborda un objeto o fenómeno”. Así pues, el nivel de investigación establece hasta qué punto se llevará a cabo el estudio del tema o problema planteado. Tomando en cuenta el tipo de investigación, se conocerá el nivel en el cual se basa todo el estudio. También el nivel permite saber qué factores tienen que intervenir para el desarrollo de toda la investigación.

Tomando en cuenta lo anteriormente expuesto, el nivel de investigación que se emplea es Exploratoria, definido por el autor Arias, F(2012) como “la investigación exploratoria es aquella que se efectúa sobre un tema u objeto desconocido o poco estudiado por lo que sus resultados constituyen una visión aproximada de dicho objeto es decir un nivel superficial de conocimientos

3.4 Población y Muestra

3.4.1. Población

La población es todo individuo de características considerables en las estadísticas de una investigación. González & Salazar (2008), realiza la siguiente definición:

“La población es considerada un todo y un universo donde puede estar referida a cualquier conjunto de elementos de los cuales se desea extraer sus características generales y específicas dentro de esto se puede delimitar en el estudio del fenómeno donde ingresa los individuos, objetos en relación a las propiedades y atributos o variables dependiendo de la investigación” (pág. 88).

En el presente trabajo de investigación, la población influyente se refleja en la fuente primaria que manejan y/o administran el sistema de seguridad a través de los sensores de movimiento y temperatura de la Universidad José Antonio Páez, así como también el personal de sistema el cual proporcionará la base de datos de los horarios de clase.

3.4.2. Muestra

La muestra es todo aquel subconjunto considerado en una determinada población, a la cual se aplicará la posterior técnica de recolección de datos. Según Arias, F. (2012), expresa que: “La muestra es un subconjunto representativo y finito que se extrae de la población accesible”. (pág. 83).

Por consiguiente, se da a entender que, la muestra de la investigación presente será considerada una muestra finita, ya que, se limita a los laboratorios del edificio 4, que a su vez conforman la población de esta investigación.

3.5 Técnicas e Instrumentos de recolección de datos

3.5.1. Técnicas de recolección de datos

Es el medio por el cual el investigador facilita la recolección de datos, valiéndose del mismo para obtener la información necesaria. Hurtado, J. (2010), concluye que: “Los aspectos metodológicos se desarrollan a lo largo del marco metodológico y se evidencian en las técnicas utilizadas para la recolección de datos y para el análisis de

resultados... Las técnicas son modos específicos de hacer algo. Por ejemplo, algunas técnicas de recolección de datos son la entrevista y la observación”. (pág. 105 y 110).

La presente investigación, tiene como técnica la entrevista estructurada, la cual, según Arias, F. (2012) define que:

“Es la que se realiza a partir de una guía prediseñada que contiene las preguntas que serán formuladas al entrevistado. En este caso, la misma guía de entrevista puede servir como instrumento para registrar las respuestas, aunque también puede emplearse el grabador o la cámara de video”. (pág. 73).

Por ello, es importante destacar que los investigadores utilizarán la entrevista estructurada como técnica de recolección de datos, seleccionando la muestra finita antes planteada, para así aplicar la misma, obteniendo entonces los resultados que se desean lograr.

De igual forma, la observación directa es un método por el cual el investigador se vale para obtener, tal y como lo dice su nombre, la información directa del análisis que se desea desarrollar. Hurtado, J. (2010) cita: “La observación directa y natural de los hechos es el punto de partida del método del empirismo. Según Bacon esta observación debe hacerse dejando de lado los prejuicios, a los que este autor llamó idola”. (pág. 112).

El presente trabajo de investigación se vale de la observación directa, específicamente en las aulas y áreas donde se encuentran la mayor vulnerabilidad para obtener rápidamente la información requerida con el deseo de lograr el objetivo de la investigación.

3.5.2. Instrumentos de recolección de datos

Un instrumento sirve como recurso material que se relaciona con el individuo al cual se le hace el análisis. Para Arias, F. (2012), los instrumentos: “Son los medios materiales que se emplean para recoger y almacenar la información. Ejemplo: fichas, cuestionario, guía de entrevista, lista de cotejo, escalas de actitudes u opinión, grabador, cámara fotográfica o de video, etc.”. (pág. 111)

En la presente investigación, tiene como instrumento de recolección de datos la ficha de registro de información que será diseñada por los autores. Esta ficha será diseñada tomando en consideración los objetivos de la investigación, a su vez estará constituida por preguntas cerradas, dicotómicas. Cabe destacar que dicho instrumento será empleado a la muestra determinada.

Cabe destacar que dicho instrumento será empleado a la muestra determinada.

3.6 Fases de la Investigación

Fase I: “Análisis de los puntos de mayor riesgo en la universidad, examinando los niveles superiores del edificio de estudios múltiples para realizar los requerimientos funcionales y no funcionales del sistema.” Actividades:

- Estudio de los niveles superiores del edificio de estudios múltiples para poder detectar las zonas menos vigiladas que estos posean
- Capturar en imágenes los puntos a atacar para demostrar de manera más específica los puntos ciegos.
- Realizar los Requerimientos Funcionales y No funcionales del sistema.

Fase II: “Diseño de la interfaz del sistema SCADA utilizando las herramientas computacionales” Actividades:

- Actualización del entorno de desarrollo de visual studio para diseñar utilizando el lenguaje de C#
- Obtener las DLL correspondientes al para desarrollar el diseño de la interfaz
- Diagramas correspondientes para realizar el sistema
- Diseño del wireframe de la interfaz donde el usuario se sienta cómodo y todo el aprendizaje sea entendible de la manera más rápida.

Fase III: “Desarrollo del sistema SCADA para la automatización de los sensores y obtener toda la información en tiempo real”. Actividades:

- Se realiza investigación mediante la web, tomando referencias e información acerca de los tipos Software disponible.

- Se propuso el tipo de software a utilizar teniendo en cuenta características favorables, que a su vez concuerdan con la idea principal del automatizado.
- Creación de los sensores a través del Programa Proteus 8.0 y realizar la conexión con el sistema Scada
- Desarrollo del sistema SCADA

CAPÍTULO IV RESULTADOS

4.1 Fase I: Análisis de los puntos de mayor riesgo en la universidad, examinando los niveles superiores del edificio 4 para realizar los requerimientos funcionales y no funcionales del sistema

4.1.1 Estudio de los Niveles Superiores

El estudio se realizó mediante a la observación directa, es un método de recolección de datos que consiste en observar al objeto de estudio dentro de una situación particular, esto se hace sin intervenir ni alterar el ambiente en el que el objeto se desenvuelve, de lo contrario, los datos obtenidos no serían válidos.

Desde el año 2014 hasta la actualidad la Universidad José Antonio Páez, ha sufrido un abandono muy excesivo del personal por la situación económica del país lo que ha provocado una vulnerabilidad a la seguridad del instituto, el personal encargado de que todas las actividades que se realicen de la manera más segura y que ningún usuario del establecimiento se vea afectado por motivos de inseguridad no se encuentra monitoreando todas las áreas de los edificios, lo que genera puntos ciegos donde se genera acontecimientos indebidos.

Pasillos Vacíos



Figura 3: 5to Piso del edificio de estudios múltiples

Fuente: Valente Victor (2019)

En la figura 3 se puede observar el 5to Piso del edificio de estudios múltiples esta fotografía fue tomada alrededor de las 4:00pm, según el cronograma de la universidad este es un horario de actividades en el instituto, se realizó una espera para medir el tiempo en el que el personal de seguridad apareciera y la presencia de la vigilancia en un promedio de 40 min fue completamente imperceptible en el área, en este tiempo se puede desarrollar muchos acontecimientos que en el peor de los casos puede llegar a terminar en una tragedia.

Por otra parte tenemos el 4to Piso de este edificio donde también se realizó un tiempo de espera de aproximadamente 40 min y se logró visualizar el personal encargado de los laboratorios de computación pero estos solo se dirigen a un área específica y no están monitoreando todos los laboratorios a cada momento lo que de igual forma permite que se

pueda desarrollar cualquier acontecimiento, las aulas que posee este nivel también se encuentran desprotegidas y con una falta excesiva de vigilancia.

Falta de Personal de Seguridad



Figura 4: 4to Piso del edificio de estudios múltiples
Fuente: Valente Victor (2019)

4.1.2 Capturar en imágenes los puntos a atacar para demostrar de manera más específica los puntos ciegos.

En esta actividad se realizó un recorrido por la universidad evaluando cada punto ciego en específico, demostrando la vulnerabilidad de la universidad y el riesgo que poseen de sufrir pérdidas tanto en daño en la infraestructura como hurtos de los equipos que aunque no sean de última generación son útiles para el aprendizaje y representa un alto costo a la hora de recuperar lo robado, en esta actividad se demuestra que la José Antonio Páez necesita

atacar con urgencia estos puntos ciegos para la seguridad de ellos y de los usuarios de las aulas del instituto.

Equipos de aprendizaje de computación



Figura 5: Laboratorio de Computación del edificio de estudios múltiples
Fuente: Valente Victor (2019)

En la imagen anterior se muestra un laboratorio con equipos de aprendizaje para los estudiantes, al final se aprecia una cámara que puede grabar los acontecimientos en el laboratorio pero no hay un sistema que pueda avisar de manera inmediata si hay movimiento, o peor si hay altas temperaturas por algún corto circuito en las computadoras lo que provocaría pérdidas en materiales e infraestructura. También puede haber hurtos por algún usuario que esté disponiendo del laboratorio en un horario fuera de clases, o venta de productos ilegales.

Equipos electrónicos para el aprendizaje



Figura 6: Laboratorio de Electrónica del edificio de estudios múltiples

Fuente: Valente Victor (2019)

En la Figura 6 observamos el mismo caso que en la Figura 4, vemos como en el nivel superior del edificio de estudios múltiples se encuentra los laboratorios de electrónica que poseen equipos para el aprendizaje de los estudiantes, se aprecia una cámara al final pero esto no asegura que puede haber una respuesta en tiempo real a la hora de un acontecimiento en este laboratorio, también es un área donde solo se veían 3 usuarios del aula y el resto del nivel completamente solo, estas personas están en una situación de peligro e inseguridad bastante grande.

Aula de Clases sin Control ni Vigilancia

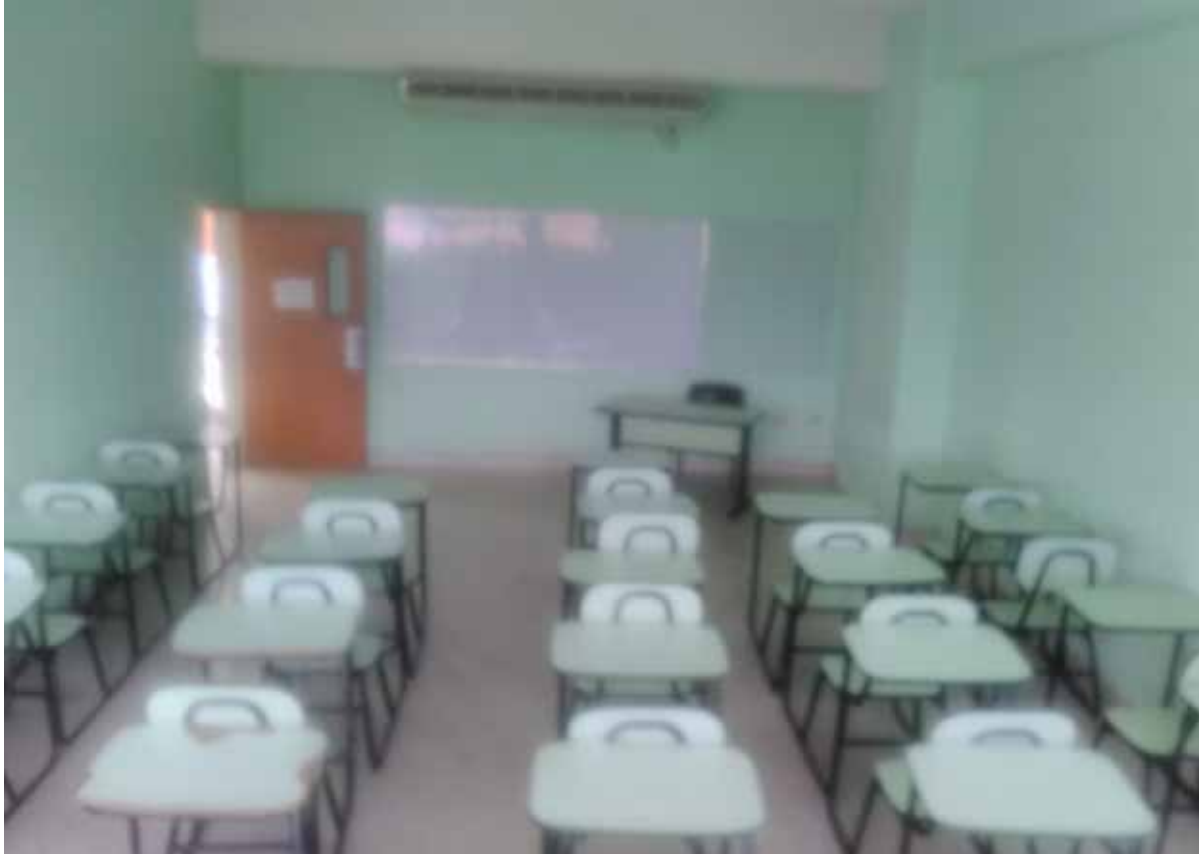


Figura 7: Salón de clases del edificio de estudios múltiples nivel 4

Fuente: Valente Victor (2019)

En la Figura 7 se puede apreciar que los salones se encuentran abiertos sin ningún tipo de vigilancia, y como apreciamos en la Figura 4 se aprecia que es un nivel donde no hay presencia de la seguridad y cualquier usuario se puede encerrar en las aulas a realizar todo tipo de actos indebidos, desde venta y consumo de sustancia estupefacientes a actos sexuales o violentos, dañando la mesa o las sillas, lo que provoca una pérdida a la universidad tanto en aspectos físico como en aspecto sociales, con un sensor de movimiento se puede detectar el uso de los espacios fuera del horario de clases y responder de manera inmediata a la situación.

Banco sin ninguna vigilancia



Figura 8: Banco Nacional de Crédito del edificio de estudios múltiples
Fuente: Valente Victor (2019)

En la Figura 8 podemos apreciar un banco dentro de la Universidad José Antonio Páez, esta sucursal está en el primer piso donde hay el mayor tránsito de gente y es más difícil que sufra un ataque vandálico, pero siempre existe la posibilidad de que alguna persona se esconda espere el cierre del instituto y salga a cometer el crimen alrededor de la universidad hay un terreno donde los asaltantes pueden simplemente escapar, con un sensor de movimiento que detecte la presencia de estos individuos se podrá evitar o responder mucho más rápido a la situación que pueda llegar acontecer en la universidad.

