



UNIVERSIDAD
NACIONAL
DE LOJA

PFC-N-CIS-001



Área de la Energía, las Industrias y los Recursos Naturales No Renovables

CARRERA DE INGENIERIA EN SISTEMAS

“Sistema de visión artificial para el control de temperatura ambiente integrado con la plataforma arduino”

Tesis previa a la obtención del título de Ingeniero en Sistemas

Autores:

- Yandry René Ramírez Saritama
- Juan Pablo Guamán Rodríguez

Director:

- Ing. Luis Roberto Jácome Galarza Mg. Sc.

Autorizado por:

- Ing. Hernán Leonardo Torres Carrión, Mg, Sc.

Certificación del Director

Ing. Luis Roberto Jácome Galarza, Mg. Sc.

DOCENTE DE LA CARRERA DE INGENIERIA EN SISTEMAS, DEL ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES NO RENOVABLES DE LA UNIVERSIDAD NACIONAL DE LOJA.

CERTIFICA:

Haber asesorado y revisado detenida y minuciosamente durante todo su desarrollo, el Proyecto de Fin de Carrera titulado: **“SISTEMA DE VISIÓN ARTIFICIAL PARA EL CONTROL DE TEMPERATURA AMBIENTE INTEGRADO CON LA PLATAFORMA ARDUINO”**. Realizado por los postulantes Juan Pablo Guamán Rodríguez y Yandry René Ramírez Saritama.

Por lo tanto, autorizo proseguir los trámites legales pertinentes para su presentación y defensa.

Loja; Julio 2015



Ing. Luis Roberto Jácome Galarza, Mg. Sc.

DIRECTOR DEL PFC

Autoría

Nosotros, **YANDRY RENÉ RAMÍREZ SARITAMA** y **JUAN PABLO GUAMÁN RODRÍGUEZ**, declaramos ser autores del presente trabajo de tesis y eximimos expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos de posibles reclamos o acciones legales por el contenido de la misma.

Adicionalmente aceptamos y autorizamos a la Universidad Nacional de Loja, la publicación de nuestra tesis en el Repositorio Institucional – Biblioteca Virtual.

Firma:



Cédula: 1104387509

Fecha: 14 - XII - 2015

Firma:



Cédula: 1104904667

Fecha: 14 - XII - 2015

Carta de Autorización

Nosotros, **YANDRY RENÉ RAMÍREZ SARITAMA** y **JUAN PABLO GUAMÁN RODRÍGUEZ**, declaramos ser autores de la tesis titulada: **“SISTEMA DE VISIÓN ARTIFICIAL PARA EL CONTROL DE TEMPERATURA AMBIENTE INTEGRADO CON LA PLATAFORMA ARDUINO”**, como requisito para optar al grado de: **INGENIERO EN SISTEMAS**; autorizamos al Sistema Bibliotecario de la Universidad Nacional de Loja para que con fines académicos, muestre al mundo la producción intelectual de la Universidad, a través de la visibilidad de su contenido de la siguiente manera en el Repositorio Digital Institucional:

Los usuarios pueden consultar el contenido de este trabajo en el RDI, en las redes de información del país y del exterior, con las cuales tenga convenio la Universidad.

La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia de la tesis que realice un tercero.

Para constancia de esta autorización, en la ciudad de Loja, a los catorce días del mes de diciembre del dos mil quince.

Firma:



Autor: Yandry René Ramírez Saritama

Cédula: 1104387509,

Dirección: Loja, Av. Occidental de paso y Bolívar Bailón

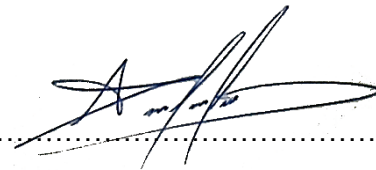
Teléfono: s/t

Celular: 0991740744

Correo Electrónico:

yampier07@gmail.com

Firma:



Autor: Juan Pablo Guamán Rodríguez

Cedula: 1104904667

Dirección: Loja, Nicaragua y Guinea Ecuatorial.

Teléfono: s/t

Celular: 0999241938

Correo Electrónico:

juanpa.desert@gmail.com

DATOS COMPLEMENTARIOS

Director de Tesis: Ing. Luis Roberto Jácome Galarza, Mg. Sc.

Tribunal de Grado: Ing. Alex Vinicio Padilla Encalada, Mg. Sc.

Ing. Franco Hernán Salcedo López, Mg. Sc

Ing. Ángel| José Ordoñez Mendieta, Mg. Sc

Agradecimiento

De forma especial hacemos extensivo el agradecimiento y sentimiento de estima a todos y cada uno de los profesores que formaron parte de nuestro crecimiento universitario; ya que, con sus enseñanzas nos brindaron las herramientas necesarias para culminar con éxito nuestros estudios; así mismo, a todos los Directivos de la Universidad Nacional de Loja, en especial a quienes forman parte de la carrera de Ingeniería en Sistemas. Finalmente y no menos grato nuestro sentimiento de gratitud y estima a nuestro Director de Tesis quien con su tiempo y conocimiento hizo posible que la presente tesis tenga el resultado final que entregamos para la obtención del Título de Ingeniero en Sistemas.

Yandry René Ramírez Saritama

Juan Pablo Guamán Rodríguez

Dedicatoria

Dedico este proyecto de titulación especialmente a mi madre Anabella Saritama por todo su esfuerzo y apoyo incondicional que me ha brindado ya que gracias a ella y a mi dedicación he sabido sobre salir a muchas dificultades, gracias madre por impulsar valores y principios fundamentales para desarrollarme como persona, para de esta manera apoyar al desarrollo de la familia y la sociedad en general.

Las palabras son extintas como para decirte cuán importante eres en mi vida no solo te mereces este logro exitoso sino que además hoy es el punto de partida para darte muchas más alegrías durante toda la vida.

Solo me queda decir gracias Dios por darme una madre excelente ya que todo lo que he logrado te lo debo a ti, gracias por ser mi madre.

Yandry René Ramírez Saritama

El presente trabajo de tesis lo dedico de forma especial en primer lugar a Dios que con su fortaleza forjó mi carácter para culminar con éxito mis estudios universitarios; mi especial cariño, respeto y dedicatoria a mis señores padres Herman Aníbal Guamán Reyes y Patricia Marlene Rodríguez Rodríguez, quienes con su ejemplo de vida, responsabilidad, amor y dedicación me brindaron la posibilidad de estudiar y con su apoyo incondicional logré que una de mis metas se hagan realidad.

Juan Pablo Guamán Rodríguez

1. Título

SISTEMA DE VISIÓN ARTIFICIAL PARA EL CONTROL DE TEMPERATURA
AMBIENTE INTEGRADO CON LA PLATAFORMA ARDUINO.

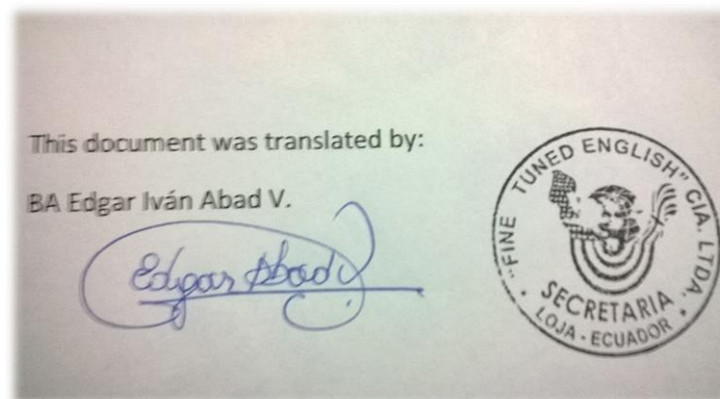
2. Resumen

El presente proyecto consistió en el diseño y puesta a prueba de un sistema capaz de detectar si se encuentran personas en una sala de conferencias con el objetivo de modificar la temperatura ambiente; estabilizando en un rango denominado temperatura confort. El sistema se construyó utilizando la placa Arduino mega 2560, el sensor de temperatura elegido fue el LM35 y el sensor de infrarrojo el PIC 26043SM, ya que el rango de voltaje de estos elementos está dentro de los 5v al cual trabaja la placa Arduino; además, de poseer gran documentación disponible. Conjuntamente un LED de infrarrojos para la comunicación con el aire acondicionado Split marca Indurama modelo ASI-24PL, el cual fue un préstamo de la bodega de la Universidad Nacional de Loja.

El sistema fue programado usando el lenguaje de programación JAVA, pues la librería JADE como la biblioteca OpenCV se implementan de manera muy sencilla en este lenguaje. Posteriormente se realizó la integración del sistema usando comunicación serial y finalmente se realizaron las pruebas para comprobar su correcto funcionamiento.

Abstract

This project involved the design and testing of a system capable of detect if there are people in a conference room in order to modify it is current ambient temperature. The system was built using Arduino Mega 2560, temperature sensor chosen was the LM35 and PIC 26043SM the infrared sensor for the voltage range of these elements It is within the 5v which works Arduino; besides, having great documentation available. Along an infrared LED for communication Split with air conditioning brand ASI-24PL Indurama model. The system was programmed using the JAVA programming language, since the JADE library as the library openCV implemented is very simply in this language. Later the system integration was performed using serial communication and finally the tests were conducted to verify proper operation.



Índice

Certificación del Director	1
Autoría	III
Carta de Autorización	IV
Agradecimiento	V
Dedicatoria	VI
1. Título	VII
2. Resumen	VIII
Summary	¡Error! Marcador no definido.
Índice	X
Índice de figuras	XIV
Índice de tablas	XVI
Índice de diagramas	XVII
3. Introducción	1
4. Revisión de Literatura	3
4.1 Temperatura	3
4.1.1 Definición	3
4.1.2 Normas de temperatura ambiente en una sala de conferencia	3
4.1.3 Confort térmico	4
4.2 Aire acondicionado	4
4.2.1 Definición	4
4.2.2 Características del Aire acondicionado Split	4
4.2.3 Identificación de las partes de la Unidad de Aire Acondicionado	5
4.2.4 Temperatura de Funcionamiento del Aire Acondicionado	7
4.2.5 Capacidad de los Aires Acondicionados de acuerdo al número de personas	8
4.3 Arduino	10
4.3.1 Definición	10

4.3.2	Comparativa de Arduino con otras placas	10
4.3.3	Comparativa entre las diferentes placas Arduino.....	11
4.3.4	Selección del tipo de placa Arduino a utilizar	12
4.3.5	Alimentación.....	13
4.3.6	Ventajas de Arduino	14
4.3.7	Librería IRremote.....	14
4.4	Sensores	15
4.4.1	Definición.....	15
4.4.2	Sensor de Infrarrojos	15
4.4.3	Diferentes tipos de sensores infrarrojos.....	15
4.4.4	Sensor de infrarrojo a utilizar	16
4.4.5	LED IR.....	17
4.4.6	Diferentes tipos de sensores de temperatura.....	17
4.4.7	Sensor de temperatura a utilizar	20
4.5	Cámara Web	20
4.5.1	Definición.....	20
4.5.2	Tipos de cámaras de video.....	20
4.5.3	Cámara a utilizar	21
4.6	OpenCV.....	22
4.6.1	Definición.....	22
4.6.2	Algoritmo de reconocimiento de Objetos	23
4.6.3	Algoritmo de Haar-like Features	24
4.6.4	Etapas del reconocimiento de objetos	25
4.7	JADE (Java Agent DEvelopment Framework)	27
4.7.1	Definición.....	27
4.7.2	Características.....	27
4.7.3	Especificaciones FIPA (Foundation for Intelligent Physical Agents).....	27
5.	Materiales y Métodos.....	28