4.1.3 Requerimientos del Sistema

4.1.3.1 Requerimientos Funcionales

- El sistema permite la configuración de la base de datos específica donde se almacena toda la información
- El sistema posee un usuario administrador que permita registrar otros usuarios por perfil y tipo de permiso
- El sistema permite que el usuario acceda a la cámara específica seleccionando la fotografía del aula que desea supervisar
- El sistema permite que el usuario administrador registre los horarios de uso por área de trabajo
- El sistema puede conectarse a un dispositivo electrónico que sea capaz de detectar movimiento, temperatura, humo.
- Posee una luz o una señal que detecte que existe movimiento durante la transmisión del video de vigilancia.
- Muestra a través de un medidor de temperatura los grados Celsius que posee el área a monitorear
- El sistema detecta señales de Humo y activa una alarma en caso de que aumenten superando el nivel permitido
- El sistema realiza todas las funciones en tiempo real para que la respuesta a las detecciones sean inmediatas
- El sistema grabará todas las acciones que se realice dentro de los espacios donde esté conectado.
- El sistema tiene un espacio donde el usuario puede ver las grabaciones en un tiempo determinado.
- Se necesita una computadora con procesadores de 2.50GHz en adelante
- Para una correcta funcionalidad se necesita una versión de Windows 7 en adelante con una memoria ram mínimo de 4gb

4.1.3.2 Requerimientos no Funcionales

- El sistema se conecta a los equipos electrónicos por puerto serial
- Se utilizó Microsoft SQL server como gestor de base de datos
- El sistema registra toda las acciones que ocurran durante su actividad en el servidor SQL utilizando las sentencias adecuada
- Los usuarios que ingresen pasan por una medida de seguridad que determina cuanto acceso tienen y que acciones pueden realizar
- Al momento de crear los usuarios se le asigna un perfil dependiendo del tipo de perfil el usuario tiene a algunos módulos y a poder modificar o no el contenido de los módulos
- El lenguaje donde se desarrolló el sistema es C#, este lenguaje se comunica muy bien con los equipos electrónicos
- Se utilizó el simulador de Proteus 8.0, este programa trae la librería de arduino con todos sus componentes para así poder demostrar la funcionalidad del sistema.
- Se realiza la conexión con el proteus y el lenguaje c# a través del puerto serial utilizando el programa virtual Serial Port Driver para simular los puertos
- El sistema se puede conectar con cualquier cámara, todas las áreas que se deseen vigilar es posible con la instalación de los equipos electrónicos adecuados

4.2 Fase II: Diseño de la interfaz del sistema SCADA utilizando las herramientas computacionales

4.2.1 Actualizar el entorno de desarrollo de visual studio para diseñar utilizando el lenguaje de C#

Para el desarrollo del sistema se utilizó las siguientes herramientas de Microsoft, todas ellas funcionan en conjunto con el entorno de desarrollo de Visual Studio, a continuación las herramientas que lograron que el sistema funcione con el objetivo general planteado:

4.2.1.1 Visual Studio

Microsoft Visual Studio. Es un entorno de desarrollo integrado(IDE, por sus siglas en inglés) para sistemas operativos Windows. Soporta varios lenguajes de

programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic .NET, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros. Se lanzó en 1998 con la versión Visual Studio 2016 esta versión fue la base para el sistema de desarrollo de Microsoft para los siguientes 4 años, en los que Microsoft migró su estrategia de desarrollo al .NET Framework. Luego de esos 4 años en el 2002 Microsoft lanza al mercado la versión Visual Studio .NET,

Esta es una plataforma de ejecución intermedia multilenguaje, de forma que los programas desarrollados en .NET no se compilan en lenguaje máquina, sino en un lenguaje intermedio (CIL - Common Intermediate Language) denominado Microsoft Intermediate Language (MSIL). En una aplicación MSIL, el código no se convierte a lenguaje máquina hasta que ésta se ejecuta, de manera que el código puede ser independiente de plataforma (al menos de las soportadas actualmente por .NET). En el 2005 se realiza una nueva actualización de visual studio La actualización más importante que recibieron los lenguajes de programación fue la inclusión de tipos genéricos, similares en muchos aspectos a las plantillas de C++.

Con esto se consigue encontrar muchos más errores en la compilación en vez de en tiempo de ejecución, incitando a usar comprobaciones estrictas en áreas donde antes no era posible. C++ tiene una actualización similar con la adición de C++/CLI como sustituto de C# manejado. Se incluye un diseñador de implantación, que permite que el diseño de la aplicación sea validado antes de su implantación. También se incluye un entorno para publicación web y pruebas de carga para comprobar el rendimiento de los programas bajo varias condiciones de carga. Visual Studio 2010, el IDE se rediseña para una mejor legibilidad. Se han eliminado gradientes y líneas innecesarias para hacer más simple su uso.

Por ultimo 9 años después nos trae a la versión actual del visual studio 2019, este entorno de desarrollo permite trabajar con Windows SQL Server que nos permite almacenar toda la información que genera la aplicación, puede manejar los proyectos que se realiza en un departamento donde se tiene a varios desarrolladores trabajando en equipo, este entorno de desarrollo permite que se pueda realizar las tareas de todos

de manera organizada ya que te permite conectar a herramientas de trabajo como el Azure ops o Git Lab.

Entorno donde se Desarrolló el proyecto

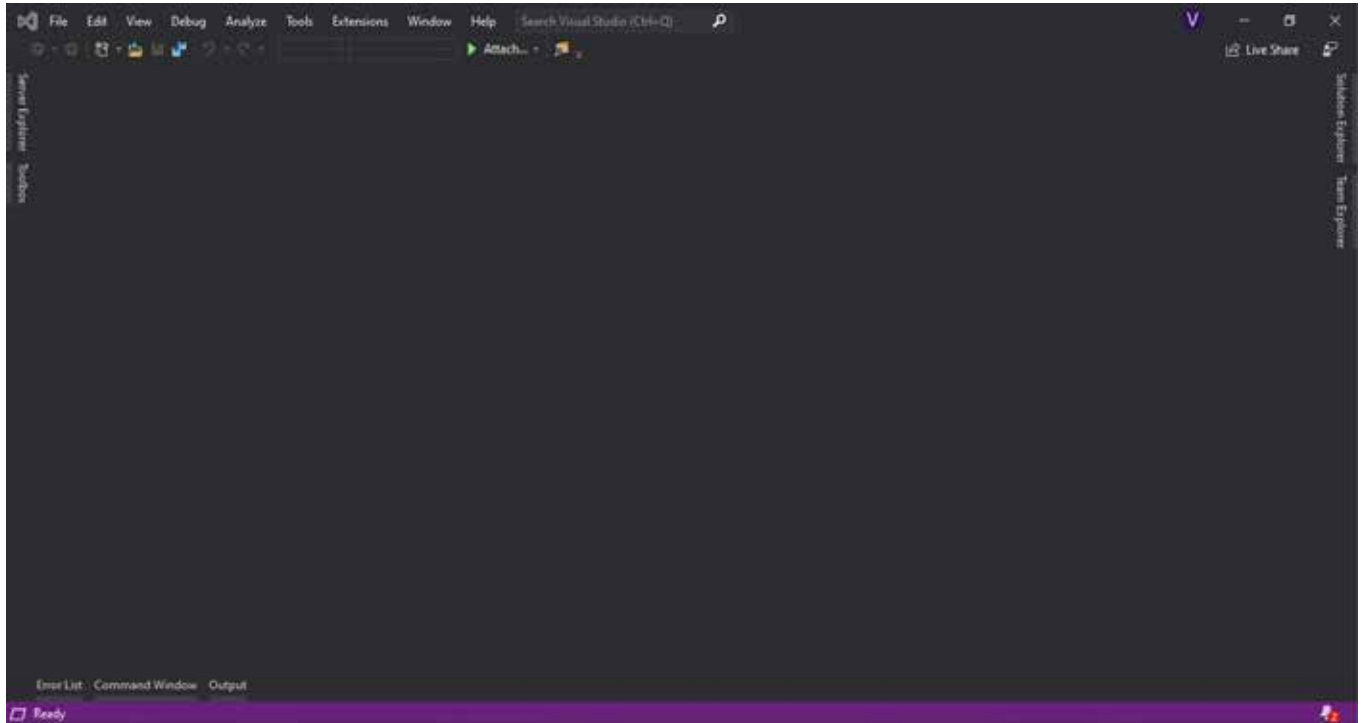


Figura 9: Entorno de desarrollo Visual Studio 2019

Fuente: Valente Victor (2019)

4.2.1.2 Microsoft SQL Server

Microsoft SQL Server es un sistema de gestión de base de datos relacional (RDBMS) producido por Microsoft. Su principal lenguaje de consulta es Transact-SQL, una aplicación de las normas ANSI / ISO estándar Structured Query Language (SQL) utilizado por ambas Microsoft y Sybase. Puede ser configurado para utilizar varias instancias en el mismo servidor físico, la primera instalación lleva generalmente el nombre del servidor, y las siguientes - nombres específicos (con un guion invertido entre el nombre del servidor y el nombre de la instalación).

Características de Microsoft SQL Server:

- Soporte de transacciones.
- Escalabilidad, estabilidad y seguridad.

- Soporta procedimientos almacenados.
- Incluye también un potente entorno gráfico de administración, que permite el uso de comandos DDL y DML gráficamente.
- Permite trabajar en modo cliente-servidor, donde la información y datos se alojan en el servidor y las terminales o clientes de la red sólo acceden a la información.
- Además permite administrar información de otros servidores de datos.

Este sistema incluye una versión reducida, llamada MSDE con el mismo motor de base de datos pero orientado a proyectos más pequeños, que en su versión 2005 pasa a ser el SQL Express Edition, que se distribuye en forma gratuita. Microsoft SQL Server constituye la alternativa de Microsoft a otros potentes sistemas gestores de bases de datos como son Oracle, Sybase ASE, PostgreSQL o MySQL.

Interfaz de Usuario

SQL Server proporciona unos interfaz que han cambiado durante los años, de los cuales los más conocidos tales como los interfaz gráficos que están utilizados como herramienta de desarrollo estándar a los desarrolladores y administradores. La interfaz gráfica hasta 2005 incluyó el Enterprise Manager con una vista de árbol de los distintos objetos y con la capacidad de manejarlos; y el Query analyzer como interfaz textual para ejecutar comandos de TSQL. En la versión 2005 las dos herramientas se unificaron a una –el SQL Server Management Studio (SSMS), y a partir de 2008 fue incluida la opción de trabajar con el Visual Studio– la interfaz estándar de desarrollo de Microsoft (a los distintos lenguajes, BI, etc.).

Otro interfaz opcional es la utilización de Línea de comandos, con herramientas como SQLCmd, ISQL, OSQL que posibilita la ejecución de scripts y procesamiento por lotes. Desde 2008 se puede desarrollar con SQLCmd (SQL Command) a través del SSMS sin interconectarse al interfaz textual de Windows. Otra opción en el ámbito de scripts es la utilización del lenguaje de scripts Powershell de Microsoft. Aparte de los interfaces estándares de SQL Server, se puede ejecutar comandos de TSQL con herramientas de conexión como ODBC y OLE-DB.

Gestor de Base de datos

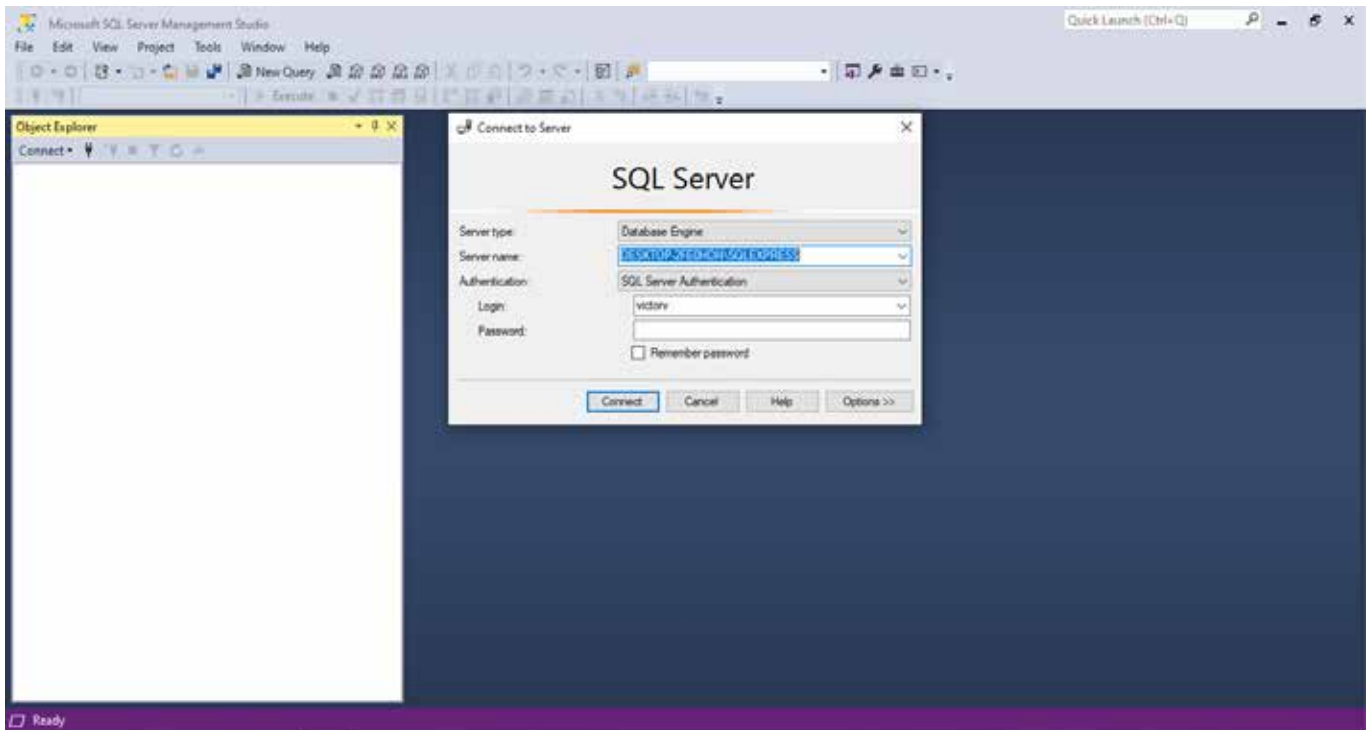


Figura 10: Interfaz de SQL Server Management Studio

Fuente: Valente Victor

4.2.1.3 Visual Studio Code

Visual Studio Code es un editor de código fuente desarrollado por Microsoft para Windows , Linux y macOS. Incluye soporte para la depuración, control integrado de Git, resaltado de sintaxis, finalización inteligente de código, fragmentos y refactorización de código. También es personalizable, por lo que los usuarios pueden cambiar el tema del editor, los atajos de teclado y las preferencias. Es gratuito y de código abierto, aunque la descarga oficial está bajo software propietario.

Visual Studio Code se basa en Electron, un framework que se utiliza para implementar aplicaciones Node.js para el escritorio, que se ejecuta en el motor de diseño Blink. Aunque utiliza el framework Electron, el software no usa Atom y en su lugar emplea el mismo componente editor (Monaco) utilizado en Visual Studio Team Services (anteriormente llamado Visual Studio Online), Fue anunciado el 29 de abril

de 2015 por Microsoft en la conferencia Build de 2015. Una versión preliminar se lanzó poco después.