5.1	Recursos Materiales	28
5.2	Técnicas de Observación	29
5.3	Metodología Prometheus.....	31
5.3.1	Fases de la Metodología Prometheus	31
6.	Resultados.....	33
6.1	Perspectiva del sistema	33
6.1.1	Requerimientos Funcionales	33
6.1.2	Requerimientos no Funcionales.....	34
6.2	Diagrama de bloques del sistema	35
6.3	Diseño del circuito con la placa Arduino	37
6.3.1	Recibir variación de temperatura desde Arduino con un LM35	37
6.3.2	Emitir señal infrarroja desde Arduino con un led de infrarrojo	37
6.3.3	Decodificar una señal de infrarrojo con Arduino.....	38
6.4	Diseño del sistema	42
6.4.1	Especificación del sistema.....	42
6.4.2	Diseño Arquitectónico.....	46
6.4.3	Desarrollo Detallado.	47
6.5	Diagramas de componentes	49
6.5.1	Diagrama de componentes integración del sistema.....	49
6.6	Diagramas de flujo.....	50
6.6.1	Diagrama de flujo captura de la imagen.....	50
6.6.2	Diagrama de flujo procesamiento de la imagen.	52
6.6.3	Diagrama de flujo comunicación Sistema con Arduino	54
6.6.4	Diagrama de flujo programación del Arduino	56
6.7	Interfaces de usuario	57
6.8	Plan de pruebas	60
6.8.1	Alcance.....	60
6.8.2	Requerimientos	60

6.8.3	Procedimiento.....	60
6.8.4	Pruebas de Caja Negra	61
6.8.5	Pruebas de Caja Blanca	64
6.8.6	Resultados de pruebas de caja blanca y caja negra	66
6.9	Pruebas Funcionales	67
6.10	Pruebas del Sistema.....	71
7.	Discusión.....	76
7.1	Desarrollo de la propuesta alternativa.....	78
7.1.1	Objetivo Específico 1: “Diseñar y programar el módulo de visión artificial, encargado de la detección y conteo de personas en una habitación.”	78
7.1.2	Objetivo Específico 2: “Diseñar e integrar un circuito con sensores y actuadores en la placa Arduino encargada de la detección de temperatura ambiente y activación del aire acondicionado.”	80
7.1.3	Objetivo Específico 3: “Integrar de módulo de visión artificial con el circuito en la placa Arduino.”	81
7.1.4	Objetivo Específico 4: “Realizar pruebas de control en una habitación a escala para probar su funcionalidad y fiabilidad.”	82
7.2	Valoración técnica económica ambiental	84
7.2.1	Valoración técnica económica	84
7.2.2	Valoración ambiental	87
8.	Conclusiones	88
9.	Recomendaciones	89
10.	Bibliografía	90
11.	Anexos	93
11.1	Glosario de Términos	93
11.2	Código arduino decodificación señales infrarrojas	94
11.3	Código arduino lectura de temperatura y envío de señales infrarrojas.....	97
11.4	Entrevista realizada a los hoteles de la ciudad de Loja.....	99

Índice de figuras

Figura 1: Partes del aire acondicionado.....	7
Figura 2: Placa Arduino Mega 2560.....	13
Figura 3: LED IR.....	17
Figura 4: Sensor de temperatura DS18B20.....	18
Figura 5: Sensor de temperatura TMP36.....	18
Figura 6: Sensor de temperatura LM35.....	19
Figura 7: Webcam Genius.....	21
Figura 8: Ubicación de la Cámara.....	22
Figura 9: Rectángulos adyacentes.....	24
Figura 10: Diseño circuito (LM35).....	37
Figura 11: Diseño circuito (LED Infrarrojo).....	38
Figura 12: Diseño del circuito (IR Receiver).....	39
Figura 13: Diseño del circuito (LM35, LED Infrarrojo y IR Receiver).....	40
Figura 14: Esquema del circuito (LM35, LED Infrarrojo y IR Receiver).....	41
Figura 15: PCB del circuito (LM35, LED Infrarrojo y IR Receiver).....	42
Figura 16: Descripción diagrama de componentes integración del sistema.....	49
Figura 17: Descripción diagrama de flujo captura de la imagen.....	51
Figura 18: Descripción diagrama de flujo procesamiento de la imagen.....	53
Figura 19: Descripción diagrama de flujo comunicación Sistema con Arduino.....	55
Figura 20: Descripción diagrama de flujo programación del Arduino.....	57
Figura 21: Interfaz principal, configuraciones del sistema.....	58
Figura 22: Interfaz ejecución del sistema.....	59
Figura 23: Parámetros de configuración.....	61
Figura 24: Testing de parámetros de configuración.....	62
Figura 25: Pantalla de Información.....	63
Figura 26: Creación de los agentes.....	64
Figura 27: Comunicación Arduino.....	65
Figura 28: Funcionamiento de los agentes.....	66
Figura 29: Escenario de Pruebas Funcionales.....	68
Figura 30: Webcam.....	68
Figura 31: Personas Detectadas.....	68
Figura 32: Datos del Servidor.....	69
Figura 33: Circuito en la placa Arduino.....	70

Figura 34: Temperatura fijada en el Aire Acondicionado.....	71
Figura 35: Inicio del sistema.	72
Figura 36: Guardar nuevas configuraciones del sistema.....	73
Figura 37: Datos procesados por el sistema.	74
Figura 38: Pantalla de información.....	74
Figura 39: Temperatura del Aire acondicionado.....	75
Figura 40: Filtro de perfil y de frente	79
Figura 41: Filtro parte superior cuerpo	79
Figura 42: Distancia filtro parte superior cuerpo.....	79
Figura 43: Distancia filtro de perfil y de frente	79
Figura 44: Monitor serial ID Arduino.....	81
Figura 45: Sala de prueba.....	82
Figura 46: Campo captura webcam	82
Figura 47: Prueba de funcionalidad y fiabilidad.....	83

Índice de tablas

Tabla 1: Características aire acondicionado split.....	5
Tabla 2: Temperatura del aire acondicionado.....	7
Tabla 3: Cálculo capacidad aire acondicionado.....	8
Tabla 4: Capacidad de aire acondicionado.....	9
Tabla 5: Comparativa microcontroladores	11
Tabla 6: Comparativa de placas Arduino.....	12
Tabla 7: Diferentes tipos de sensores infrarrojos.....	16
Tabla 8: Materiales físicos.....	28
Tabla 9: Servicios.....	29
Tabla 10: Requerimientos funcionales	34
Tabla 11: Requerimientos no funcionales.....	35
Tabla 12: Descripción caso de uso obtener temperatura.....	43
Tabla 13: Descripción caso de uso obtener número de Personas.....	44
Tabla 14: Descripción caso de uso Decodificar Señal	45
Tabla 15: Descripción caso de uso fijar temperatura	45
Tabla 16: Temperaturas agent_temp_cis	76
Tabla 17: Recursos humanos.....	84
Tabla 18: Recursos materiales.....	85
Tabla 19: Recursos técnicos y tecnológicos	86
Tabla 20: Presupuesto del proyecto	86

Índice de diagramas

Diagrama 1: Diagrama de bloques del sistema.....	35
Diagrama 2: Casos de uso del sistema.....	42
Diagrama 3: Especificación del sistema.....	46
Diagrama 4: Diseño Arquitectónico.....	47
Diagrama 5: Diseño detallado.....	48
Diagrama 6: Diagrama de componentes integración del sistema.....	49
Diagrama 7: Diagrama de flujo captura de la imagen.....	50
Diagrama 8: Diagrama de flujo procesamiento de la imagen.	52
Diagrama 9: Diagrama de flujo comunicación Sistema con Arduino	54
Diagrama 10: Diagrama de flujo programación del Arduino	56

3. Introducción

Hoy en día las empresas buscan automatizar cada vez más los procesos que llevan a cabo, siendo estos procesos un factor muy importante en la productividad de las mismas; ya que, hacerlos automáticamente resulta ser realmente útil y necesario.

Uno de estos procesos es el control de temperatura ambiente dentro de las salas de conferencias, el cual al no estar automatizado depende de una persona encargada y en la mayoría de los casos no capacitada en el uso del aire acondicionado, lo que ocasiona gastos innecesarios para la empresa.

El presente proyecto surge, tomando como punto de partida la automatización del proceso de variación de temperatura ambiente, haciendo uso de un aire acondicionado, el cual debe ser activado y modificada la temperatura a emitir, considerando si en la sala se encuentran o no personas, además de la temperatura ambiente de la misma para evitar los cambios bruscos en ésta.

La siguiente fase del proyecto considera la manipulación del aire acondicionado, lo cual se lo consigue con el hardware libre Arduino, el mismo que además de ser económico brinda una forma sencilla de comunicación, el que se aprovecha para la integración con el sistema multiagentes que ejecutará la acción de activación o desactivación del aire acondicionado.

La revisión literaria describe los elementos físicos necesarios para la implementación, así como también variables teóricas como temperatura ambiente y temperatura confort para el ser humano; así como, una breve descripción de OpenCV para el procesamiento de imágenes.

En materiales y métodos se expone el costo del proyecto así como también las técnicas y metodologías a usar para el desarrollo del sistema multiagentes, que contará con dos agentes, uno reactivo y uno deliberativo, los mismos que se encargarán: uno de informar la temperatura ambiente de la sala y de la detección de personas que se encuentran en ese instante; y, el otro de ejecutar las acciones necesarias para el cambio de temperatura del aire acondicionado respectivamente.

Los resultados presentan el desarrollo de cada una de las etapas que planteamos así como de la metodología Prometheus y los diagramas de componentes que se usaron

para la descripción de la integración de cada una de las partes que conformarán el sistema; además, diagramas de flujo para las pruebas de caja blanca y caja negra.

La discusión expone la implementación del sistema y como cada uno de los objetivos fueron cumplidos gracias a la ayuda de la librería OpenCV (Detección de personas) y JADE (Creación de sistemas multiagentes), además de la librería RXTX para la comunicación con la placa Arduino mediante el uso del puerto serial.

Finalmente se presentan las conclusiones y recomendaciones conforme a lo realizado durante el proceso de todo el proyecto investigativo.