El código combina la interfaz de usuario optimizada de un editor moderno con asistencia y navegación de código enriquecido y una experiencia de depuración integrada, sin la necesidad de un IDE completo. Visual Studio Code, cuenta con herramientas de Debug hasta opciones para actualización en tiempo real de nuestro código en la vista del navegador y compilación en vivo de los lenguajes que lo requieran (por ejemplo en el caso de SASS a CSS). Además de las extensiones, tendremos la posibilidad de optar por otros themes o bien configurarlo a nuestro gusto. Para modificar el esquema de colores y los iconos

Características

- Se puede utilizar como lenguajes de programación.
- Visual Studio Code es una herramienta que tiene soporte nativo para gran variedad de lenguajes, entre ellos podemos destacar los principales del desarrollo Web: HTML, CSS, JavaScript, python, C# entre otros.
- Posibilidad de configurar la vista a nuestro gusto. De esta forma, podremos tener más de un código visible al mismo tiempo, las carpetas de nuestro proyecto y también acceso a la terminal o un detalle de problemas, entre otras posibilidades

Entorno Visual Studio para programa en c++

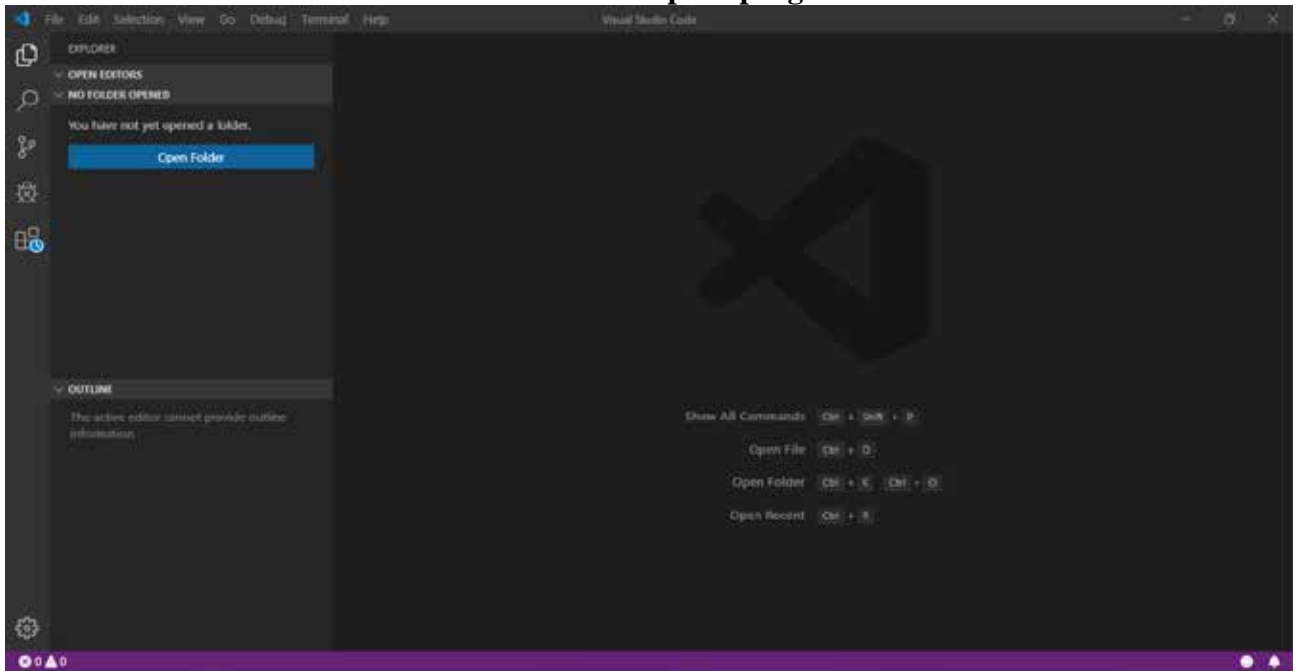


Figura 11: Interfaz del Visual Studio Code

Fuente: Valente Victor

4.2.2 Obtención de las DLL correspondientes al para desarrollar el diseño de la interfaz

Para el desarrollo del sistema se obtuvo las siguientes DLL, esto con el fin de permitir que la aplicación realizara y demostrara de manera más atractiva y específica todas las funcionalidades

4.2.2.1 AGauge.dll

Esta librería permite desarrollar un indicador de temperatura en C#, un indicador de temperatura es un instrumento de instalación que puede procesar la señal de sensores de temperatura e indicarlos en pantalla, sí, los indicadores de temperatura permiten una valoración sencilla y económica de sensores de resistencia, los sensores se pueden conectar directamente a los indicadores de temperatura, lo que evita tener que instalar un transductor. Algunos indicadores de temperatura sólo permiten la valoración de determinados sensores

Cómo funciona el código

El código consta de una aplicación C # y un control personalizado. El control personalizado es realmente la parte interesante.

Derivando del control

Derivamos de, Control ya que esto no nos da todas estas propiedades que en realidad no se utiliza como la propiedad usercontrol nos proporciona, por ejemplo.

```
public partial class AGauge : Control
```

Anulación de propiedades útiles

Para las propiedades, se puede utilizar la override palabra clave (si se puede anular) para indicarle al programa que llame a esta propiedad anulada en lugar de la implementación de la clase base, que en nuestro caso es la implementación Control.

```
public override System.Drawing.Color BackColor..  
public override System.Drawing.Font Font..  
public override System.Windows.Forms.ImageLayout BackgroundImageLayout..
```

Propiedades personalizadas

Para poder personalizar aún más el control en el diseñador, se necesita agregar algunas propiedades propias. Por ejemplo:

```
[System.ComponentModel.Browsable(true),  
System.ComponentModel.Category("AGauge"),  
System.ComponentModel.Description("The value.")]  
public Single Value..
```

El Browsable atributo muestra la propiedad en la caja de herramienta o no, el Category atributo indica donde mostrar la propiedad si se selecciona la vista categorizada, por último el Description atributo agrega una descripción a la propiedad que el diseñador puede mostrar en la barra de herramientas.

Argumentos de eventos personalizados

Para agregar algunos datos al evento, se deriva de los argumentos de eventos estándar y se agrega una variable que se inicializa en el constructor. Esto mantendrá la información adicional enviada.

```

public class ValueInRangeChangedEventArgs : EventArgs
{
    public Int32 valueInRange;
    public ValueInRangeChangedEventArgs(Int32 valueInRange)
    {
        this.valueInRange = valueInRange;
    }
}

```

Delegado de Eventos

El controlador de eventos que "escucha" para que un evento sea de un tipo que "entienda" un evento similar. Con la declaración de delegado, se define el tipo como:

```

public delegate void ValueInRangeChangedDelegate(Object sender,
    ValueInRangeChangedEventArgs e);

```

Y el evento

```

[Description("This event is raised if the value falls into a defined range.")]
public event ValueInRangeChangedDelegate ValueInRangeChanged;

```

El evento es del tipo que se define en la delegatedeclaración. El Descriptionatributo permite al diseñador mostrar una descripción del evento en la Caja de herramientas.

Constructor

Se llama al constructor cuando se crea el control, aquí se establece el estilo del control para habilitar el doble almacenamiento en búfer. Esto no es realmente necesario ya que se puede realizar un búfer doble, pero no está de más hacerlo.

```

public AGauge()
{
    InitializeComponent();
    SetStyle(ControlStyles.OptimizedDoubleBuffer, true);
}

```

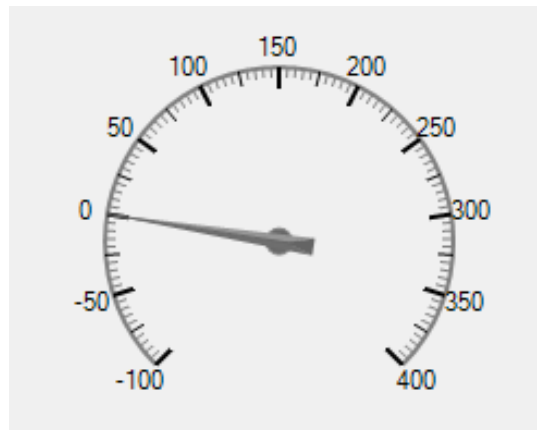


Figura 12: Medidor de la librería de AGauge.dll

Fuente: Valente Victor

4.2.2.2 AForge.dll

AForge.NET es un marco de código abierto C # diseñado para desarrolladores e investigadores en los campos de Visión por Computadora e Inteligencia Artificial: procesamiento de imágenes, redes neuronales, algoritmos genéticos, lógica difusa, aprendizaje automático, robótica, etc. El marco está compuesto por el conjunto de bibliotecas y aplicaciones de muestra, que demuestran sus características:

- **AForge.Imaging** - biblioteca con rutinas y filtros de procesamiento de imágenes;
- **AForge.Vision** - biblioteca de visión por computadora;
- **AForge.Video**: conjunto de bibliotecas para el procesamiento de video;
- **AForge.Neuro** - biblioteca de computación de redes neuronales;
- **AForge.Genetic** - biblioteca de programación de evolución;
- **AForge.Fuzzy** - biblioteca de cálculos difusos;
- **AForge.Robotics** - biblioteca que proporciona soporte para algunos kits de robótica;
- **AForge.MachineLearning** - biblioteca de aprendizaje automático;
- etc.

Esta aplicación de muestra el uso de diferentes clases de fuente de video de espacios de nombres como AForge.Video y AForge.Video.DirectShow . Con la ayuda

de esta sencilla aplicación, es posible reproducir cámaras web USB locales, archivos de video y transmisiones de video JPEG / MJPEG, que generalmente proporcionan las cámaras de video IP. Además, la aplicación demuestra el uso de VideoSourcePlayer , que convierte toda la tarea de reproducir video en pocas líneas de código.

Además, la aplicación puede ser útil para probar el trabajo de simulación de dos cámaras del mismo modelo, el uso de la clase VideoCaptureDevice para capturar video desde una cámara web USB y hacer instantáneas con él (la cámara debe tener un botón de obturador para hacer instantáneas o admitir la activación de software externo). La aplicación muestra cómo establecer el tamaño del video capturado y las instantáneas

El marco AForge.NET está representado no solo por un conjunto de bibliotecas y la documentación correspondiente, sino que también consiste en la recopilación de diferentes aplicaciones de muestra, que tienen como objetivo demostrar el uso de API de diferentes marcos. La recopilación de muestras junto con la documentación del marco proporcionado sirve una buena fuente de información y permite a las personas comenzar a trabajar rápidamente con el marco y aplicarlo para resolver sus propias tareas en sus diferentes aplicaciones.

Biblioteca de procesamiento de Imagen

La biblioteca de procesamiento de imágenes contiene un conjunto de filtros y herramientas de procesamiento de imágenes diseñados para abordar muchas tareas diferentes de visión por computadora y análisis / procesamiento de imágenes. Por el momento, la biblioteca contiene el siguiente conjunto de filtros, que crece cada vez más a medida que se desarrollan nuevos:

- Filtros de color (escala de grises, sepia, inversión, rotación de canales, extracción de canales, reemplazo de canales, filtrado de canales, filtrado de colores, filtrado de colores euclidianos, corrección lineal de canales RGB)
- Filtros HSL (corrección lineal, brillo, contraste, saturación, modificador de tono, filtrado HSL)

- Filtros YCbCr (corrección lineal, filtrado YCbCr, extracción / reemplazo de canales)
- Filtros de binarización (umbral, umbral con acarreo, tramado ordenado, tramado Bayer, Floyd-Steinberg, Burkes, Jarvis-Judice-Ninke, Sierra, Stevenson-Arce, métodos de tramado Stucki)
- Binarización adaptativa (estadísticas simples de imágenes);
- Filtros de morfología matemática (erosión, dilatación, apertura, cierre, golpear y fallar, adelgazamiento, engrosamiento)
- Filtros de convolución (media, desenfoque, nitidez, bordes, desenfoque gaussiano, nitidez basada en el núcleo gaussiano)
- 2 filtros de origen (fusionar, intersectar, sumar, restar, diferencia, mover hacía, transformar)
- Detectores de bordes (homogeneidad, diferencia, sobel, astuto)
- Corrección gamma, filtro mediano, suavizado conservador, nerviosismo, pintura al óleo, pixellate, esqueletización simple
- Contador de blobs y filtro de etiquetado de componentes conectados
- Generadores de textura (nubes, mármol, madera, laberinto, textil), texturador, filtro con textura, filtro de fusión de textura
- Cambio de tamaño y rotación (vecino más cercano, bilineal, bicúbico)
- Filtrado de frecuencia con FFT
- Estadísticas de imagen

```
// load an image
System.Drawing.Bitmap image = (Bitmap) Bitmap.FromFile( fileName );
// format image
AForge.Imaging.Image.FormatImage( ref image );
```

La biblioteca describe dos interfaces principales `IFilter` y `IInPlaceFilter`, que deben implementarse con todos los filtros de procesamiento de imágenes. La primera interfaz es obligatoria para todos los filtros y describe su funcionalidad. Esto permite la aplicación del filtro a la imagen de origen sin su modificación. En lugar de esto, se

devuelve una nueva imagen como resultado de la rutina de procesamiento de imágenes, pero la imagen de origen se deja intacta. La segunda interfaz se implementa solo con esos filtros, que pueden aplicarse directamente a la imagen de origen, actualizándola como resultado de la rutina de procesamiento de imágenes.



Figura 13: Imagen de una Rosa

Fuente: <https://www.codeproject.com/Articles/16859/AForge-NET-open-source-framework>

La siguiente muestra demuestra el uso de un filtro, que produce una nueva imagen como resultado de su trabajo:

```
// create filter  
IFilter filter = new FloydSteinbergDithering( );  
// apply the filter  
Bitmap newImage = filter.Apply( sourceImage );
```

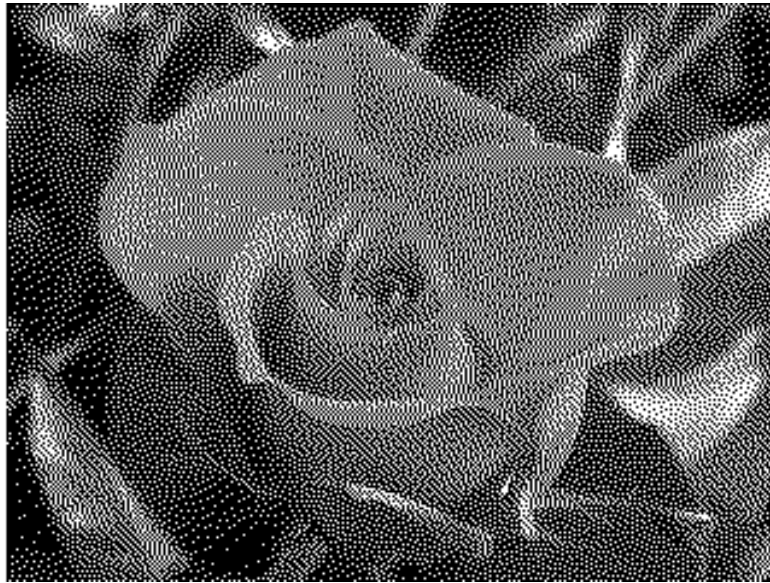


Figura 14: Imagen de una Rosa con el algoritmo de binarización

Fuente: <https://www.codeproject.com/Articles/16859/AForge-NET-open-source-framework>

4.2.2.3 System.IO.dll

El espacio de nombres System.IO (Librería de visual studio) contiene tipos que permiten leer y escribir en los archivos y secuencias de datos, así como tipos que proporcionan compatibilidad básica con los archivos y directorios, esta librería permite instanciar variables del tipo Serial Port, con este tipo de variables podemos indicar el número del puerto que posee el dispositivo al cual vamos a conectar nuestra aplicación. Esta librería maneja todas las variables como ficheros y directorios.

Ficheros y directorios

La biblioteca de clases de la plataforma .NET proporciona una serie de clases que nos permiten trabajar con el sistema de archivos:

System.IO.Directory y System.IO.File proporcionan métodos estáticos para manipular directorios y ficheros, respectivamente. System.IO.DirectoryInfo Y System.IO.FileInfo incluyen métodos para manipular instancias de directorios y ficheros, respectivamente. System.IO.DirectoryInfo representa un directorio concreto, a partir del cual se pueden obtener sus subdirectorios (con GetDirectories([mask])) y los ficheros que incluye (con GetFiles([mask])). Por su parte, System.IO.FileInfo

representa un fichero concreto, que se puede obtener directamente especificando su path o a partir de la enumeración de ficheros de un directorio con `GetFiles()`.

Una vez que se posee un fichero con el que trabajar, se utiliza uno de sus métodos `Open...` para poder acceder y modificar su contenido: `Open()`, `OpenRead()`, `OpenWrite()`, `OpenText()`. Cualquiera de los métodos anteriores devuelve una instancia de `System.IO.Stream`. La clase base de los streams en la plataforma .NET es la clase abstracta `System.IO.Stream`, que proporciona funciones de acceso síncrono (`Read()`, `Write()`) y asíncrono (`BeginRead()`, `BeginWrite()`, `EndRead()`, `EndWrite()`). `System.IO.FileStream` se utiliza para acceder directamente al contenido de ficheros. De hecho es el tipo devuelto por una llamada a `File.Open()`. `System.IO.MemoryStream` permite construir streams en memoria.

Los stream readers proporcionan acceso de lectura a un stream. `System.IO.BinaryReader` permite leer datos en binario: `ReadInt16()`, `ReadBoolean()`, `ReadDouble()`, etc. `System.IO.TextReader` es una clase abstracta utilizada como base común de las dos siguientes:

- `System.IO.StreamReader` hereda de `TextReader` e implementa los métodos `ReadLine()` para leer una línea de texto y `ReadToEnd()` para leer lo que quede de stream.
- `System.IO.StringReader` también hereda de `TextReader` y se utiliza para simular streams a partir de cadenas de caracteres.

```
static void Main(string[] args)
{
    string filename = @"../AssemblyInfo.cs";
    string line;

    Console.WriteLine
("*****");
```

```
StreamReader stream = new StreamReader(filename);
// StreamReader stream = File.OpenText (filename);

do {
    line = stream.ReadLine();
    Console.WriteLine (line);
} while (line != null);

stream.Close();

Console.WriteLine
("*****");

Console.ReadLine();
}
```

Los stream writers sirven para escribir en streams:

- System.IO.BinaryWriter permite escribir datos en binario, mediante la utilización del método sobrecargado Write().
- System.IO.TextWriter es la clase abstracta que sirve de base para StreamWriter y StringWriter.
- System.IO.StreamWriter hereda de TextWriter y sirve para escribir cadenas de texto en streams.
- System.IO.StringWriter también hereda de TextWriter y simula streams en cadenas de caracteres.

```

public class MyWriter
{
    private Stream stream;

    public MyWriter(Stream stream)
    {
        this.stream = stream;
    }

    // Almacena la representación binaria de un double

    public void WriteDouble(double myData)
    {
        byte[] b = myData.GetBytes();
        stream.Write(b,0,b.Length);
    }

    public void Close()
    {
        stream.Close();
    }
}

static void Main(string[] args)
{
    string filenameI = @"../AssemblyInfo.cs";
    string filenameO = @"../COPIA_AssemblyInfo.cs";
    string line;

    StreamReader streamI = new StreamReader(filenameI);
    StreamWriter streamO = new StreamWriter(filenameO);

    while ((line = streamI.ReadLine()) != null) {
        streamO.WriteLine (line);
    }

    streamI.Close();
    streamO.Close();
}

```

La clase `System.IO.Ports.SerialPort` que se incluye con .NET es una excepción evidente. Una vez que este insertado en el código, se puede pasar a definir la variable que contendrá el control Serial Port, de la siguiente manera.

```
SerialPort spPuertoSerie = new SerialPort("COM1", 9600, Parity.None, 8, StopBits.One);
```

De esta forma se crea el objeto «spPuertoSerie», cuyos parámetros son:

- Nombre del puerto: el puerto al que el programa se va a conectar, en el ejemplo «COM1».
- Velocidad de transmisión: la velocidad de transmisión.
- Paridad: Sirve para verificar si hay errores en la transmisión. Existen cuatro tipos posibles de paridad: par (even), impar (odd), marcada (mark) y espaciada (space). También está permitido no usar paridad (none), como en el ejemplo.
- Bits de datos: Es la cantidad de datos que se van a enviar en cada paquete. Este valor puede ser 5, 7 u 8. Valor más habitual es el 8.
- Bits de parada: Indica el final del paquete, Los valores permitidos son 1, 1.5 o 2. También vale para la sincronización de la comunicación.

También la librería permite crear los objetos de una manera más sencilla, si no se conoce los parámetros de la comunicación.

```
// Se instancia la variable para el manejo del puerto serie
```

```
SerialPort spPuertoSerie = new SerialPort();
```

Para pasar los parámetros se puede realizar de esta otra forma:

```
// Parametros para iniciar el puerto serie
```

```
spPuertoSerie.PortName = "COM1";
```

```
spPuertoSerie.BaudRate = 9600;
```

```
spPuertoSerie.Parity = Parity.None;
```

```
spPuertoSerie.StopBits = StopBits.One;
```

```
spPuertoSerie.DataBits = 8;
```

Después de tener ya creado el objeto que se va a encargar de la comunicación, solo falta abrir la comunicación de la siguiente manera.

```
// Abrir la comunicación
```

```
spPuertoSerie.Open();
```

Para cerrar la comunicación del puerto serie lo se hace de la siguiente forma.

```
// Cerrar la comunicación
```

```
spPuertoSerie.Close();
```

4.2.2.4SensorPIR.dll

Es una librería de proteus que permite simular un detector de movimiento, los detectores PIR (Passive Infrared) o Pasivo Infrarrojo, reaccionan sólo ante determinadas fuentes de energía tales como el calor del cuerpo humano o animales. Básicamente reciben la variación de las radiaciones infrarrojas del medio ambiente que cubre. Es llamado pasivo debido a que no emite radiaciones, sino que las recibe. Estos captan la presencia detectando la diferencia entre el calor emitido por el cuerpo humano y el espacio alrededor.

Su componente principal son los sensores piroeléctrico. Se trata de un componente electrónico diseñado para detectar cambios en la radiación infrarroja recibida. Generalmente dentro de su encapsulado incorporan un transistor de efecto de campo que amplifica la señal eléctrica que genera cuando se produce dicha variación de radiación recibida, la información infrarroja llega al sensor piroeléctrico a través de una lente de fresnell que divide el área protegida en sectores. Se distribuyen lentes con diferentes características: gran angular, cortina, corredor, antimascotas, etc.

Cuando se instalan los sensores infrarrojo (PIR) y se lo energiza por primera vez, este comenzara a “acostumbrarse” a la radiación infrarroja del ambiente (todos los cuerpos que están a una temperatura superior a 0° Kelvin, emiten radiación infrarroja negativa; esta radiación infrarroja aumenta si aumenta la temperatura del cuerpo en cuestión. Es decir, que recibe la información infrarroja de una pared, el piso, los muebles, etc).Una vez que se mantiene estable, si un intruso ingresa al recinto se experimentará un cambio en la radiación infrarroja del ambiente y el PIR dará una condición de alarma.

Una vez realizada la calibración y ajuste del detector, el led de prueba deberá quedar apagado por norma de instalación. Con objeto de lograr total confiabilidad, esta tecnología integra además, un filtro especial de luz que elimina toda posibilidad de falsas detecciones causadas por la luz visible (rayos solares), así como circuitos especiales que dan mayor inmunidad a ondas de radio frecuencia, así es como funcionan los sensores de movimiento.

4.2.2.5 LM35.dll

Es una librería de Proteus que permite simular un LM35, es un circuito electrónico sensor que puede medir temperatura. Su salida es analógica, es decir, proporciona un voltaje proporcional a la temperatura. El sensor tiene un rango de precisión con la que se puede medir la temperatura. Incluso no es necesario de un microprocesador o microcontrolador para medir la temperatura. Dado que el sensor LM35 es analógico, basta con medir con un multímetro, el voltaje a salida del sensor. Para convertir el voltaje a la temperatura, el LM35 proporciona 10mV por cada grado centígrado. También cabe señalar que ese sensor se puede usar sin offset, es decir que si se mide 20mV a la salida, estaremos midiendo 2°C.

LM35 Y SUS CARACTERÍSTICAS PRINCIPALES

- Resolución: 10mV por cada grado centígrado.
- Voltaje de alimentación. Por ejemplo, este sensor se puede alimentar desde 4Vdc hasta 20Vdc.
- Tipo de medición. Salida analógica.
- Numero de pines: 3 pines, GND, VCC y VSalida.
- No requiere calibración.
- Tiene una precisión de $\pm 1/4^\circ\text{C}$.
- Esta calibrado para medir $^\circ\text{C}$.
-
- Empaquetados comunes:

- TO-CAN.
- TO-220.
- TO-92.
- SOIC8.

4.2.2.6 GasSensor.dll

Una librería de Proteus que ofrece las herramientas para simular un detector de Gas, un detector de gas es un dispositivo que detecta la presencia de gases en un área, a menudo como parte de un sistema de seguridad. Este tipo de equipo se utiliza para detectar una fuga de gas u otras emisiones y puede interactuar con un sistema de control para que un proceso se pueda cerrar automáticamente. Un detector de gas puede hacer sonar una alarma a los operadores en el área donde se produce la fuga, dándoles la oportunidad de irse. Este tipo de dispositivo es importante porque hay muchos gases que pueden ser dañinos para la vida orgánica, como los humanos o los animales.

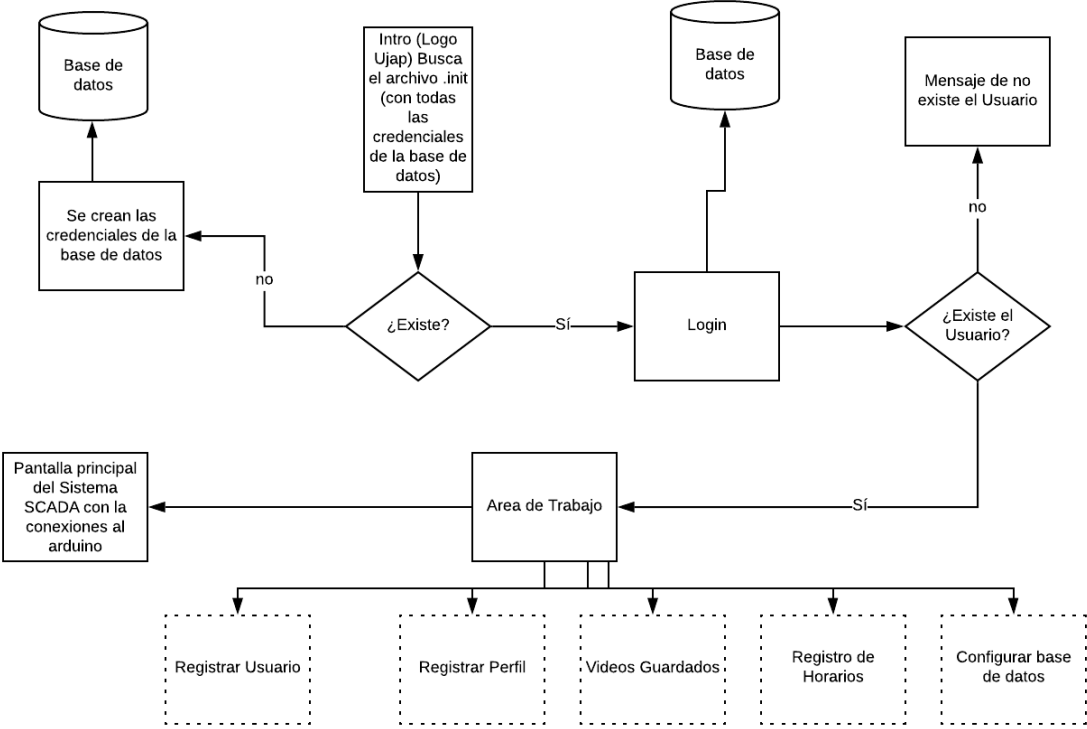
Los detectores de gas se pueden usar para detectar gases combustibles, inflamables y tóxicos, y el agotamiento de oxígeno. Este tipo de dispositivo se usa ampliamente en la industria y se puede encontrar en ubicaciones, como en plataformas petroleras, para monitorear procesos de fabricación y tecnologías emergentes como la fotovoltaica. Pueden ser utilizados en la lucha contra incendios

Características del Sensor de Gas

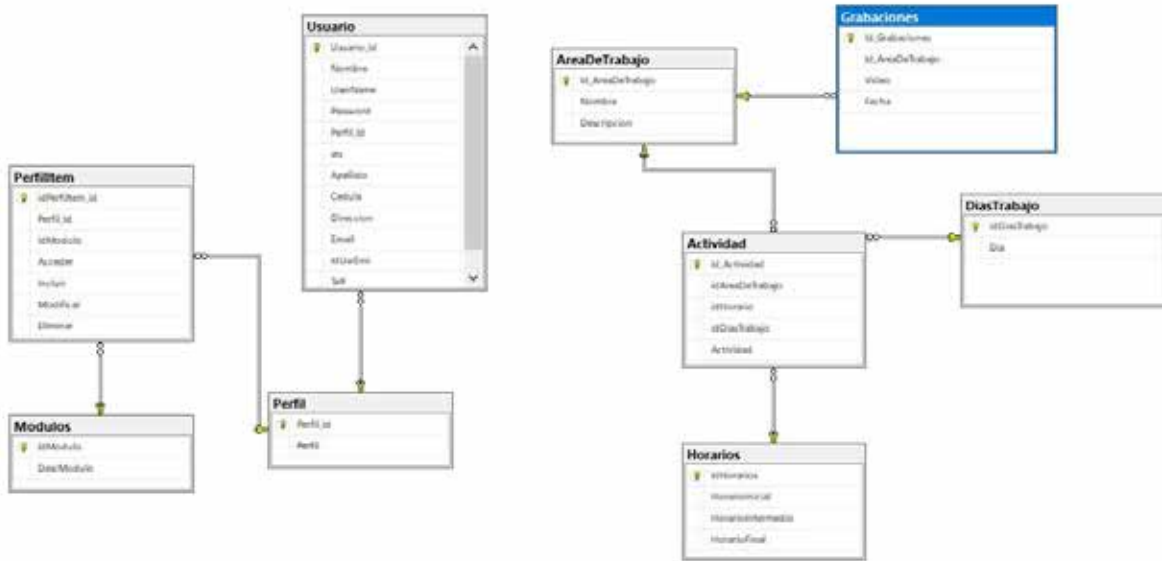
- El sensor se alimenta con 5 volts
- Alta sensibilidad a LPG (LPG) y Gas Natural (LNG)
- Poca sensibilidad a alcohol y humo
- Rápida velocidad de respuesta
- Estable y de larga vida
- Circuito de manejo simple

4.2.3 Diagramas correspondientes para realizar el sistema

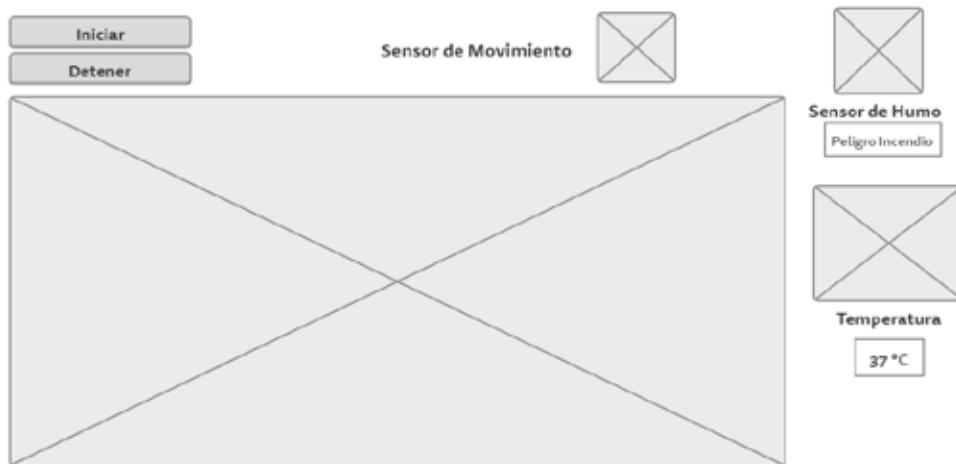
4.2.3.1 Diagrama de procesos



4.2.3.2 Diagrama Entidad Relación



4.2.4 Diseño del wireframe de la interfaz donde el usuario se sienta cómodo y todo el aprendizaje sea entendible de la manera más rápida.



4.3 Fase III: “Desarrollo del sistema SCADA para la automatización de los sensores y obtener toda la información en tiempo real”.

4.3.1 Investigación mediante la web, tomando referencias e información acerca de los tipos Software disponible.

La investigación acerca de los software disponibles para la simulación de cada uno de los sensores fue extensa ya que existen muchas herramientas que te permiten simular los dispositivos electrónicos, pero muchas con restricciones que otras no contemplan pero estas otras aplicaciones no tienen todas las herramientas para poder conseguir alcanzar los objetivos de la investigación, muchas de ellas son útiles para simular la conexiones de los dispositivos pero no permiten la conexión con su aplicaciones.