4. Revisión de Literatura

4.1 Temperatura

4.1.1 Definición

La temperatura es una magnitud física que refleja la cantidad de calor, ya sea de un cuerpo, de un objeto o del ambiente. Dicha magnitud está vinculada a la noción de frío (menor temperatura) y caliente (mayor temperatura). [1]

La medición de la temperatura es una de las más comunes y puede ser una de las más importantes que se efectúan en los ambientes laborales, ya que de esto depende el máximo desarrollo y capacidad de productividad en la empresa.

4.1.2 Normas de temperatura ambiente en una sala de conferencia

El reglamento de seguridad y salud de los trabajadores y mejoramiento del medio ambiente del trabajo de Ecuador indica que, en general, se deben evitar las temperaturas y las humedades extremas, los cambios bruscos de temperatura, las corrientes de aire, y la radiación solar excesiva a través de los equipos de acondicionamiento.

Además, el reglamento antes mencionado describe sobre disposiciones mínimas de seguridad y salud en los lugares de trabajo el cual establece las condiciones mínimas ambientales que deben reunir los lugares de trabajo.

Como principio general se establece que el ambiente de trabajo no debe suponer un riesgo para la seguridad y la salud de los trabajadores, y en la medida de lo posible, se debe evitar que constituya una fuente de incomodidad o molestia.

El reglamento de seguridad y salud de los trabajadores y mejoramiento del medio ambiente del trabajo de Ecuador establece que en los locales de trabajo cerrados deberán cumplirse las siguientes condiciones:

Temperatura: entre 17° C y 27° C, si se realizan trabajos sedentarios o entre 14°C y 25°C, si son trabajos ligeros. [2]

4.1.3 Confort térmico

El confort térmico se define en la Norma ISO 7730 como " Esa condición de mente en la que se expresa la satisfacción con el ambiente térmico". Una definición en que la mayoría de las personas puede estar de acuerdo, pero también es una definición que no se traslada fácilmente a parámetros físicos.

De la misma manera, se denomina Confort Térmico, cuando las condiciones de temperatura, humedad y movimientos del aire son agradables, confortables en referencia a actividad que desarrollan, es decir, las personas no experimentan sensación de calor, ni de frío. [3]

4.2 Aire acondicionado

4.2.1 Definición

Entendemos por aire acondicionado al sistema de refrigeración del aire, que se utiliza de modo doméstico para refrescar los ambientes, cuando las temperaturas del ambiente son muy altas y/o calurosas. El aire acondicionado, si bien hace referencia al aire en sí, es un aparato que se instala en casas, locales y demás espacios cerrados con el objetivo de proveer de aire fresco que se renueva permanentemente. [4]

4.2.2 Características del Aire acondicionado Split

El aire acondicionado que se va a utilizar cuenta con las siguientes características:

Modelo	ASI-24PL
Capacidad/Enfriamiento	24000Btu/h
Refrigerante	R22/950g
Voltaje	220V

Frecuencia	60Hz
Corriente Máxima	15.0 A
Entrada Máxima	3000 W
Potencia Nominal	2750 W

Tabla 1: Características aire acondicionado split

4.2.3 Identificación de las partes de la Unidad de Aire Acondicionado

El aire acondicionado está formado por dos unidades interior y exterior respectivamente, como se muestra en la figura 1, a continuación se describe cada una de ellas.

- 1) Marco de panel delantero.-** Comprende la cubierta de la unidad interior del aire acondicionado.
- 2) Panel delantero.-** Se encuentra en la unidad interior del aire acondicionado y sirve para alojar algunos componentes como los sensores, rejillas, filtros, receptor de señales etc.
- 3) Filtro de aire.-** Se encuentra debajo del panel y sirve para eliminar partículas sólidas como por ejemplo polvo, polen y bacterias del aire.
- 4) Rejilla de aire Horizontal.-** Sirve para propagar el aire a través de unas ranuras situadas en el panel delantero del aire acondicionado.
- 5) Lámina de caudal del aire vertical.-** Sirve para controlar la humedad relativa en la habitación, esta lamina debe ser colocada a un ángulo máximo en vertical con el suelo.
- 6) Sensor de temperatura.-** Es un dispositivo eléctrico o mecánico capacitado para detectar acciones o estímulos en valores medibles para las magnitudes eléctricas.

7) Panel de visualización.- Es un pequeño espacio donde se muestra los valores de configuración del aire acondicionado.

8) Receptor de señales infrarrojas.- Son pequeños microchips con una célula fotoeléctrica que están sintonizados para escuchar a la luz infrarroja.

9) Mando a distancia.- Es un dispositivo electrónico usado para realizar una operación remota (o telemando) sobre una máquina.

10) Manguera de desagüe.- Es un tubo hueco flexible diseñado para transportar fluidos, está conectado de la parte interna a la parte externa del aire acondicionado.

11) Cable conexión.- Está disponible para conectar el equipo a corriente eléctrica.

12) Válvula de parada.- Es un componente clave en sistemas de refrigeración y aire acondicionado, que tiene la capacidad de generar la caída de presión necesaria entre el condensador y el evaporador en el sistema. [5]

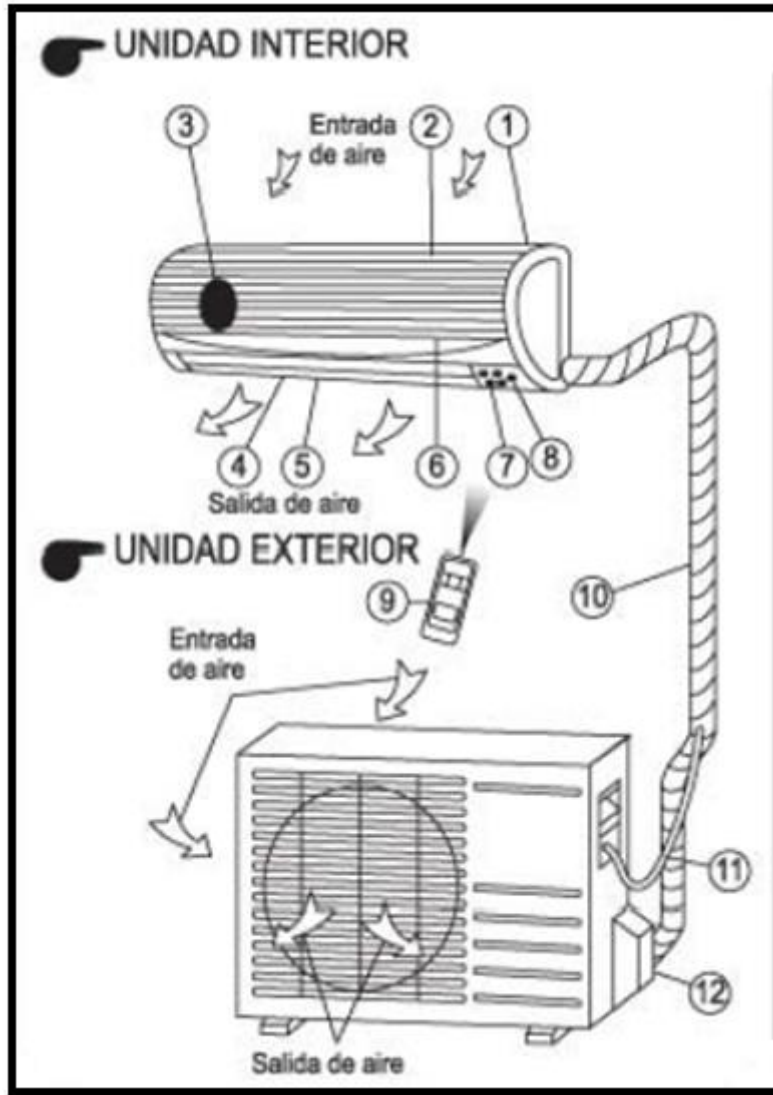


Figura 1: Partes del aire acondicionado

4.2.4 Temperatura de Funcionamiento del Aire Acondicionado

La temperatura de funcionamiento del aire acondicionado a utilizar se detalla a continuación en la tabla 2. [5]

Temperatura/Modo	Refrigeración	Calefacción
Temperatura Ambiente	17°C – 30°C	0°C – 30°C

Tabla 2: Temperatura del aire acondicionado

4.2.5 Capacidad de los Aires Acondicionados de acuerdo al número de personas

Para conocer la capacidad del aire acondicionado que se debe usar para determinado lugar se deben tener en cuenta varios factores como lo son:

- Número de personas.
- Número de equipos que se encuentran en el lugar que disipen calor (computadores televisores, electrodomésticos en general).
- Ventilación (posibles fugas de aire que pueda haber como ventanas, puertas)
- Área del lugar en metros cúbicos (m³) Largo X Ancho X Alto.

Para realizar el cálculo de capacidad se debe tener en cuenta lo siguiente:

12.000 BTU/h	1 TON. DE REFRIGERACION
1 BTU/h	0,252164 Kcal/h
1KW	860 Kcal/h
1HP	642 Kcal/h

Tabla 3: Cálculo capacidad aire acondicionado

Cálculo de capacidad

El cálculo de la capacidad se la realiza de acuerdo a la fórmula obtenida del libro de "Aires Acondicionados y Calefacción" cuyo autor es el Ing. Nestor Quadri. [6]

$$C = 230 \times V + (\# \text{ PyE} \times 476)$$

Dónde:

230 = Factor calculado para América Latina "Temp máxima de 40°C"

V = Volumen del AREA donde se instalará el equipo, Largo x Alto x Ancho en m³

PyE = # de personas + Equipos instalados en el área

476 = Factores de ganancia y pérdida aportados por cada persona y/o electrodoméstico

Por ejemplo, para instalar un aire acondicionado en un cuarto de 2,8 m de ancho por 3,5 m de largo y 2 m de altura, donde generalmente van a estar 20 personas.

$$V = 2,8 \times 3,5 \times 2 = 19.6 \text{ m}^3$$

$$\# \text{ PyE} = 20$$

$$C = (230 \times 19.6) + (20 \times 476)$$

$$C = 4508 + 9520$$

$$C = 14028 \text{ BTU}$$

La capacidad no es exactamente la que viene el aire acondicionado pero se compra un aire acondicionado que tenga la capacidad suficiente para el número adecuado de BTU que se requiere, en este caso se necesitará un equipo acondicionador de aire de 24000 BTU. [7]

Número de personas	Capacidad requerida	Capacidad de diferentes Aires Acondicionados
10	9268 Btu	12000 Btu
20	14028 Btu	24000 Btu
50	31255 Btu	36000 Btu
100	52108 Btu	60000 Btu

Tabla 4: Capacidad de aire acondicionado

4.3 Arduino

4.3.1 Definición

Arduino es una plataforma de electrónica abierta para la creación de prototipos basados en software y hardware abierto. Arduino puede tomar información del entorno a través de sus pines de entrada de toda una gama de sensores y puede interactuar con todo aquello que le rodea controlando luces, motores y otros actuadores. El microcontrolador en la placa Arduino se programa mediante el lenguaje de programación Arduino (basado en Wiring) y el entorno de desarrollo Arduino (basado en Processing). [8]