4.3.1.1 Arduino

Arduino es una plataforma de creación de electrónica de código abierto, la cual está basada en hardware y software libre, flexible y fácil de utilizar para los creadores y desarrolladores. Esta plataforma permite crear diferentes tipos de microordenadores de una sola placa a los que la comunidad de creadores puede darles diferentes tipos de uso, para poder entender este concepto, primero vas a tener que entender los conceptos de hardware libre y el software libre. El hardware libre son los dispositivos cuyas especificaciones y diagramas son de acceso público, de manera que cualquiera puede replicarlos. Esto quiere decir que Arduino ofrece las bases para que cualquier otra persona o empresa pueda crear sus propias placas, pudiendo ser diferentes entre ellas pero igualmente funcionales al partir de la misma base.

El software libre son los programas informáticos cuyo código es accesible por cualquiera para que quien quiera pueda utilizarlo y modificarlo. Arduino ofrece la plataforma Arduino IDE (Entorno de Desarrollo Integrado), que es un entorno de programación con el que cualquiera puede crear aplicaciones para las placas Arduino, de manera que se les puede dar todo tipo de utilidades.

Placa de Arduino Mega



Figura 15: Dispositivo Electrónico Arduino

Fuente: <https://www.xataka.com/basics/que-arduino-como-funciona-que-puedes-hacer-uno>

4.3.1.2 Arduino Aplicación

El software de Arduino es un IDE, entorno de desarrollo integrado (siglas en inglés de Integrated Development Environment). Es un programa informático compuesto por un conjunto de herramientas de programación. El IDE de Arduino es un entorno de programación que ha sido empaquetado como un programa de aplicación; es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI). Además incorpora las herramientas para cargar el programa ya compilado en la memoria flash del hardware.

Es destacable desde la aparición de la versión 1.6.2 la incorporación de la gestión de librerías y la gestión de placas muy mejoradas respecto a la versión anterior y los avisos de actualización de versiones de librerías y cores. En principio el IDE de arduino solo tenía soporte para las placas Arduino y los clones o forks con los mismos microcontroladores que los Arduinos oficiales. Desde la versión 1.6.2 del IDE de arduino.cc y gracias al gestor de placas, podemos añadir soporte a otros microcontroladores y placas al IDE de Arduino, como al ESP8266.

El software hecho para Arduino con el IDE es portable, es decir, el mismo firmware que hemos hecho para un Arduino/Microcontrolador, sirve para otras placas Arduino u otras placas compatibles Arduino como el ESP8266.

Programa Arduino

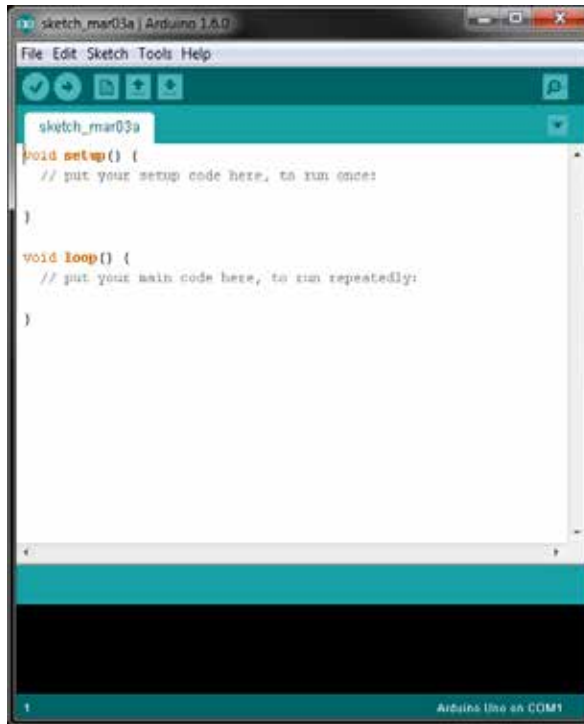


Figura 16: Lenguaje Arduino

Fuente: Valente Victor

4.3.1.3 Proteus

Es un software de simulación para circuitos eléctricos tanto análogos como digitales, además de que te permite crear el layout del PCB y visualizarlo en 3D. Permite diseñar y simular circuitos electrónicos de forma práctica y accesible. Con Proteus las fases de prueba no suponen la necesidad de volver a construir nuevos prototipos, con el ahorro de costos y tiempo que ello supone, los diferentes módulos que componen Proteus se pueden adquirir de forma independiente añadiendo nuevas funcionalidades a medida que aumentan nuestras necesidades de desarrollo y producción.

Además, la capacidad de simular cada una de las familias de microprocesadores también es objeto de adquisición por separado. De esta manera podemos empezar adquiriendo unas funcionalidades básicas e ir adquiriendo progresivamente nuevas características aprovechando al máximo nuestras inversiones en la herramienta y

asegurar al máximo los costes de inversión en el software. En el mundo de la formación, Proteus se muestra como una herramienta magnífica porque permite realizar modificaciones tanto en el circuito como en el programa, experimentando y comprobando de forma inmediata los resultados y permitiéndole de esta forma aprender de forma práctica y sin riesgos de estropear materiales de elevado coste, Proteus consta de dos programas principales Ares e Isis, y los módulos VSM y Electra.

ISIS

El Programa 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.

Módulo VSM

Una de las prestaciones de Proteus, integrada con ISIS, es VSM, el Virtual System Modeling (Sistema Virtual de Modelado), una extensión integrada con ISIS, con la cual se puede simular, en tiempo real, con posibilidad de más rapidez; todas las características de varias familias de microcontroladores, introduciendo nosotros mismos el programa que controlará el microcontrolador y cada una de sus salidas, y a la vez, simulando las tareas que queramos que lleve a cabo con el programa. Se pueden simular circuitos con microcontroladores conectados a distintos dispositivos, como motores, lcd's, teclados en matriz, etc. Incluye, entre otras, las familias de PIC's PIC10, PIC12, PIC16, PIC18, PIC24 y dsPIC33. ISIS es el corazón del entorno integrado PROTEUS. Combina un entorno de diseño de una potencia excepcional con una enorme capacidad de controlar la apariencia final de los dibujos.

ARES

ARES, o Advanced Routing and Editing Software (Software de Edición y Ruteo Avanzado); es la herramienta de enrutado, ubicación y edición de componentes, se

utiliza para la fabricación de placas de circuito impreso, permitiendo editar generalmente, las capas superficial (Top Copper), y de soldadura (Bottom Copper).

Forma Manual

Ejecutando ARES directamente, y ubicando cada componente en el circuito. Tener cuidado al DRC, Design Rules Checker (Verificador de Reglas de Diseño)

Forma Automática

El propio programa puede trazar las pistas, si se guarda previamente el circuito en ISIS(Sistema de Enrutado de Esquemas Inteligente), y haciendo clic en el ícono de ARES, en el programa, el programa compone la Netlist

Método 1 (Autorouter)

- Poner SOLO los componentes en la board
- Especificar el área de la placa (con un rectángulo, tipo "Board Edge")
- Hacer clic en "Autorouter", en la barra de botones superior

Editar la estrategia de ruteo en "Edit Strategies"

- Hacer clic en "OK"

Método 2 (Electra Autorouter)

Utilizando el módulo Electra (Electra Auto Router), el cual, una vez colocados los componentes trazará automáticamente las pistas realizando varias pasadas para optimizar el resultado, con Ares (Software de Edición y Ruteo Avanzado) además se puede tener una visualización en 3D del PCB que se ha diseñado, al haber terminado de realizar la ubicación de piezas, capas y ruteo, con la herramienta "3D Visualization", en el menú output, la cual se puede demorar, solo haciendo los trazos un periodo de tiempo un poco más largo que el de los componentes.

Simulador de los Sensores con Arduino

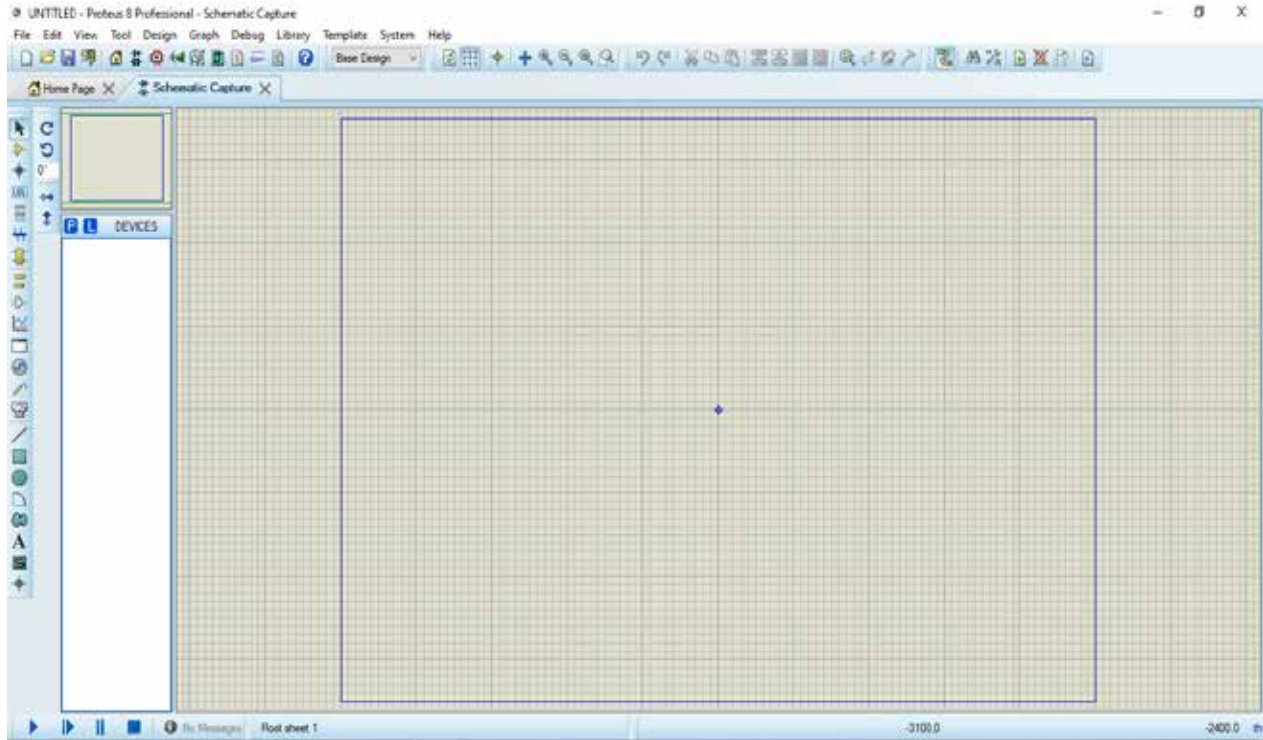


Figura 17: Interfaz de Proteus 8 Profesional

Fuente: Valente Victor

4.3.1.4 Virtual BreadBoard

Virtual Breadboard o 'VBB' es un entorno de desarrollo y emulación para aplicaciones integradas que utilizan microcontroladores con diseños de panel de pruebas que pueden hacer realidad sus diseños. VBB es capaz de simular microcontroladores de las líneas PIC16 y PIC32, también la simulación de Arduino, virtual Breadboard (VBB) te permite conectar "virtualmente" (en la pantalla de tu computadora) los componentes para formar tu circuito o hardware virtual y probar programas o código sin necesidad de contar con el hardware real.

Cuenta con una interfaz intuitiva y fácil y le proporciona acceso a una amplia colección de plantilla de circuito. Es capaz de analizar de simulación, así que si eso es lo que estás buscando es posible que desee ir con algo más en su lugar.

VirtualBreadboard es un programa para el desarrollo de circuitos digitales de factor de forma de tablero y crear el software microcontrolador que los inicia. VirtualBreadboard es la construcción de un asistente diseño inteligente artificial para computación física y aplicaciones de Internet-de-cosas.

Características: VirtualBreadboard

- Una interfaz intuitiva que podría ser utilizado fácilmente por cualquier persona que sabe lo que están haciendo.
- Le permite elegir entre una gran colección de plantillas de circuito o en su lugar opta por crear su propia cuenta.
- Te da la oportunidad de personalizar su proyecto con varios componentes (instrumentos, temporizadores, generadores de funciones, motores, pantallas LCD / LED, y más).
- Muestra el registro de errores, mientras que el circuito está emulando para que pueda ver lo que salió mal en tiempo real.
- Le da la capacidad de utilizar el componente terminal para enviar comandos UART a niveles TTL.
- Perfecto para su uso por los profesores, los estudiantes, como para los ingenieros.

Funciona en base a una librería de componentes virtuales que vienen incluidos en el paquete de software: botones pulsadores, LEDs, potenciómetros, displays LCD, motores servo y steppers, chips de lógica e incluso memorias EEPROM y convertidores ADC y DAC. Adicionalmente VBB incluye una plataforma llamada Micromorph CDK (Kit de Desarrollo de Componentes Micromorph) que te permite construir y añadir tus propios componentes con simulación de entradas y salidas.

Simulador de Arduino que especifica las conexiones

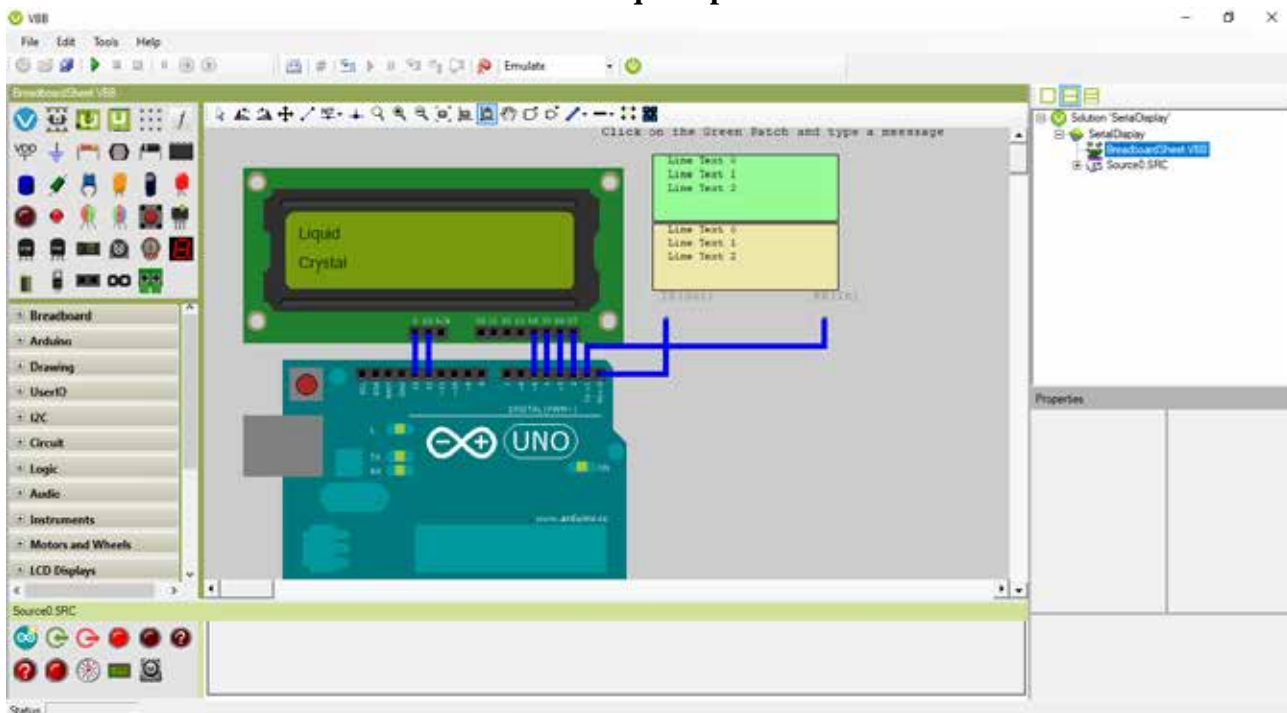


Figura 18: Interfaz de VirtualBreadboard (VBB)

Fuente: Valente Victor

4.3.1.5 Virtual Serial Port Driver

El emulador de puerto serie de Eltima Software permite crear un gran número de puertos COM virtuales y le da la gran posibilidad de emular el comportamiento del puerto serie en su totalidad, le provee una creación, administración, y eliminación flexible, te permite testear software, soportar líneas de control y alta velocidad de transferencia de datos con el puerto virtual. Las aplicaciones pueden intercambiar datos en los puertos virtuales mediante a un cable null-modem virtual. Los puertos virtuales mediante un cable null-modem virtual, los datos enviados de un puerto al otro serán recibidos al instante.