4.3.2 Comparativa de Arduino con otras placas

Hay muchos otros microcontroladores y plataformas de microcontroladores disponibles para computación. De tal manera que se ha realizado una tabla comparativa entre los siguientes más conocidos: Basic Stamp, Propeller Tessel, Arduino. Estos ofrecen funcionalidades similares, todas estas herramientas toman los detalles frágiles de programación de microcontroladores y se envuelve en un paquete fácil de usar. Arduino también simplifica el proceso de trabajar con microcontroladores, pero ofrece algunas ventajas para los profesores, estudiantes y aficionados interesados sobre este microcontrolador el cual lo detallaremos más adelante. [9]

CARACTERISTICAS	ARDUINO	BASIC STAMP	PROPELLER	TESSEL
Bajo Costo	sí	sí	sí	sí
Fácil disponibilidad	sí	sí	sí	sí
Uso educativo	sí	sí	sí	sí
Programación sencilla	sí	sí	no	no

Uso en Ingeniería	sí	sí	sí	sí
Multiplataforma	sí	sí	sí	sí
Menor consumo de energía	sí	no	sí	no
Facilidad de uso	sí	sí	no	no
Multinucleo	no	no	sí	no
Código abierto	sí	sí	sí	sí

Tabla 5: Comparativa microcontroladores

4.3.3 Comparativa entre las diferentes placas Arduino

Característica de Arduino	UNO	Mega 2560	Leonardo	DUE
Tipo de microcontrolador	ATmega 328	ATmega 2560	ATmega 32u4	AT91SAM3X8E
Velocidad de reloj	16 MHz	16 MHz	16 MHz	84 MHz
Pines digitales de E/S	14	54	20	54
Pines Entradas analógicas	6	16	12	12
Pines Salidas analógicas	0	0	0	2 (DAC)

Memoria de programa (Flash Memory)	32 Kb	256 Kb	32 Kb	512 Kb
Memoria de datos (SRAM)	2 Kb	8 Kb	2.5 Kb	96 Kb
Memoria auxiliar (EEPROM)	1 Kb	4 Kb	1 Kb	0 Kb
Voltaje operativo	5V	5V	5V	3.3V
Voltaje de entrada	7-12V	7-12V	7-12V	7-12V
Pin de corriente DC de 3.3V	50 mA	50 mA	50 mA	800 mA
Tipo de USB	Estándar	Mini	Estándar	Mini

Tabla 6: Comparativa de placas Arduino

4.3.4 Selección del tipo de placa Arduino a utilizar

La placa Arduino que se va a utilizar es el Arduino Mega 2560 (Figura 2) aunque el Arduino Uno es suficiente para el desarrollo del proyecto, ya que, nos provee de los pines tanto analógicos como digitales necesarios para la ubicación de los sensores LM35, el sensor de infrarrojos y el LED IR, la placa puede ser alimentada automáticamente mediante el puerto USB o con una fuente externa de poder con un voltaje de 5V, además se puede programar de una manera sencilla utilizando el

lenguaje propio de Arduino junto con la interfaz Arduino IDE de tal manera que se nos facilita el uso de esta placa inclusive el costo es moderado asumiendo que todo el presupuesto va por cuenta de los investigadores.

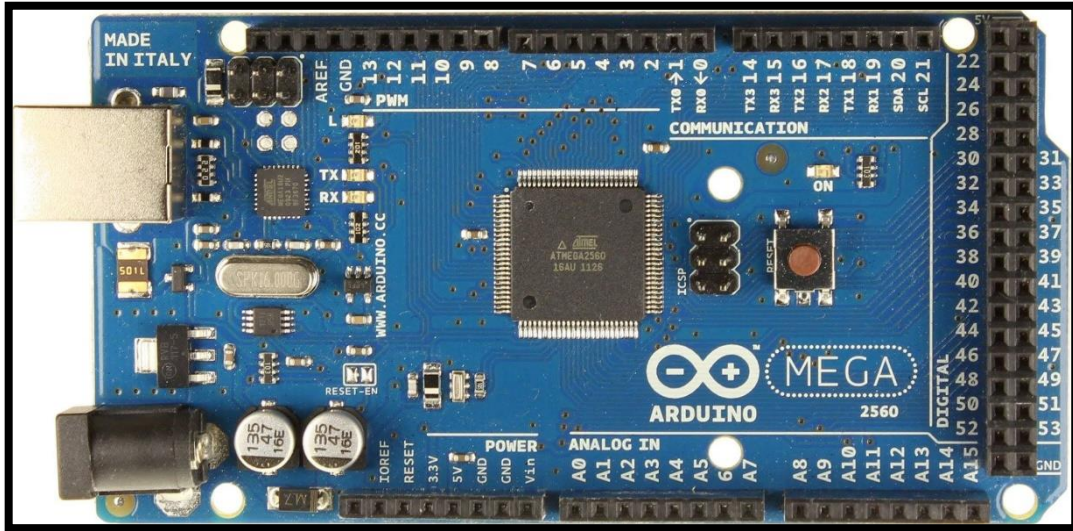


Figura 2: Placa Arduino Mega 2560

4.3.5 Alimentación

Arduino Mega puede ser alimentado mediante el puerto USB o con una fuente externa de poder. La alimentación es seleccionada de manera automática.

Cuando se trabaja con una fuente externa de poder se debe utilizar un convertidor AC/DC (Corriente Alterna /Corriente Directa) y regular dicho voltaje en el rango operativo de la placa. De igual manera se puede alimentar el micro mediante el uso de baterías. Preferiblemente el voltaje debe estar en el rango de los 7V hasta los 12V.

Arduino Mega posee algunos pines para la alimentación del circuito aparte del adaptador para la alimentación:

- **VIN:** A través de este pin es posible proporcionar alimentación a la placa.
- **5V:** Podemos obtener un voltaje de 5V y una corriente de 40mA desde este pin.
- **3.3V:** Podemos obtener un voltaje de 3.3V y una corriente de 50mA desde este pin.

- **GND:** El ground (0V) de la placa (o tierra).

Arduino puede ser programado de una manera muy fácil utilizando el lenguaje propio de Arduino junto con la interfaz Arduino IDE.

4.3.6 Ventajas de Arduino

Factible: Las placas Arduino son más accesibles y factibles comparadas con otras plataformas de microprocesadores

Multiplataforma: El software Arduino funciona en los diferentes sistemas operativos como Windows, Macintosh y Linux. La mayoría de los entornos para microprocesadores están limitados para Windows.

Ambiente de programación sencillo y directo: Arduino está basado en el entorno de programación de Processing con el que el usuario aprenderá a programar y se familiariza con el dominio del desarrollo de Arduino.

Software ampliable y de código abierto: El software de Arduino está publicado bajo una licencia libre y preparada para ser distribuido y ampliado por desarrolladores experimentados. [8]

4.3.7 Librería Irremote

La librería IrRemote decodifica la salida de muchos mandos a distancia por infrarrojos como: Sony, NEC, RC5 y RC6. Sin embargo, muchos mandos a distancia utilizan otros protocolos los cuales esta librería nos facilita la decodificación de igual manera que con los protocolos conocidos.

La librería IrRemote registra la duración de cada pulso enviado por el control remoto. Cada tecla del mando a distancia corresponde a un valor de código particular, que se convierte en una secuencia particular de pulsos. De tal modo que se puede determinar el valor de código, y por lo tanto la tecla pulsada. Sin embargo, para muchas aplicaciones no importa realmente lo que el valor del código original es, siempre y cuando se puede distinguir de forma única cada tecla. Por lo tanto, si se puede convertir cada secuencia única de pulsos en un valor único, entonces este valor indicará la tecla deseada. [10]

Con el uso de esta librería se obtuvo los códigos del mando a distancia del aire acondicionado y ya con ello se puede enviar las señales desde el LED de infrarrojos conectado a la placa Arduino hacia el aire acondicionado.

4.4 Sensores

4.4.1 Definición

Es un dispositivo eléctrico o mecánico capacitado para detectar acciones o estímulos en valores medibles para las magnitudes eléctricas.

Al ser medido un fenómeno físico este es captado por el sensor en el cual muestra en su salida una señal eléctrica dependiente del valor de la variable física. [11]

4.4.2 Sensor de Infrarrojos

Son pequeños microchips con una célula fotoeléctrica que están sintonizados para escuchar a la luz infrarroja. Casi siempre se utiliza para la detección de control remoto, cada televisor, reproductor de DVD y aires acondicionados tiene uno de estos en la parte delantera para escuchar la señal IR. En el interior del mando a distancia se encuentra un LED IR, que emite pulsos IR para decirle al aire acondicionado para activar, desactivar o cambiar de temperatura. La luz infrarroja no es visible para el ojo humano, lo que significa que se necesita un poco más de trabajo para probar una configuración. [12]

4.4.3 Diferentes tipos de sensores infrarrojos

Elemento	Pic 26043SM	Alta velocidad	NPN de silicio	Silicio
Sensor				
Voltaje	4,5 – 5,5V	12,5 V	32 V	32 V

Temperatura soldadura	260°	245°	260°	260°
Angulo medio	45°	60°	12°	65°
Potencia de disipación	150mW	150mW	159 mW	150mW

Tabla 7: Diferentes tipos de sensores infrarrojos

4.4.4 Sensor de infrarrojo a utilizar

El sensor de infrarrojo que se va a utilizar es el PIC 26043SM ya que está dentro del rango permitido de energía que vamos a utilizar con la placa Arduino de 4,5 a 5,5 Voltios con una frecuencia de 38 KHz, además cuenta con un ángulo medio de 45° suficiente como para establecer conexión con el aire acondicionado ya que la placa Arduino se la va a colocar junto al equipo de aire acondicionado por lo tanto no existirá ninguna interposición sobre la luz infrarroja y el receptor.

Ventajas:

- Requerimientos de bajo voltaje por lo tanto es ideal para mandos a distancia genéricos.
- Circuitería simple: no requiere hardware especial, puede ser incorporado en el circuito integrado de un producto.

Desventajas:

- Se bloquea la transmisión con materiales comunes: personas, paredes, plantas, etc.
- Corto alcance: la performance cae con distancias más largas.
- Velocidad: la transmisión de datos es más baja que la típica transmisión cableada.

4.4.5 LED IR

El LED IR sirve para enviar señales infrarrojas las cuales viajan por el medio y llegan al dispositivo que lo recibe por medio de un receptor de señales infrarrojas que decodifica la señal y toma una decisión según la señal que llega.

Tiene un ánodo y cátodo, (Ver Figura 3) que hace la diferencia entre negativo y positivo respectivamente. Esta diferencia se observa en las patitas que tiene los led, donde una es más larga que la otra, la que es más larga debe ir al positivo (en este caso a un pin digital de la Arduino) y la más corta a negativo, o sea a GND del Arduino. [13]



Figura 3: LED IR

4.4.6 Diferentes tipos de sensores de temperatura

A continuación procedemos a evaluar los principales sensores de temperatura que son totalmente compatibles con las distintas versiones de la plataforma Arduino.