Las computadoras modernas generalmente no manejan Puerto seriales físicos lo que se conoce como puerto COM, todo esto debido a que la mayoría de los dispositivos manejan la comunicación por medio del bluetooth o por TCP/IP, sin embargo a veces es necesario manejar un puerto serial, allí es cuando virtual serial port se vuelve

necesario porque te permite emular las conexiones (null-modem) lo que te permite intercambiar datos sin ningún hardware adicional, puedes asignarle nombres personalizados a los puertos virtuales creados.

Los puertos virtuales emulan totalmente a los reales, por lo que las aplicaciones no ven la diferencia, esta aplicación te permite crear pares de puertos serial lo que significa que cualquier datos que se envíe por un puerto inmediatamente lo recibe el otro, cada par de puertos soporta toda la línea de señal de hardware y permiten configurar cualquier tipo de línea de señal de conexión. Si el dispositivo emulado trabaja a una velocidad fija no hay problema en emular la velocidad de baudios necesaria con virtual serial port.

Características de Virtual Serial Port Driver

Las principales características de Virtual Serial Port Driver son:

- Puerto Virtuales ilimitados: te permite crear pares del puerto serial ilimitados, se puede asignar cualquier nombre al puerto incluido el nombre de un puerto serie existente
- Puertos conectados virtualmente: Virtual COM Port Drive conecta los puertos virtuales creados en pares con un cable de modem nulo virtual, asegurando una comunicación aún más rápida que con los puertos reales conectados mediante un cable de modem nulo real
- Trabajan igual que un puerto físico: Los puertos virtuales se comportan exactamente igual que los puertos COM reales.
- Los puertos virtuales soportan todas las líneas de señal y permiten cualquier tipo de información
- Si el dispositivo emulado trabaja a una velocidad fija no hay problema en emular la velocidad de baudios necesaria

Simulador de Puerto serial para la laptop

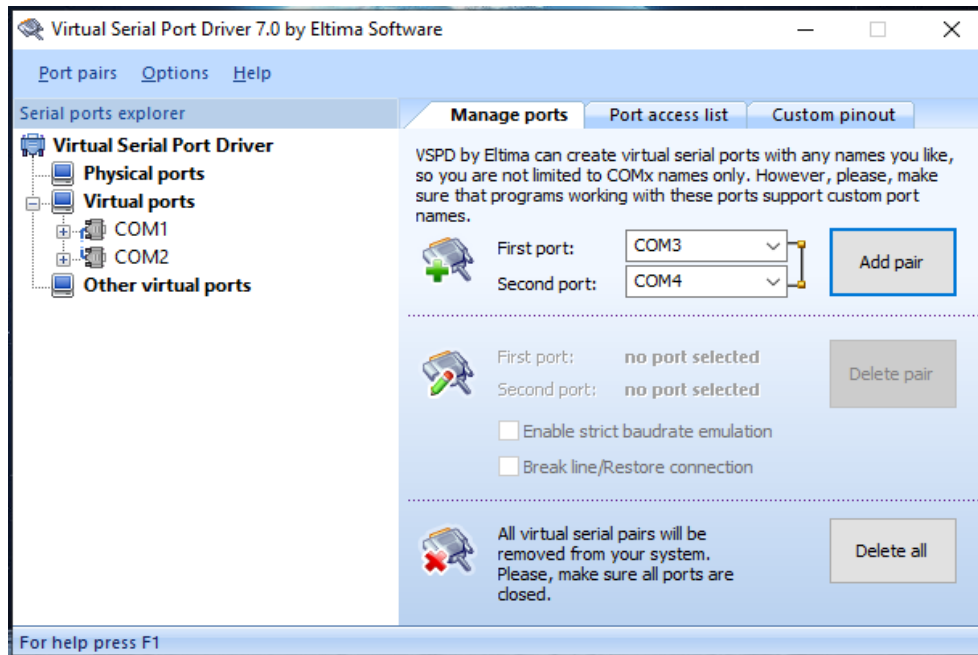


Figura 19: Interfaz de Virtual Serial Port

Fuente: Valente Victor

4.3.2 Tipo de software a utilizar teniendo en cuenta características favorables, que a su vez concuerdan con la idea principal del automatizado.

4.3.2.1 Lenguaje C#

El lenguaje de programación C# fue creado por el danés Anders Hejlsberg que diseño también los lenguajes Turbo Pascal y Delphi. El C# (pronunciado en inglés “C sharp” o en español “C sostenido”) es un lenguaje de programación orientado a objetos. Con este nuevo lenguaje se quiso mejorar con respecto de los dos lenguajes anteriores de los que deriva el C, y el C++, con el C# se pretendió que incorporase las ventajas o mejoras que tiene el lenguaje JAVA. Así se consiguió que tuviese las ventajas del C, del C++, pero además la productividad que posee el lenguaje JAVA y se le denominó C#, algunas de las características del lenguaje de programación C# son:

- Su código se puede tratar íntegramente como un objeto.
- Su sintaxis es muy similar a la del JAVA.
- Es un lenguaje orientado a objetos y a componentes.

- Armoniza la productividad del Visual Basic con el poder y la flexibilidad del C++.
- Ahorramos tiempo en la programación ya que tiene una librería de clases muy completa y bien diseñada.

A pesar que el lenguaje C# forma parte de la plataforma .NET, que es una interfaz de programación de aplicaciones. C# es un lenguaje independiente se relaciona muy bien con equipos físicos electrónicos, la sintaxis y estructuración de C# es muy similar a la C++, ya que la intención de Microsoft con C# es facilitar la migración de códigos escritos en estos lenguajes a C# y facilitar su aprendizaje a los desarrolladores habituados a ellos. Sin embargo, su sencillez y el alto nivel de productividad son equiparables a los de Visual Basic.

Características de C#

A continuación se recoge de manera resumida las principales características de C#, también se comentan aquí también en tanto que tienen repercusión directa en el lenguaje, aunque se indicará explícitamente cuáles son este tipo de características cada vez que se toquen:

- **Sencillez:** C# elimina muchos elementos que otros lenguajes incluyen y que son innecesarios en .NET. Por ejemplo:
 - El código escrito en C# es auto contenido, lo que significa que no necesita de ficheros adicionales al propio fuente tales como ficheros de cabecera o ficheros IDL
 - El tamaño de los tipos de datos básicos es fijo e independiente del compilador, sistema operativo o máquina para quienes se compile (no como en C++), lo que facilita la portabilidad del código.
 - No se incluyen elementos poco útiles de lenguajes como C++ tales como macros, herencia múltiple o la necesidad de un operador diferente del punto (.) acceder a miembros de espacios de nombres (::)

- **Modernidad:** C# incorpora en el propio lenguaje elementos que a lo largo de los años ha ido demostrándose son muy útiles para el desarrollo de aplicaciones y que en otros lenguajes como Java o C++ hay que simular, como un tipo básico decimal que permita realizar operaciones de alta precisión con reales de 128 bits (muy útil en el mundo financiero), la inclusión de una instrucción foreach que permita recorrer colecciones con facilidad y es ampliable a tipos definidos por el usuario, la inclusión de un tipo básico string para representar cadenas o la distinción de un tipo bool específico para representar valores lógicos.
- **Orientación a objetos:** Como todo lenguaje de programación de propósito general actual, C# es un lenguaje orientado a objetos, aunque eso es más bien una característica del CTS que de C#. Una diferencia de este enfoque orientado a objetos respecto al de otros lenguajes como C++ es que el de C# es más puro en tanto que no admiten ni funciones ni variables globales sino que todo el código y datos han de definirse dentro de definiciones de tipos de datos, lo que reduce problemas por conflictos de nombres y facilita la legibilidad del código.
- **Seguridad de tipos:** C# incluye mecanismos que permiten asegurar que los accesos a tipos de datos siempre se realicen correctamente, lo que permite evita que se produzcan errores difíciles de detectar por acceso a memoria no perteneciente a ningún objeto y es especialmente necesario en un entorno gestionado por un recolector de basura
- **Instrucciones seguras:** Para evitar errores muy comunes, en C# se han impuesto una serie de restricciones en el uso de las instrucciones de control más comunes. Por ejemplo, la guarda de toda condición ha de ser una expresión condicional y no aritmética, con lo que se evitan errores por confusión del operador de igualdad (==) con el de asignación (=); y todo caso de un switch ha de terminar en un break o goto que indique

cuál es la siguiente acción a realizar, lo que evita la ejecución accidental de casos y facilita su reordenación.

- **Versionable:** C# incluye una política de versionado que permite crear nuevas versiones de tipos sin temor a que la introducción de nuevos miembros provoquen errores difíciles de detectar en tipos hijos previamente desarrollados y ya extendidos con miembros de igual nombre a los recién introducidos.

Se propuso este lenguaje ya que tiene una gran cantidad de librerías que permite que el sistema desarrollado (en este caso el sistema SCADA) pueda realizar la comunicación entre el software y los equipos electrónicos que cumplan con los objetivos de este sistema.

Lenguaje donde se desarrolló el proyecto



Figura 20: Logo del lenguaje de programación Csharp(C#)

Fuente: Andres Álvarez - Wikipedia

4.3.2.1 Lenguaje Python

Python fue creado a finales de los ochenta por Guido van Rossum en el Centro para las Matemáticas y la Informática (CWI, Centrum Wiskunde & Informatica), en los Países Bajos, como un sucesor del lenguaje de programación ABC, capaz de manejar excepciones e interactuar con el sistema operativo Amoeba, el nombre del lenguaje proviene de la afición de su creador por los humoristas británicos Monty Python, Una innovación mayor en Python fue la unificación de los tipos en Python

(tipos escritos en C), y clases (tipos escritos en Python) dentro de una jerarquía. Esa unificación logró un modelo de objetos de Python puro y consistente. También fueron agregados los generadores que fueron inspirados por el lenguaje Icon.

Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis que favorezca un código legible, se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, dinámico y multiplataforma, posee una amplia cantidad de librerías lo que permite un fácil desarrollo de las aplicaciones, es muy intuitivo y fácil de aprender en comparación a otros lenguajes.

Características

Las características del lenguaje de programación Python se resumen a continuación:

- Es un lenguaje interpretado, no compilado, usa tipado dinámico, fuertemente tipado.
- Es multiplataforma, lo cual es ventajoso para hacer ejecutable su código fuente entre varios sistemas operativos.
- Es un lenguaje de programación multiparadigma, el cual soporta varios paradigmas de programación como orientación a objetos, estructurada, programación imperativa y, en menor medida, programación funcional.
- En Python, el formato del código (p. ej., la indentación) es estructural.

Fuertemente Tipado

El fuertemente tipado significa que el tipo de valor no cambia repentinamente. Un string que contiene solo dígitos no se convierte mágicamente en un número. Cada cambio de tipo requiere una conversión explícita. A continuación un ejemplo de este concepto:

```
# variable "valor1" es integer, variable "valor2" es string  
valor1, valor2 = 2, "5"
```

```
# El metodo int() es para convertir a integer
total = valor1 + int(valor2)
# el metodo str() es para convertir a string
print "El total es: " + str(total)
```

Tipado dinámico

El tipado dinámico significa que los objetos en tiempo de ejecución (valores) tienen un tipo, a diferencia del tipado estático donde las variables tienen un tipo. A continuación un ejemplo de este concepto:

```
# "variable" guarda un valor integer
variable = 11
print variable, type(variable)
# "variable" guarda un valor string
variable = "activo"
print (variable), type(variable)
```

Multiplataforma

Python es multiplataforma, lo cual es ventajoso para hacer ejecutable su código fuente entre varios sistemas operativos, eso quiere decir, soporta las siguientes plataformas para su ejecución:

- Versiones Python para Microsoft Windows (y DOS) (arquitectura x86/x86-64 en presentación de ejecutable, archivo Zip, instalador basado en la Web).
- Versiones Python para Mac OSX (Macintosh) (arquitectura 32bit/64bit en presentación de instalador ejecutable).
- Versiones Python en código fuente (archivo tarball del código fuente comprimido con XZ y con Gz). Para la mayoría de los sistemas Linux/UNIX, usted debe descargar y compilar el código fuente.
- Versiones de Implementaciones Alternativas Python, la versión “tradicional” de Python (tiene nombre código CPython)
- Versiones de Python en otras plataformas, la versión “tradicional” de Python (tiene nombre código CPython), mas esta versión ha sido migrada a un número de plataformas especializadas y/o antiguas

Al principio se propuso utilizar este lenguaje como desarrollo del sistema SCADA ya que python ofrece muchas librerías para lograr la comunicación con los dispositivos electrónicos, pero para la interfaz la manera más efectiva es a través de Html, css, y Django lo que significa que el tiempo de desarrollo iba a ser mayor y no se disponía del tiempo suficiente para la realización por este medio.

Lenguaje de Programación Python



Figura 21: Logo de Python
Fuente: www.python.org

4.3.3 Creación de los sensores a través del Programa Proteus 8.0 y realizar la conexión con el sistema Scada

En esta fase se realizó a través del Programa Proteus 8.0 los sensores de temperatura y humo para la funcionalidad del sistema Scada, estos sensores se conectan por un puerto serial virtual con el Scada realizando la simulación real del envío y la recepción de datos, en ellos se muestra temperaturas generadas por el arduino y el sensor LM35, simultáneamente se desarrolló un sensor de Humo este envía una señal al proyecto Scada indicando si detecto humo en el sensor de incendio, al momento de recibir los datos el sistema realiza las evaluaciones correspondiente y los procesos a ejecutar para mostrar en pantalla lo que sucede.

4.3.3.1 Sensor de temperatura LM35 simulado con proteus y conectado al lenguaje de programación c#

Simulación de sensor LM35

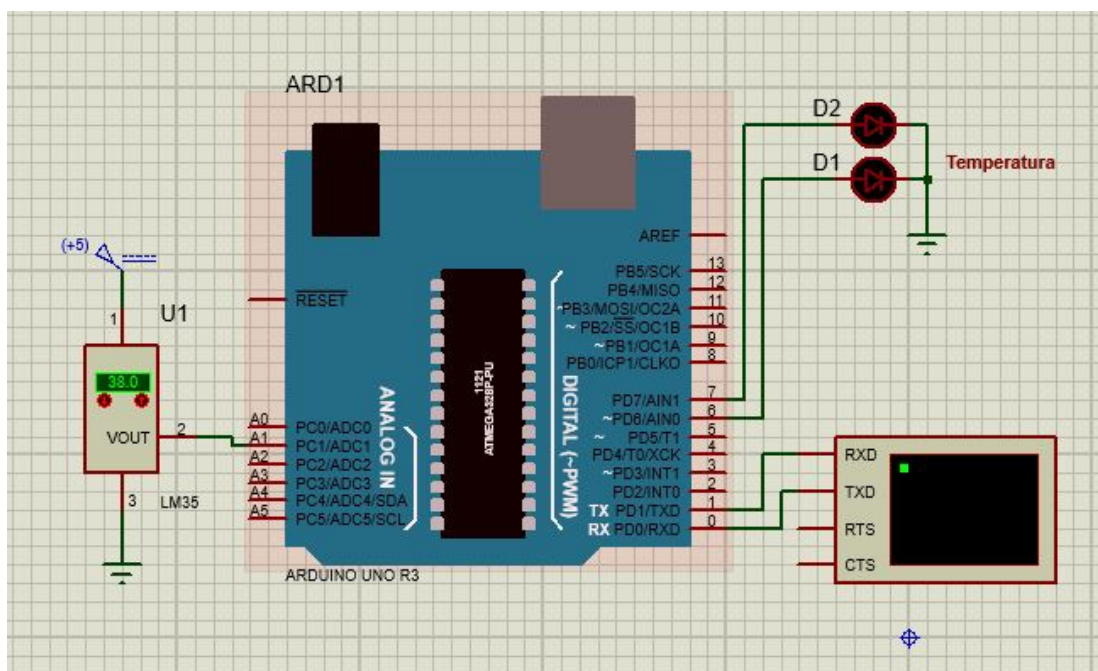


Figura 22: Sensor de Temperatura LM35 conectado a un arduino

Fuente: Victor Valente

En la Figura (Ver Figura 22) se puede apreciar un sensor LM35 conectado a un arduino, que a su vez también se le está anexando un simulador de puerto serial, se le conecta una entrada de 5V que nos devolverá un valor entre 0 a 1024 con estos datos se programa el arduino a través de la aplicación de arduino donde realizo la fórmula del cálculo de temperatura que es $\text{Temperatura} = \text{Valor} * 5 * 100 / 1024$ y se envía el resultado a través del puerto serial, colocando los puertos correspondiente y su velocidad de transmisión de datos.

El sistema SCADA (Ver Figura 23) está configurado de igual forma con su puerto serial asignado escuchando el N° de puerto que se le específico al sensor de temperatura, el programa posee un reloj (timer) donde está revisando cada segundo si ha recibido algún dato a través del puerto, en el momento de que recibe el dato se le

asignan al medidor de temperatura validando que si esta sobrepasa los grados Celsius permitidos acciona una alarma indicando que es un incendio.

Modulo Principal Scada

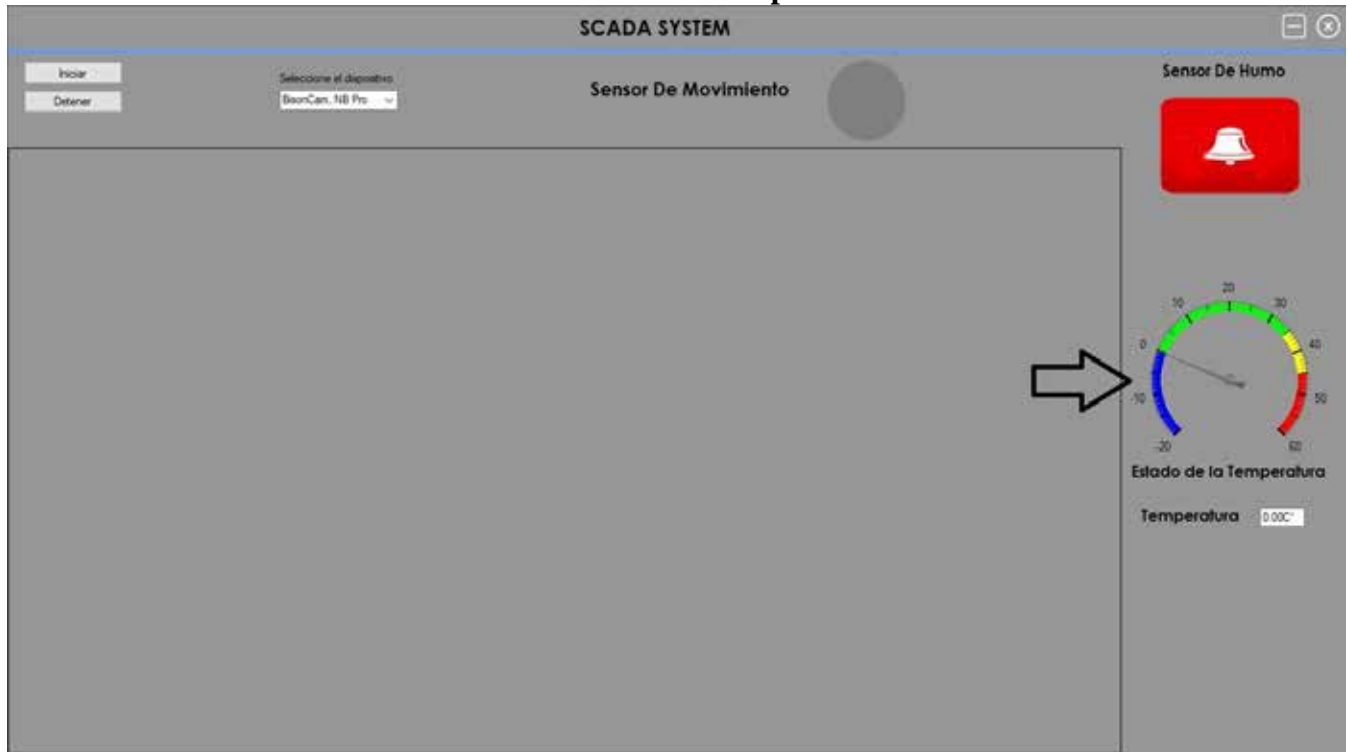


Figura 23: Interfaz Scada Medidor de Temperatura

Fuente: Victor Valente

4.3.3.2 Sensor de Humo simulado con proteus y conectado al lenguaje de programación c#

En la Figura (Ver Figura 24) podemos observar una simulación de un flame sensor que puede detectar si hay humo en el ambiente simplemente con cambiar el switch de 0 a 1 el simula si se presenta una detección de humo en el ambiente, al arduino se le conecta un LCD una pantalla que nos indica con un mensaje de si se encuentra humo o no en el ambiente, por otra parte esta simulación posee un puerto serial donde se transmite esta señal de 0 o 1 al sistema Scada por el puerto correspondiente y con su velocidad de transmisión indicada.

Simulación del Sensor de Humo

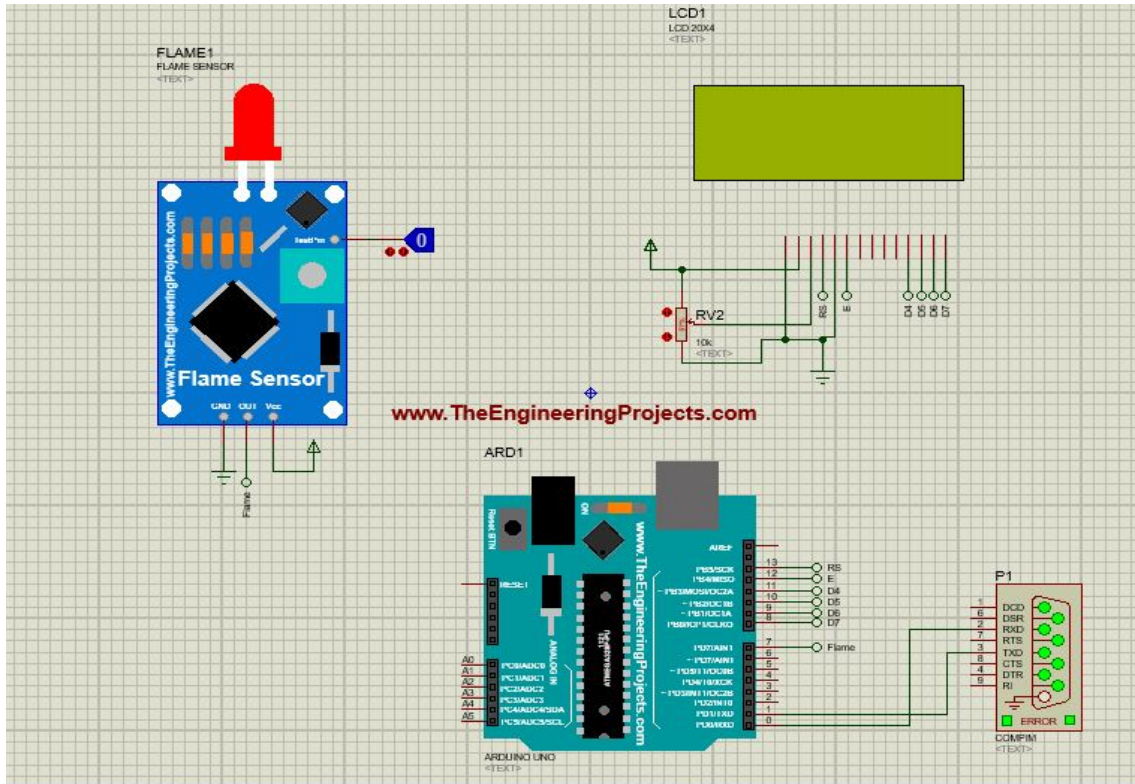


Figura 24: Sensor de Humo (Flame Sensor) conectado a un Arduino

Fuente: Victor Valente

Esta señal transmitida por el arduino la recibe el sistema Scada a través de una variable de tipo serial port, que contiene todos los datos del puerto del sistema y donde espera una señal del Simulador de Humo, este proceso se encuentra en un temporizador (timer) que está escuchando por un puerto y al momento de recibir una señal el sistema valida la información si la señal que recibe es 0 entonces el sistema seguirá actuando de forma normal, si la señal que recibe es 1 entonces sonara una alarma de humo y se muestra una campana del sistema indicando posible incendio (Ver Figura 25).

Indicador del Sensor de Humo

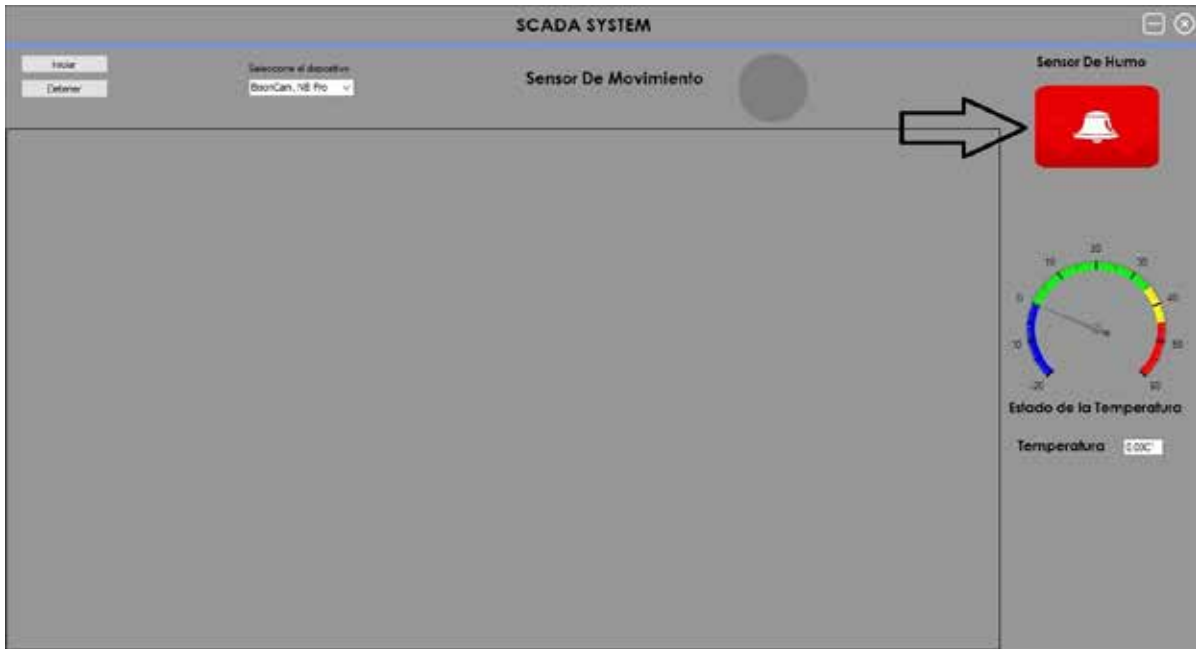


Figura 25: Interfaz Scada Sensor de Humo

Fuente: Victor Valente

4.3.4. Desarrollo del sistema SCADA

Luego de realizar las investigaciones correspondientes, la obtención de las DLL, el programa para lograr la simulación de los sensores, y lograr la comunicación del sistema con el simulador empecé con el desarrollo de la interfaz, esta interfaz cuenta con las características necesarias para hacer que el programa desarrollado sea lo más eficiente posible y permita a la universidad una fácil implementación y uso para tener un ambiente mucho más seguro.

4.3.4.1 Logo Ujap

En esta vista se muestra el Logo de la Universidad como una animación, mientras se realiza esta acción en el backend se está buscando el archivo .init, en este se encuentran las credenciales del servidor, de no encontrar el documento se abre el formulario de configuración de base de datos, de encontrarse el documento se carga la configuración en el sistema una vez que se asignen todos los parámetros del fichero a su variable respectiva el proyecto nos lleva a la interfaz del login del Usuario.

4.3.4.2 Interfaz del Login

En esta parte del sistema, el usuario tiene que colocar su nombre de usuario y su contraseña (Ver Figura 26), usando los métodos de conexión con la base de datos se realiza una consulta buscando los registros con los datos ingresados en el formulario de encontrar similitud en la tabla correspondiente en la base de datos se le permitirá el acceso a este usuario al sistema donde llegara al área de trabajo, de no retornar ningún dato en la query, el sistema arroja un mensaje de error y se debe llamar a un súper usuario (Admin) para su configuración.




Figura 26: Interfaz Login de Usuario
Fuente: Victor Valente

4.3.4.3 Interfaz Configuración Base de Datos

En esta interfaz el usuario puede realizar la configuración respecto a la base de datos, los datos que se piden para las credenciales son:

- Nombre del Servidor
- Nombre de la Base de Datos
- Usuario para ingresar a la Base de datos
- Contraseña para ingresar a la base de datos

Una vez suministrado todo estos datos el sistema te permite la opción de probar la conexión para validar que todos los datos que coloco el usuario estén bien, una vez que lo logre te pregunta si desea guardar las credenciales, en caso de ser primera vez que se ingresa en el sistema este te crea el archivo .init donde guarda toda la información, si se está modificando un campo de la configuración inicial este modifica el documento y también todas las variables del sistema donde se le asigna los parámetros.



The image shows a software window titled "Conexión con el Servidor de Base de Datos". The window has a standard Windows-style title bar with a gear icon on the left and minimize and close buttons on the right. Below the title bar, the text "Datos del Servidor de SystemSCADA" is centered. The main content area is enclosed in a dashed border and contains four text input fields labeled "Servidor SQL", "Base de Datos", "Usuario", and "Clave". Below these fields is a dark grey button labeled "Probar". At the bottom of the window, there is a database icon on the left, and two buttons: a green "Guardar" button and a red "Cancelar" button.