Sensor de temperatura digital “ds18b20”

Comenzamos por el primero de los dos sensores que mantienen el mismo diseño, concretamente en su versión digital, el modelo DS18B20. En la figura 4 vemos una imagen del diseño que presenta este tipo de sensor, y podemos hacernos una idea de su reducido tamaño.

problema, pues aun cuando tenemos conectados todos los elementos que precisa nuestro proyecto, seguimos teniendo libres varias entradas/salidas, tanto analógicas como digitales.

Su precio está por debajo de los \$2 dólares americanos, por lo que resulta un sensor bastante interesante, estima el valor de temperatura con notable precisión y detecta variaciones en la misma de manera casi instantánea. [14]

Sensor LM35

El LM35 es un sensor de temperatura muy preciso que no requiere ninguna calibración externa o ajuste para proporcionar una precisión típica de $\pm 1.4^\circ \text{C}$ a temperatura ambiente y $\pm 3.4^\circ \text{C}$ a lo largo de su rango de temperatura -55° a 100°C . (Ver Figura 6)

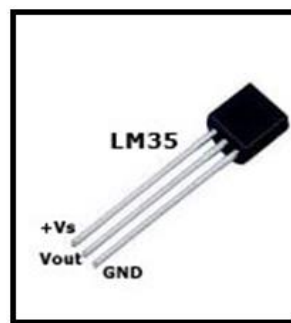


Figura 6: Sensor de temperatura LM35

Sus características más relevantes son:

- Está calibrado directamente en grados Celsius.
- La tensión de salida es proporcional a la temperatura.
- Tiene una precisión garantizada de 0.5°C a 25°C .
- Opera entre 4 y 30 voltios de alimentación.
- Baja impedancia de salida.
- Baja corriente de alimentación ($60\mu\text{A}$).
- Bajo costo. [15]

4.4.7 Sensor de temperatura a utilizar

El sensor de temperatura a utilizar es el sensor LM35, es un dispositivo económico, además cumple con los rangos de medición de temperatura que se van a manejar. Este dispositivo es diseñado por la empresa Texas Instruments, lo que garantiza que el producto esté disponible, económico y se ajusta a las necesidades del proyecto.

Se escogió este sensor de temperatura LM35 por su rango de temperatura tolerable entre los rangos de -50°C a 100°C que es más que suficiente para realizar las pruebas correspondientes, además por su precisión de respuesta con la que trabaja.

4.5 Cámara Web

4.5.1 Definición

Es un dispositivo que se conecta al puerto USB de la computadora, y así permite captar video y tomar fotos digitales con resolución no tan óptima, por lo que no ofrece una gran calidad de gráficos a diferencia de una cámara fotográfica digital, videocámara digital o un teléfono celular moderno. Son muy utilizadas para conversaciones vía Internet y hacer más personalizada la charla, así como también para actividades de vigilancia.

4.5.2 Tipos de cámaras de video

Las cámaras de vídeo son dispositivos cuyo propósito principal es registrar imágenes móviles. Entre ellas se destacan:

Cámaras profesionales de video: Tales como las usadas en la producción de televisión y de cine: Estas cámaras tienen sensores múltiples de color para mejorar la resolución (sensores rgb-rojo, verde y azul) y para mejorar la gama de colores.

Cámaras de video para aficionados: En ésta categoría se encuentran todas aquellas cámaras que graban directamente video a un dispositivo de almacenamiento de memoria. Usualmente tienen un micrófono y una pantalla LCD para supervisar la filmación.

Cámaras web: Son cámaras digitales diseñadas para funcionar conectadas directamente a una computadora. Generalmente son utilizadas para video

conferencias o para grabaciones de video; algunos modelos incluyen micro imágenes se adquieren en forma continua directamente a un formato digital de video.

Cámaras digitales compactas: Las digicams (cámaras fotográficas digitales estándar) abarcan la mayoría de las cámaras fotográficas digitales. Se caracterizan por ser bastante sencillas de operar, además de brindar funciones automáticas para el enfoque (autofoco) y el manejo de la iluminación. Debido a su reducido tamaño cuentan con distancias focales muy cortas con lo que se produce una muy amplia profundidad de campo, de modo que los objetos en diferentes distancias pueden ser enfocados al mismo tiempo. A veces esta cualidad puede provocar un efecto no deseado en la fotografía. [16]

4.5.3 Cámara a utilizar

La cámara web que se va a utilizar es VGA de 640 X 480 píxeles (Ver figura 7) ya que es suficiente para reconocer objetos con un alcance aproximado de 10 metros, distancia considerada dentro de la sala de conferencias disponible para las respectivas pruebas, permitiendo de esta manera obtener el reconocimiento de personas en tiempo real dentro de la sala de conferencias.



Figura 7: Webcam Genius.

La ubicación de la cámara tendrá que ser apropiada entendiéndose por apropiada la consideración de la altura y ángulo de visión respectivamente. (Ver figura 8)

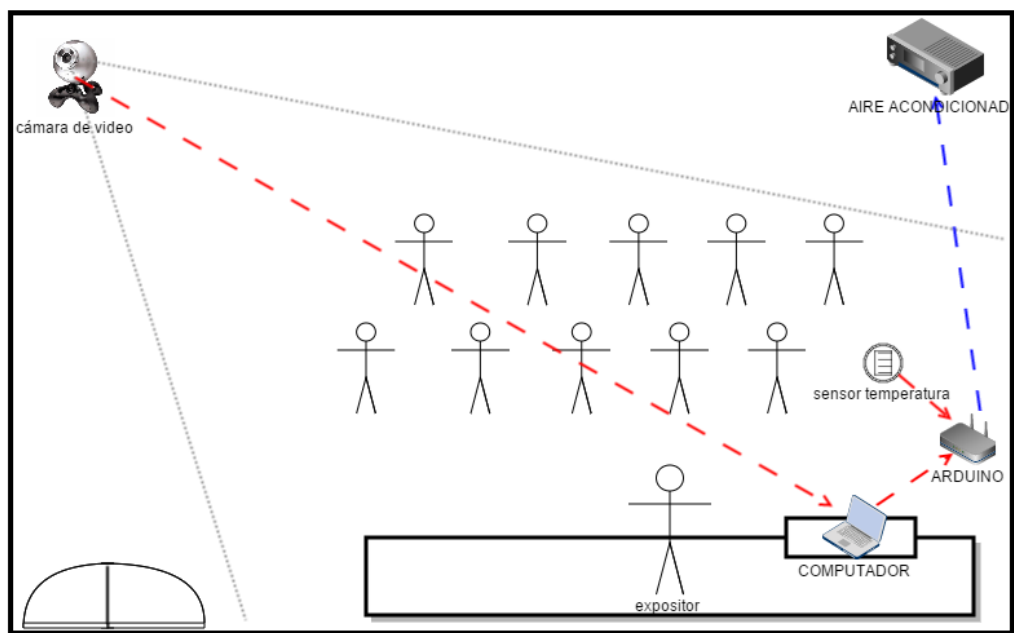


Figura 8: Ubicación de la Cámara.

Características

- Tienen la lente giratoria de hasta 360° horizontales, una base adaptable a la superficie, e incluso micrófono integrado.
- Pueden tomar fotos al instante con baja resolución.
- Su diseño es muy específico para aplicaciones de entretenimiento y en algunos casos como cámara de vigilancia. [17]

4.6 OpenCV

4.6.1 Definición

OpenCV (Open Source Computer Vision Library) es una librería de visión computacional desarrollada bajo licencia de código libre, está programada en los lenguajes C y C++ y es independiente del sistema operativo, del hardware y de la interfaz de ventanas; por lo tanto, puede ser instalada en sistemas Linux, Mac y Windows. También se planea su integración con los lenguajes Python, Ruby, Matlab entre otros.

Su desarrollo principalmente fue enfocado para dar una mayor eficiencia y optimización en uso de aplicaciones en tiempo real, esto sobre todo gracias a que está programado bajo lenguaje C optimizado que permite el uso de procesadores de múltiples núcleos.

Otra característica con la cual se desarrolló es que se puede integrar junto con las librerías IPP (Integrated Performance Primitives) de Intel para mejor eficiencia. Es una API tanto de bajo como de alto nivel y permite una manipulación genérica tanto a imágenes como a video.

Entre las 500 funciones integradas que tiene, se pueden desarrollar aplicaciones para las áreas de procesamiento de imágenes, imágenes médicas, seguridad, interfaces de usuarios, visión estéreo y como el procesamiento de imágenes la mayoría de las veces se liga con el área de la mecánica, en especial la robótica, incluye una librería de propósito general para aprendizaje de máquinas, enfocada sobre todo para reconocimiento estadístico de patrones.

Con las funciones incluidas en la librería, se puede hacer manipulación de los datos de imágenes, entrada y salida de imágenes y video, manipulación de vectores y matrices, rutinas de álgebra lineal, procesamiento de imágenes básico (filtros, detección de bordes, región de interés, interpolación, conversión a diferentes escalas de colores, histogramas), calibración de cámara, análisis de movimiento, reconocimiento de objetos y desarrollo de interfaces de usuario (GUI) simples. [18]

Su desarrollo empezó como una iniciativa del departamento de investigaciones de Intel, se buscaba una forma en que se desarrollarían aplicaciones con alto costo computacional. Se hizo un sondeo en varias universidades y se recopiló funciones y frameworks desarrollados por dichas universidades, entre ellas el MIT. Una vez que se tuvieron todas estas aplicaciones, se envió a la división de Intel en Rusia, en donde Vadim Pisarevsky manipuló, codificó, optimizó y programó la mayor parte de OpenCV, con la ayuda de Víctor Eruhimov y Valery Kuriakin pudo desarrollar la infraestructura y seguir adelante con el proyecto. [19]

4.6.2 Algoritmo de reconocimiento de Objetos

En la actualidad el reconocimiento y detección de objetos es una parte fundamental dentro del procesado de imagen y los sistemas de visión artificial. Por citar un ejemplo

las cámaras fotográficas y dispositivos móviles incluyen la detección de objetos lo cual hoy en día es muy común encontrarse con este tipo de software.

Específicamente, la detección de objetos se basa en la detección de una serie de características que identifiquen a los objetos con el menor error posible.

4.6.3 Algoritmo de Haar-like Features

Una de las técnicas típicas de detección de objetos está basada en los algoritmos propuestos por Paul Viola y Michael Jones. Estos algoritmos se basan en la extracción de características parecidas a las Integrales de Haar. Estas integrales consisten en, considerar regiones rectangulares adyacentes (Ver figura 9) en una ventana de detección, y sumar la intensidad de los píxeles en cada región, calculando después la diferencia entre estas sumas. Esta suma permite clasificar distintas secciones de la imagen. [20]

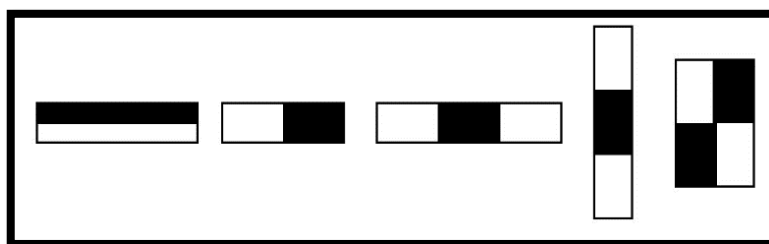


Figura 9: Rectángulos adyacentes

Si estamos tratando de detectar dónde están los ojos en una cara, una característica es que los ojos son más oscuros que las mejillas. En ese caso, se podría aplicar una región rectangular como la mostrada para detectar ojos. Así pues, una característica de Haar (o Haar-like features) común para la detección de caras es un conjunto de dos rectángulos adyacentes, uno debajo de otro, que coincidirá con la región de los ojos y mejillas. De esta forma se definen una serie de características que permiten detectar de una forma rápida y con una gran probabilidad de acierto.

Una vez comentadas las características de Haar, el algoritmo de detección en sí tiene dos etapas. Primero una etapa de entrenamiento, en la que a una serie de filtros en cascada se les pasan unos patrones positivos que coinciden con el objeto buscado y negativos que no tienen el objeto de forma que el sistema aprenda y se forme un modelo de las características de Haar del objeto a detectar. Cada una de estas

características por separado llamadas características débiles, no es suficiente para determinar si la imagen presentada corresponde o no con el objeto buscado. Pero una cascada de todas las características que debe cumplir el objeto sí son suficientes para poder discernir si la imagen corresponde o no con el objeto buscado.

La ventaja de este tipo de detectores es que las características de Haar son rápidas de calcular, y solo se calculan todas las características en el caso de que la región de la imagen en cuestión pase todos los clasificadores débiles. En el momento en que la imagen no pasa el umbral de uno de los clasificadores débiles se descarta, ahorrando el cálculo del resto de clasificadores.

Una vez se ha explicado brevemente cómo funciona la detección de objetos basada en características de Haar, se explicará la clase que ofrece OpenCV para detección de objetos, que es la clase `CascadeClassifier`. La clase `CascadeClassifier` tiene dos funciones principales que son las que permiten cargar las características de Haar del objeto a detectar, y otra para detectar propiamente dicha. Concretamente, la función que permite cargar las características de Haar es la función `load`, que necesita como parámetro de entrada el fichero xml que contiene la información generada en un entrenamiento previo.

OpenCV viene ya con unos ficheros entrenados con miles de patrones para detección de caras, manos, ojos, personas, entre otros con lo que no hace falta realizar un entrenamiento, sino tan solo usar los ficheros que ya trae OpenCV. Una vez cargado este fichero, lo siguiente es detectar en una imagen los objetos que haya, para lo cual se usa la función `detectMultiScale`. Esta función necesita como parámetros de entrada la imagen en la que se van a buscar, una variable en la que se almacenará la lista de objetos detectados, el tamaño al que se escalará la imagen, el número de vecinos que debe tener cada rectángulo de las características de Haar para tenerlo en consideración, un parámetro que no se usa en la versión actual de `detectMultiScale` y el tamaño mínimo posible de los objetos a detectar. [21]

4.6.4 Etapas del reconocimiento de objetos

Adquisición

La adquisición de las imágenes de entrada se realiza mediante cualquier dispositivo capaz de tomar imágenes como, por ejemplo, una cámara fotográfica, una cámara de

video vigilancia o la cámara integrada de un dispositivo móvil. Como con una imagen basta para comenzar el proceso, el tiempo de adquisición es muy bajo y no requiere de supervisión.

Detección

La etapa de detección en los sistemas de reconocimiento de objetos es crítica puesto que el resto de etapas se verán afectadas si no se ha realizado una detección y localización correctas. La detección está formada por dos partes:

Detección de la región de interés.

La detección de las regiones de interés en una imagen se realiza mediante los denominados Haar-like features, adaptados por Viola y Jones a partir del uso de Haar wavelets. Este sistema considera regiones rectangulares en una ventana de detección, suma las intensidades de los píxeles en cada región y calcula la diferencia entre estas sumas. La diferencia es usada para clasificar subsecciones de la imagen. La ventaja de este sistema desarrollado por Viola y Jones es la rapidez de cálculo, haciendo posible la detección de objetos en tiempo real.

Extracción de Características

La extracción de características es una de las etapas de la cual depende el desempeño del sistema de reconocimiento de objetos. El objetivo principal de esta etapa es extraer la información más discriminante de un objeto, eliminando la información innecesaria. Como extracción de componentes básicos del objeto, relaciones de distancia entre puntos específicos del objeto.

Reconocimiento

La última etapa, en un sistema de reconocimiento de objetos es la etapa de reconocimiento. Consiste en alimentar al sistema con imágenes de objetos diferentes a las utilizadas durante el entrenamiento, esperando obtener como resultado, alguna forma de codificación que nos permita saber de qué objeto se trata. [22]

4.7 JADE (Java Agent DEvelopment Framework)

4.7.1 Definición

JADE es marco de software desarrollado por Telecom Italy Lab (TILAB), antes llamado CSELT, para el desarrollo de aplicaciones distribuidas multi-agente que cumplen con las especificaciones FIPA para la interoperabilidad de sistemas multi-agente. [23]

4.7.2 Características

Las siguientes características son algunas de las más importantes que nos proporciona JADE

- Entorno de desarrollo para la creación de aplicaciones basadas en agentes.
- Entorno de ejecución para que los agentes vivan y se comuniquen.
- Arquitectura peer-to-peer (P2P)
- Interoperabilidad: especificaciones FIPA
- Cada agente se implementa como un thread
- Portabilidad: realizado en JAVA

4.7.3 Especificaciones FIPA (Foundation for Intelligent Physical Agents)

Permite la interoperabilidad entre plataformas de diferentes empresas y organizaciones, FIPA especifica la normatividad que se debe llevar a cabo para la construcción de plataformas Multi-Agente. Estas especificaciones no restringen la tecnología usada para la implementación de plataformas. [23]

5. Materiales y Métodos

5.1 Recursos Materiales

Descripción	Cantidad	Valor/unitario	Valor Total
Computadoras	2	800.00	1600.00
Webcam VGA	1	16.00	16.00
Aire acondicionado	1	450.00	450.00
Placa Arduino	1	80.00	80.00
Sensor temperatura	1	4.00	4.00
Sensor Infrarrojo	1	5.00	5.00
Impresión del circuito	1	3.00	3.00
Cables	9	0.15	1.35
Protoboard	1	20.00	20.00
LED IR	1	2.00	2.00
Total			2181.35

Tabla 8: Materiales físicos

Descripción	Costo	Total
Transporte	0.25 centavos pasaje	40.00
Internet	22 dólares el mes	110.00
Total		150.00

Tabla 9: Servicios

El método que se utilizó para realizar el presente proyecto de titulación es el método deductivo, el cual se usa para determinar el problema y de esta forma plantear los objetivos partiendo de lo general a lo específico. Obteniendo un análisis detallado del problema se deriva a desarrollar un sistema con visión artificial que identifica personas en una sala de conferencias y a través de la placa Arduino consiga manipular la temperatura ambiente adecuada en la misma.

5.2 Técnicas de Observación

La técnica a utilizar es la entrevista, la cual permite obtener datos reales y confiables de los constantes cambios climáticos que se viven en la actualidad y cómo estos afectan a las personas y por ende al desarrollo y productividad de las diferentes empresas.

El uso de esta técnica nos permitió determinar la situación actual en que se encuentran las distintas empresas respecto al control de temperatura ambiente en sus respectivas salas de conferencia.

Los resultados obtenidos se detallan a continuación.

- La temperatura ambiente adecuada para el desempeño laboral en las empresas se encuentra aproximadamente entre los 17°C y 27°C.
- La incomodidad de manipular el aire acondicionado en sus distintas variantes de temperatura presenta molestias en los usuarios.

- La constante aglomeración de personas produce sudores, deshidratación y trastornos que pueden provocar fatiga y accidentes.
- El estrés térmico genera un estado de malestar físico provocado por una exposición excesiva al frío o al calor.
- La climatización es responsable en algunos casos de más de la mitad del consumo de energía, por su incorrecto uso de los aires acondicionados.

Para mejorar las interrogantes anteriores se pretende hacer uso de los avances tecnológicos, proponiendo una solución innovadora la cual nos permite administrar la temperatura ambiente a través de la técnica de visión artificial integrada con la plataforma Arduino, para manipular correctamente el aire acondicionado optimizando el correcto uso de los recursos energéticos.

Las fases en las que se ha dividido el proyecto son:

Análisis.- En esta fase se realizó la recolección de información así como también el análisis de la misma, además se utilizó las técnicas de observación para obtener datos confiables del manejo de la temperatura dentro de salas de conferencia.

Análisis de Requerimientos.- Se definió los requerimientos funcionales y no funcionales del sistema a partir de la información obtenida en la fase anterior.

Diseño del Sistema.- Se realizó el diseño de la arquitectura Arduino con el sensor de temperatura ambiente y el actuador remoto, además se utilizó la metodología Prometheus la cual sirvió de ayuda para diseñar los agentes inteligentes utilizados en la integración del sistema.

Integración.- En esta fase se logró integrar el módulo de visión artificial con la placa Arduino de tal manera que esté listo para realizar las pruebas correspondientes en la siguiente fase.

Pruebas.- Al finalizar completamente las fases anteriores no queda más que realizar las pruebas correspondientes las cuales se realizará en esta fase documentando los resultados obtenidos y la manera como se llevó a cabo dicho proceso.

Cada una de ellas se detalla en el apartado de resultados.

5.3 Metodología Prometheus

Se utilizó esta metodología en base a que ésta soporta todo el ciclo de vida del proyecto y mantiene el uso de procesos interactivos en los cuales se perfilan todas las funcionalidades, una de sus características principales son el uso de metas planes y eventos además de las BDI (creencias deseos e intenciones) que se maneja dentro de los agentes.

5.3.1 Fases de la Metodología Prometheus

Especificación del sistema.

Identificación de las funcionalidades básicas del sistema. En esta fase hay que tener en cuenta lo siguiente:

- ¿Cómo interactúa el sistema con el entorno?
- Entrada de información desde el entorno (percepciones).
- Mecanismos para actuar sobre el entorno (acciones).
- Eventos. Se generan por las percepciones del entorno, provoca reacciones en los agentes.

Diseño arquitectónico.

Determinar qué agentes contendrá el sistema y cómo interactúan. Características a tener en cuenta.

- ¿Qué agentes deben existir?
- Asignar funcionalidades a los agentes.
- Tener en consideración coherencia y acoplamiento.
- Agrupación de funcionalidades. Si las funcionalidades usan los mismos datos.

Diseño detallado.