Figura 27: Interfaz Configuración Base de Datos
Fuente: Victor Valente

4.3.4.4 Interfaz Área de Trabajo

En esta área del sistema el usuario puede realizar varias acciones dependiendo de los permisos que posea el perfil donde está ingresando, puede acceder por el menú a la configuración de usuario, al registro del perfil, a los registros del sistema (videos), a la configuración de la base de datos, o al sistema Scada para el monitoreo de las áreas de trabajo que se hallan registrado, al momento de hacer click en cualquier opción el sistema valida si el usuario posee o no el permiso de acceder esto con el fin de evitar que cualquier persona pueda corromper la data del sistema.

selección de area

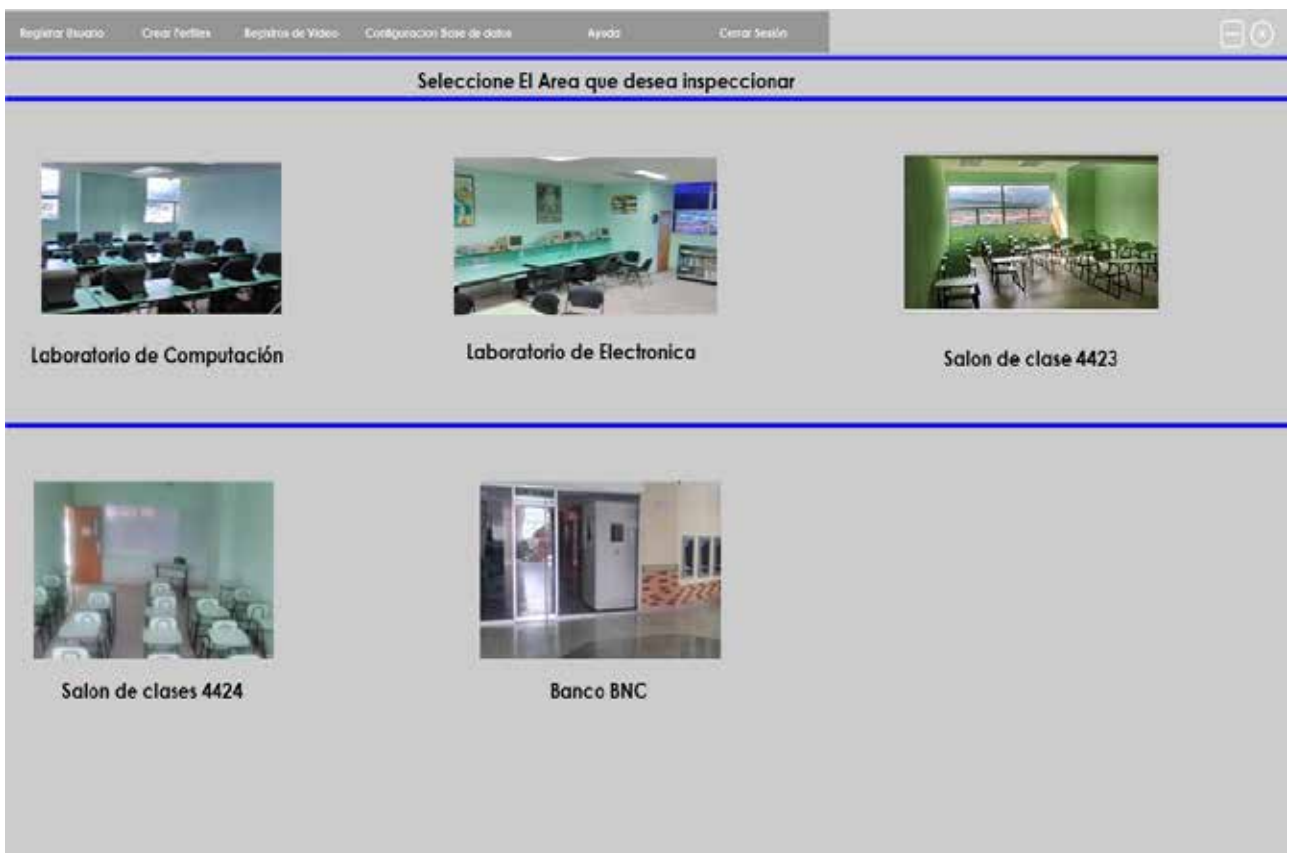


Figura 28: Interfaz Área de Trabajo

Fuente: Victor Valente

4.3.4.5 Interfaz Registrar Usuario

En esta interfaz las personas con permisos, pueden crear o modificar un nuevo usuario o uno ya existente, esto permite almacenar en el sistema toda la información personal para llevar un registro de todos los que pueden acceder al sistema y realizar cualquier tipo de acción, también permite asignar el perfil del usuario esto con la finalidad de saber los tipos de permiso y acceso que este trabajador puede poseer, luego de esto se le crea el usuario y la contraseña para ingresar más adelante y realizar su función correspondiente.



Figura 29: Interfaz Registrar Usuario

Fuente: Victor Valente

4.3.4.6 Interfaz Registrar Perfil

En esta interfaz se realiza el registro de un Perfil, esto permite al usuario crear los permisos que estos perfiles contengan, de esta forma se obtiene un mayor control en el sistema y no permite que ninguna persona que no posea acceso a un área en especifica pueda corromper la información, con este módulo el sistema posee una confiabilidad en la parte de seguridad y permite que cada trabajador se concentre en su área y haga las tareas que se le fue asignada sin la distracción de otros módulos.

Crear Perfil

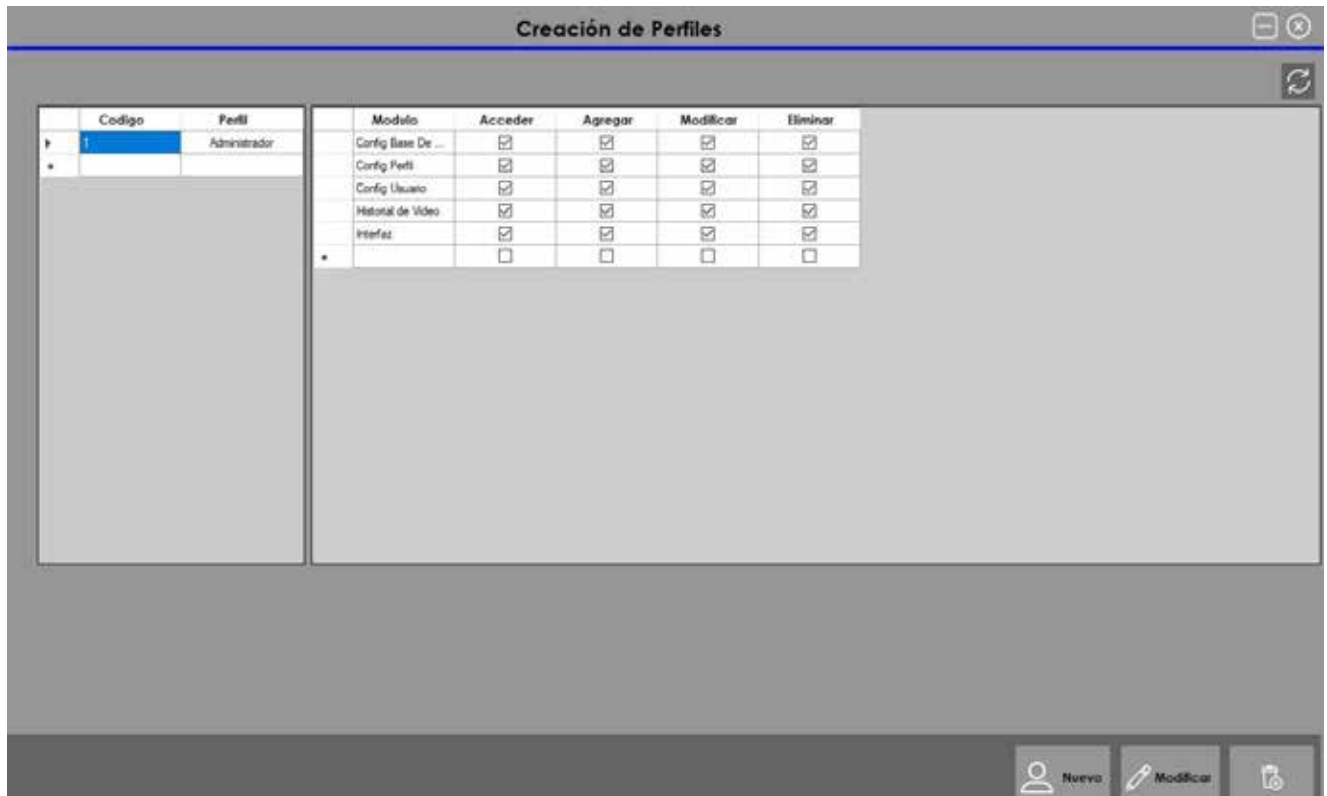


Figura 30: Interfaz Registrar Usuario

Fuente: Victor Valente

4.3.4.7 Interfaz de Registros de video

En este módulo de trabajo el programa trae todos los registros almacenados en la base de datos y llena una grid donde indica el área de trabajo y una breve descripción, también la ruta del video la fecha en que se creó y un botón que nos enviara a la ruta del video para así poder ver todo los cambios que detecto el sistema en ese espacio de trabajo, también permite eliminar cualquier registro pero como ya lo mencione antes eso dependerá del permiso que tenga el usuario asignado en su perfil

Registro de Video



Figura 31: Interfaz Registros de Video

Fuente: Victor Valente

4.3.4.8 Interfaz del sistema SCADA

En esta interfaz se desarrolla todo lo que tiene que ver con el sistema Scada, aquí muestra (Ver Figura31) todo el trabajo que tiene que realizar la aplicación en el momento de iniciar el proceso, la interfaz cuenta con sensor de movimiento, temperatura y de humo el sensor de movimiento se desarrolla a través de una librería que funciona con inteligencia artificial para el reconocimiento de las partes que cambien de posición , permite la conexión de cualquier dispositivo visual ya sea web

cam que esté conectado a la estación de trabajo donde se está ejecutando el proyecto cuenta con la conexión con los dispositivos arduinos, o con cualquier dispositivo que se conecte por puerto serial.

También cuenta con todos los recursos necesarios para demostrar el comportamiento del sistema desde una luz parpadeando al detectar el movimiento hasta los sonidos al detectar incendio o humo en el ambiente, también cuenta con el área de video donde podemos observar todo lo que está ocurriendo a través de la cámara, así al momento de ver patrones extraños en la imagen el sistema mandara la señal adecuada para poder solventar de inmediato la situación que se pueda presentar.



Figura 32: Interfaz Sistema SCADA

Fuente: Victor Valente

CONCLUSIONES

Al culminar este trabajo de grado, se llegaron a las siguientes conclusiones:

- Se desarrolló un sistema Scada capaz de evaluar las áreas específicas indicando si hay movimiento, cambio de temperatura o si el espacio a monitorear se llena de humo, esto con la finalidad de controlar el espacio indicado ya sea cuando se esté utilizando en el horario académico o cuando se detecte irregularidades fuera de esos horarios académicos, con esta aplicación la Universidad José Antonio Páez puede monitorear todo los espacios que desee en cualquier momento, garantizando un ambiente educativo y seguro para los usuarios de las instalaciones, el sistema está configurado para conectarse con cualquier dispositivo a través de un puerto serial, esto permite al usuario de la aplicación poder adquirir los sensores sin importar la marca, modelo o el precio.
- Según la investigación que se realizó a través del instrumento de la observación directa, se pudo detectar la mayoría de los espacios de la universidad que poseen el riesgo de que se realicen actos indebidos, están ubicados en el edificio de estudios múltiples ya que es el más grande de la institución y gracias al sistema Scada se puede cubrir esos puntos ciegos de los niveles superiores permitiendo que el instituto pueda tener el control y saber el estatus del área en tiempo real, de esta forma ayuda a que la escases de personal de seguridad no afecte a la tranquilidad que la José Antonio Páez ofrece a los usuarios que disponen de las instalaciones
- Con los requerimientos del sistema se puede lograr de forma más precisa y organizada lo que se desea desarrollar, te permite trabajar modulo por modulo siguiendo con cada requerimiento funcional y de esa manera no se olvida lo que se tiene que realizar, de esta forma ahorras tiempo en pensar en que necesita tu sistema para funcionar mejor

- Utilizando el lenguaje de C# en el entorno de desarrollo de visual studio, se pudo notar que es una herramienta que posee una cantidad elevada de DLL que permite desarrollar una interfaz con todas las herramientas necesarias para demostrar de manera intuitiva el funcionamiento del sistema, también ayudo a crear una vista de un Scada diferente a las vistas habituales que tienden a ser con imágenes que son de poca calidad y en la mayoría de las veces sin una explicación previa llegan a ser confusa para los usuarios que la visualicen.
- Para el desarrollo del sistema Scada se realizaron los diagramas respectivos, esos ayudaron a no desviarme del camino a la meta, teniendo en cuenta la falta de conocimiento acerca de los sensores y su funcionamiento se pudo lograr la comunicación entre el simulador de arduino y la aplicación desarrollada en c#
- Este proyecto también demuestra que el estudio de la ingeniería es aplicar un poco de conocimiento en cualquier área y con un poco de interés podemos entrar en cualquier rama de la ingeniería y combinarlas para así construir un sistema que funcione con todas las características necesarias para trabajar en cualquier ambiente

RECOMENDACIONES

Al realizar las conclusiones de la solución al problema, se procedió a efectuar las siguientes recomendaciones para lograr la mejora del sistema con la finalidad de atender de mejor forma a las necesidades de los usuarios

- Al trabajar con 3 sensores se recomienda que donde se vaya a tener instalado el sistema sea en un ambiente en buena temperatura y con una computadora con buena capacidad de procesamiento, ya que el sistema estará realizando la comunicación con 3 equipos diferentes y al ejecutar los procesos al mismo tiempo puede que el tiempo de respuesta no sea el más efectivo o el adecuado si se quiere cumplir con todos los casos ya planteados anteriormente
- El sistema puede conectarse a cualquier dispositivo electrónico por puerto serial, pero se recomienda que ese equipo electrónico sea programable así al momento del sistema escuchar por un puerto la respuesta que reciba sean las más parecidas posible para una óptima funcionalidad de la aplicación.
- El sistema cuenta con un módulo de creación de perfiles, aquí se asignan todos los permisos que posee cada perfil para modificar los registros del sistema, se recomienda que antes de crear los perfiles se realice un estudio de cuáles son los límites de acceso y modificación para así crear los más óptimos posible en el sistema, del mismo modo cuenta con la creación de usuario donde se le tiene que asignar un perfil, aquí también requiere de un análisis de que actividades va a cumplir esta persona para así asignarle el perfil más óptimo para su cargo

- Realizar una actualización del software que permita a través del sensor de movimiento poder detectar el peso y la altura aproximado de los individuos que el sistema logre reconocer
- Realizar un sensor que detecte los componentes de los otros sensores así detecta cuando uno deja de funcionar para lograr evitar el retraso o los inconvenientes que pueda generar la falla imprevista

REFERENCIAS BIBLIOGRÁFICAS

- Andrew K. (2006). **Aforge.NET Framework librería de visión artificial e inteligencia artificial**. Obtenido en: <http://www.aforgenet.com/framework/>
- Arias F. (2012). **El proyecto de investigación: Introducción a la metodología científica**. 6a Edición Recuperado en: <https://ebevidencia.com/wp-content/uploads/2014/12/EL-PROYECTO-DE-INVESTIGACION%20C3%93N-6ta-Ed.-FIDIAS-G.-ARIAS.pdf>
- Castillo, L y Fuenmayor, R (2019) **Sistema scada (supervisión, control y adquisición de datos) para el control automático del encendido de los aires acondicionados de la universidad José Antonio Páez**. Trabajo de Grado para optar por los títulos de Ingeniería de Computación e Ingeniería Electrónica. San Diego-Venezuela
- David Cuartielles (2018) **Cofundador de Arduino, Página oficial** <https://www.arduino.cc/en/Main/Education>
- Ferretti C., Ángel L. (2012) **Actualización del sistema Scada del manipulador de tuberías del taladro de perforación HH-200 de la empresa petreven**. Trabajo de Grado para optar por el título de Ingeniero Electricista. Caracas-Venezuela
- González & Salazar (2008) **Introducción a la Metodología de la Investigación Científica** Primera edición electrónica. Recuperado de: <http://repositorio.espe.edu.ec/jspui/bitstream/21000/15424/1/Introduccion%20a%20la%20Metodologia%20de%20la%20investigacion%20cientifica.pdf>
- Gutiérrez, J.A (1994) Significado Económico - Social y Técnico de Automatización.
- Hurtado, J. (2010). **El proyecto de investigación**. Caracas: Editorial Quirón.
- Lucid Software Inc. (2019) **Diagramas UML online** generados en <https://www.lucidchart.com/pages/es/diagrama-uml>

- Raxy M. (2016). **Bases legales Sistema SCADA.** Recuperado en:
<https://independent.academia.edu/RaxyMonterola>
- Rojas, X y Correa, G (2015) **Diseño e implementación de un sistema de control y supervisión HMI para maquina barnizadora de la empresa el telégrafo.** Para optar por el título de ingeniería en electrónica Quito-Ecuador
- Stracuzzi S., Pestana F. (2010) **Metodología De La Investigación Cuantitativa.** 3ra Ed Recuperado en:
<https://es.calameo.com/books/000628576f51732890350>
- Sql Server Management Studio (2018) **Gestor de Base de datos extraído de**
<https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-2017>
- Tamayo y Tamayo (2003). **El Proceso de la Investigación Científica.** Editorial Limusa Noriega. 4ta Edición
- Villafranca D. (2002). **Metodología De La Investigación.** Mc Graw Hill, México (2003).
- VisualStudio. (2019) **Entorno de desarrollo de visual estudio,** extraído de
<https://visualstudio.microsoft.com/es/>