Desarrollar la estructura interna de cada agente. Se debe especificar el uso de:

- Planes definidos
- Disparadores para metas y eventos
- Implementación de sistemas BDI
- Habilidades (módulos o comportamientos)
- Estructuras de datos

Habilidades o comportamientos

- Las habilidades implementarán las funcionalidades que se han descrito anteriormente.
- Para cada agente habrá un conjunto de habilidades que lo definan.
- Una habilidad puede estar formada a su vez por otras habilidades.
- En último nivel las habilidades están formadas por planes, eventos y datos.
- Una misma habilidad puede ser usada por otros comportamientos. [24]

6. Resultados

El sistema para la manipulación de temperatura ambiente, nace por la necesidad que tienen las empresas en sus salas de conferencia para el manejo automático de la temperatura ambiente, brindando de esta manera un mejor confort a sus asistentes, además de optimizar los recursos del aire acondicionado instalado en dicha sala.

El sistema fue diseñado pensando en el usuario final quienes son los asistentes a las salas a los cuales se les pretende asegurar una temperatura con una variación mínima de tres grados, evitando los cambios bruscos de temperatura.

6.1 Perspectiva del sistema

La idea fundamental del sistema permite al encargado de las salas de conferencia, automatizar el proceso de activación y manipulación del aire acondicionado, así, obteniendo una temperatura estable, comparada con la forma manual de hacer esto.

Esta información fue obtenida por la entrevista realizada a los responsables de las salas de conferencias en los hoteles de la ciudad de Loja. Dicho esto se pudo realizar el análisis del sistema del cual se obtuvieron los siguientes requerimientos.

6.1.1 Requerimientos Funcionales

Código	Descripción	Categoría
RF001	Al administrador modificar los parámetros de configuración del sistema.	Evidente
RF002	Al administrador visualizar la temperatura ambiente obtenida a través del sensor de temperatura.	Evidente
RF003	Al administrador visualizar el número de personas	Evidente

	identificadas por la cámara.	
RF004	Fijar la temperatura ambiente al aire acondicionado a través de la placa Arduino.	Evidente
RF005	Identificar la temperatura ambiente actual mediante el sensor de temperatura.	Evidente
RF006	Contar el número de personas encontradas dentro de una sala.	Evidente
RF007	Actualizar cada 1 segundos los datos obtenidos por el sensor de temperatura y la cámara.	Evidente
RF008	Al administrador ingresar la capacidad máxima de la sala (número de personas).	Evidente
RF009	AL administrador visualizar un mensaje cuando se haya llegado al límite máximo de capacidad de la sala.	Evidente

Tabla 10: Requerimientos funcionales

6.1.2 Requerimientos no Funcionales

Código	Descripción
RNF001	El sistema será desarrollado en el lenguaje de programación JAVA.

RNF002	El sistema será desarrollado bajo un entorno de escritorio
RNF003	El sistema de visión artificial estará integrado con la placa Arduino.
RNF004	El sistema tendrá un tiempo de respuesta de aproximadamente 3 segundos
RNF 005	El sistema contará con la cámara como periférico de entrada de datos.
RNF 006	La placa Arduino estará compuesta por un sensor y un actuador para la captura de datos.
RNF 007	El sistema será multiplataforma.
RNF 008	El sistema contará con una licencia GPL

Tabla 11: Requerimientos no funcionales

6.2 Diagrama de bloques del sistema

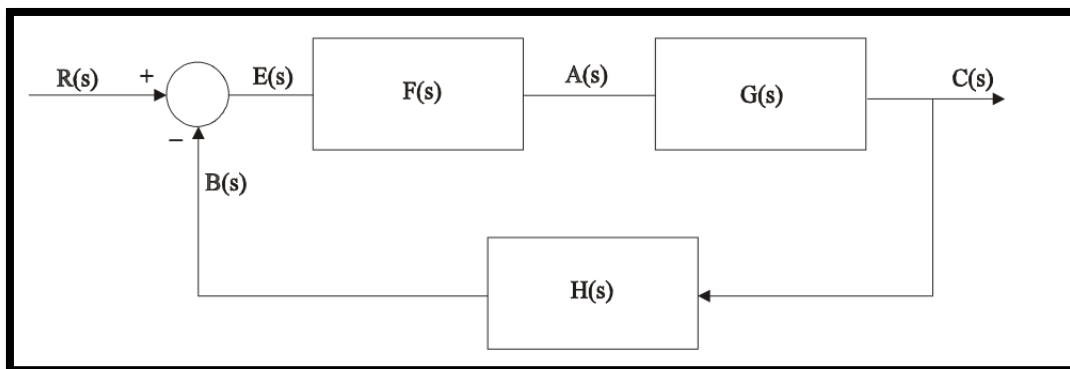


Diagrama 1: Diagrama de bloques del sistema

En donde cada una de las funciones se describe a continuación

1. R(s): Entrada de referencia (temperatura confort)
2. C(s): Señal de salida (temperatura LM35)
3. B(s): Señal de realimentación
4. E(s): Señal de error (variación de temperatura confort y LM35)
5. A(s): Señal activadora (Led de infrarrojos)
6. H(s): Cadena de retorno o realimentación
7. G(s): Planta o proceso a controlar (aire acondicionado)
8. F(s): Regulador o controlador (sistema de agentes)

Iniciando el análisis desde la variable controlada hacia atrás, tendríamos:

$$C(s) = A(s) \cdot G(s) ; A(s) = E(s) \cdot F(s) ; E(s) = R(s) - B(s) ; \\ B(s) = C(s) \cdot H(s)$$

Sustituyendo progresivamente, y en sentido inverso al anterior, tenemos:

$$E(s) = R(s) - C(s) \cdot H(s) ; A(s) = [R(s) - C(s) \cdot H(s)] \cdot F(s) \\ C(s) = [R(s) - C(s) \cdot H(s)] \cdot F(s) \cdot G(s) = R(s) \cdot F(s) \cdot G(s) - C(s) \cdot H(s) \cdot F(s) \cdot G(s)$$

Despejando para obtener la relación C(s)/R(s):

$$\frac{C(s)}{R(s)} = M(s) = \frac{F(s) \cdot G(s)}{1 + F(s) \cdot G(s) \cdot H(s)}$$

Ecuación 1: Función de transferencia (M(s))

6.3 Diseño del circuito con la placa Arduino

6.3.1 Recibir variación de temperatura desde Arduino con un LM35

La captura de la temperatura con un LM35 se la consigue con una variación de 5v por lo que la primera pata del mismo es conectada a la salida de 5v del Arduino, la recolección de estos datos se lo hace analógicamente por ello se conecta la pata del centro en cualesquier pin analógico del Arduino es nuestro caso el pin analógico 0, y la última pata del LM35 es conectado al GND del Arduino. De esta forma el diseño del circuito en la conexión del LM35 es igual al que se muestra en la figura 10.

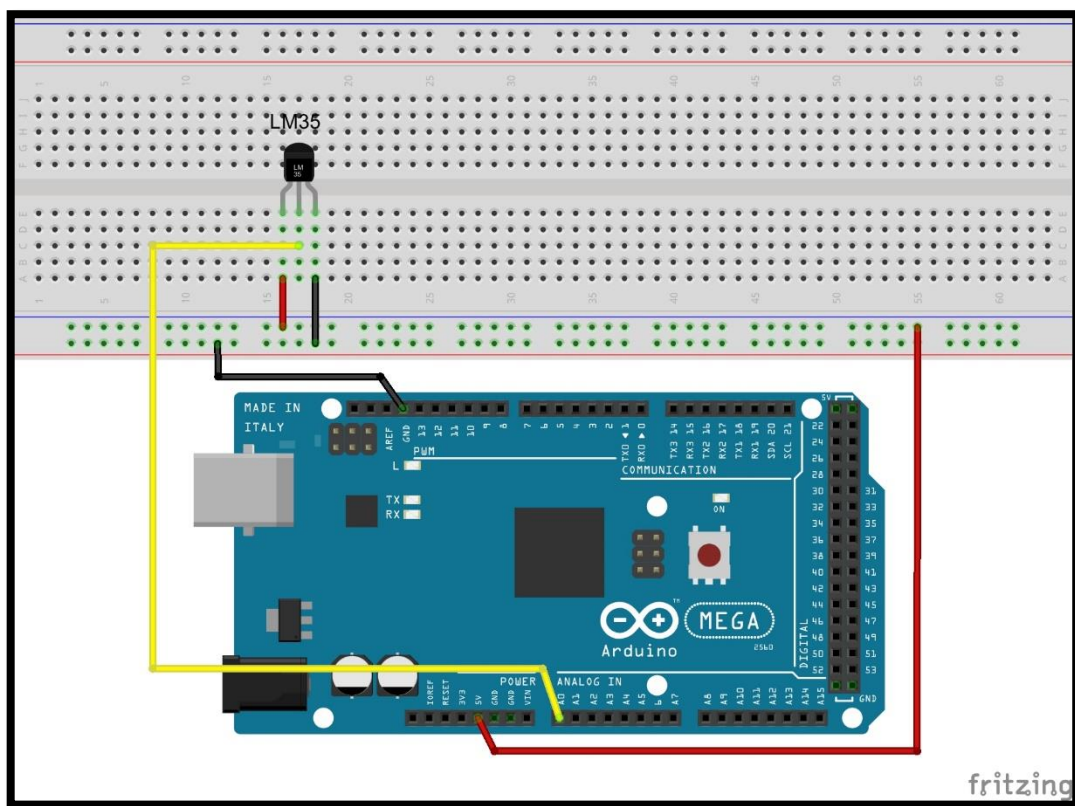


Figura 10: Diseño circuito (LM35)

6.3.2 Emitir señal infrarroja desde Arduino con un led de infrarrojo

El cátodo del led es el que conectamos al GND del Arduino. Y el ánodo es conectado al pin digital 9 del Arduino, pero antes de hacer tal conexión por precaución conectamos una resistencia de 1k ohm. El uso de la librería IRemote.h facilita la implementación del envío de estas señales y por ende debemos regirnos a su

documentación la cual nos dice que el pin digital de salida en el Arduino mega 2506 es el pin 9 obligatoriamente. De esta forma el diseño del circuito en la conexión del led de infrarrojo es igual al que se muestra en la figura 11.

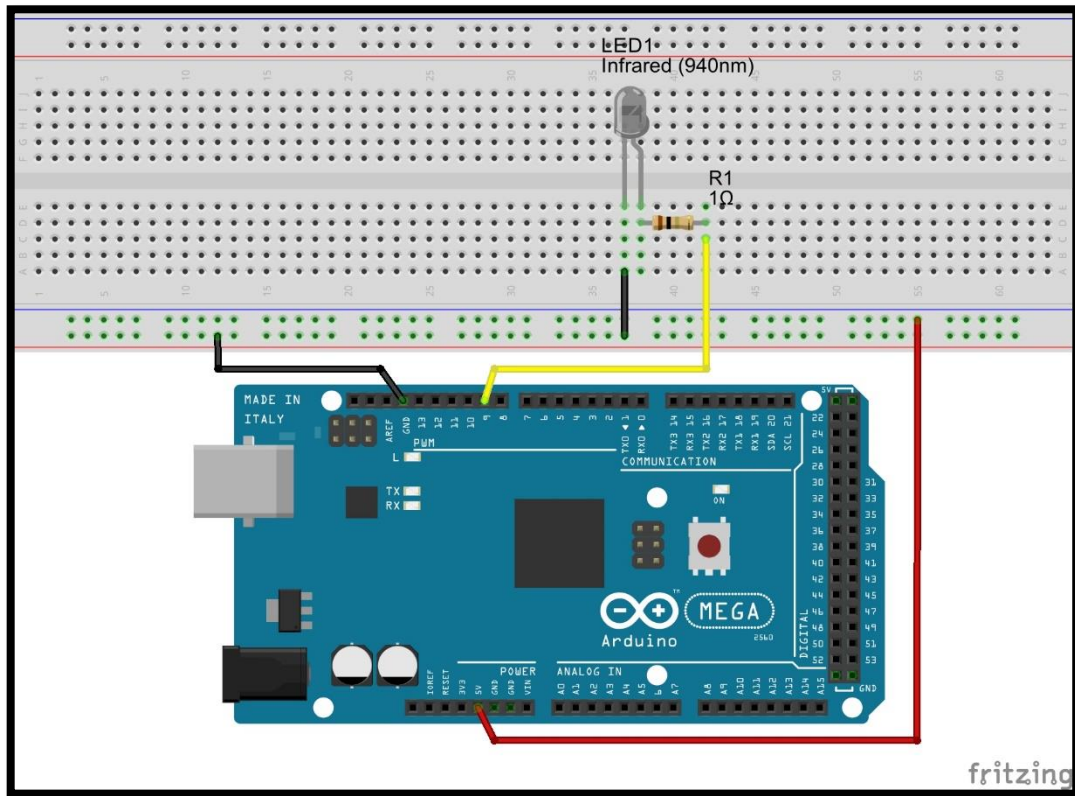


Figura 11: Diseño circuito (LED Infrarrojo)

6.3.3 Decodificar una señal de infrarrojo con Arduino

La primera pata del IR Receiver es la que conectaremos al pin digital 11 del Arduino, esto por comodidad ya que se lo puede hacer en cualquier pin en tanto este sea digital, descartando el pin 9 del Arduino, ya que éste está siendo usado para el envío de señales infrarrojas. La pata del medio al GND del Arduino y la última a la salida de 5V que el Arduino nos brinda. De esta forma el diseño del circuito en la conexión del IR Receiver es igual al que se muestra en la figura 12.

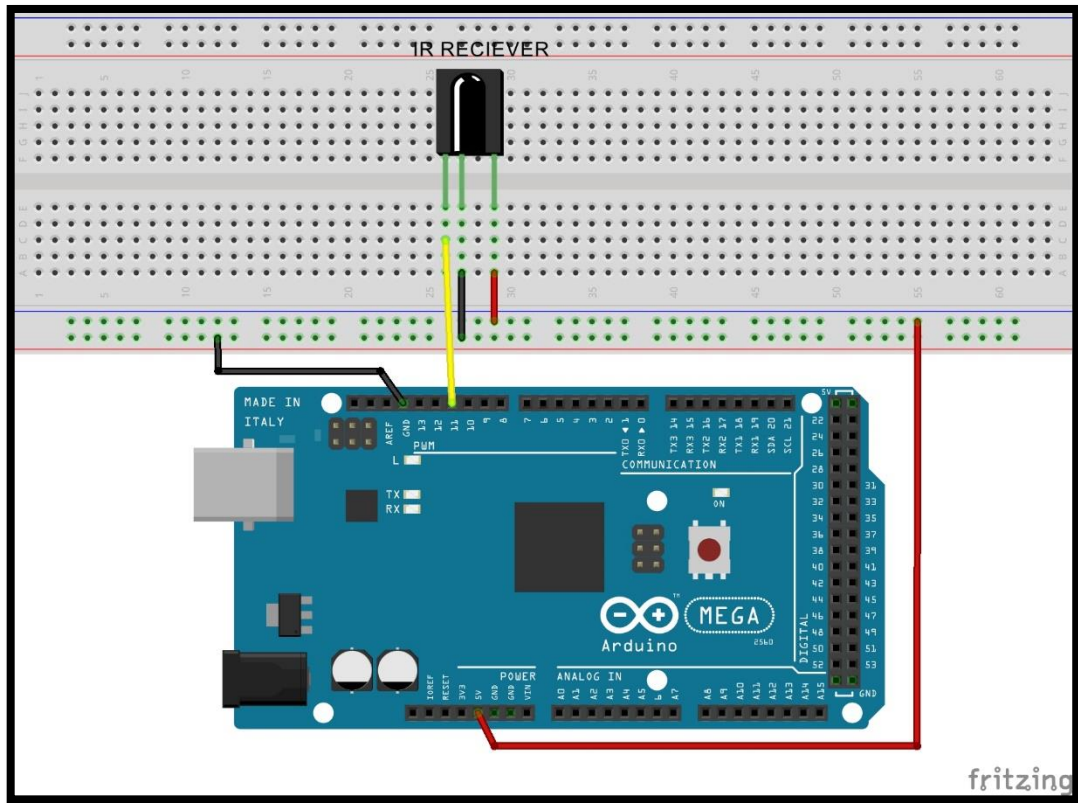


Figura 12: Diseño del circuito (IR Receiver)

El IR Receiver conjuntamente con el LM35 y el LED de Infrarrojo se integran en un solo diagrama como se muestran en las figuras 13, 14 y 15 las mismas que muestran el diseño del circuito en el protoboard, el esquema del circuito y por último el PCB del circuito respectivamente.

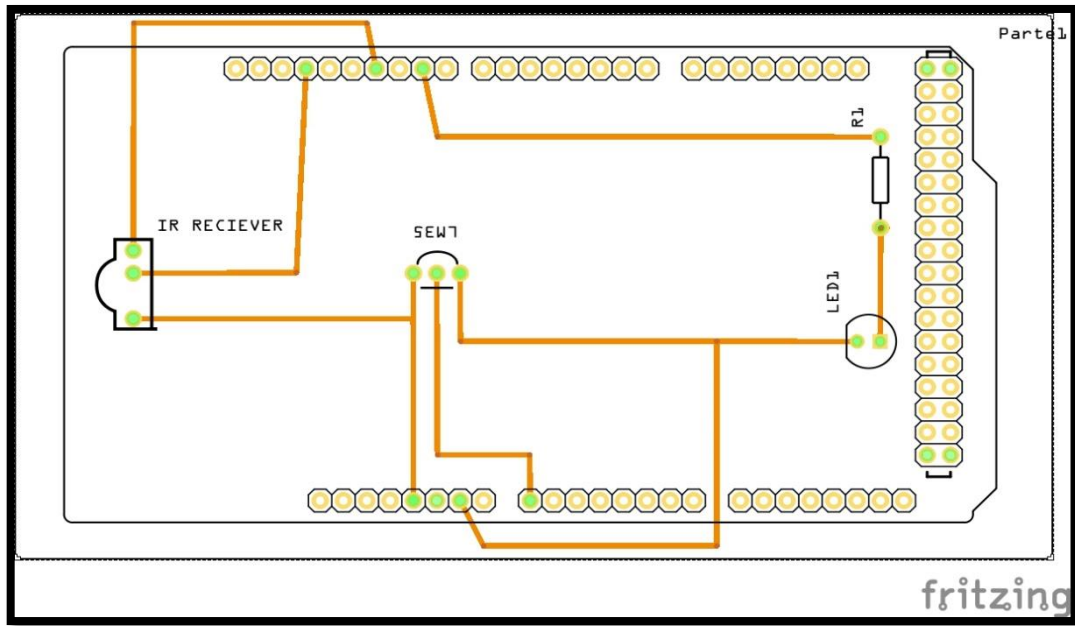


Figura 15: PCB del circuito (LM35, LED Infrarrojo y IR Receiver)

6.4 Diseño del sistema

6.4.1 Especificación del sistema.

Casos de uso.

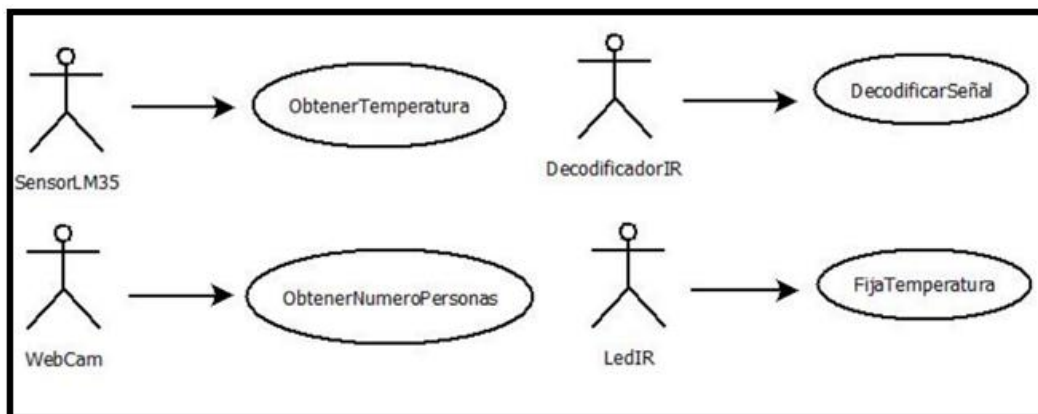


Diagrama 2: Casos de uso del sistema

Descripción de Casos de Uso.

Caso de Uso:	Obtener Temperatura	Actor:	SensorLM35
Resumen:	El sensor LM35 obtendrá la temperatura ambiente, esta información la enviará el agente reactivo hacia el agente deliberativo a través de un mensaje, este agente consulta sus BDI y luego ejecuta la acción correspondiente desde la placa Arduino hacia el aire acondicionado.		
Objetivos:	<ul style="list-style-type: none"> ● Obtener temperatura ambiente. ● Enviar la información hacia el agente deliberativo. ● Actualizar datos. 		
Precondiciones:	<ul style="list-style-type: none"> ● Sensor LM35 conectado. ● Placa Arduino conectada. 		
Poscondiciones:	<ul style="list-style-type: none"> ● Disponibilidad de datos. 		

Tabla 12: Descripción caso de uso obtener temperatura

Caso de Uso:	Obtener número de personas	Actor:	WebCam
Resumen:	La webcam captura la imagen de la sala de conferencia y con OpenCV se identifica el número de personas a través de la técnica de visión artificial, esta información la envía el agente reactivo hacia el agente deliberativo, este consulta sus BDI y luego ejecuta la acción correspondiente desde la placa Arduino		

	hacia el aire acondicionado.
Objetivos:	<ul style="list-style-type: none"> • Obtener número de personas. • Enviar la información hacia el agente deliberativo. • Actualizar datos.
Precondiciones:	<ul style="list-style-type: none"> • Cámara disponible. • Existencia de personas.
Poscondiciones:	<ul style="list-style-type: none"> • Disponibilidad de datos.

Tabla 13: Descripción caso de uso obtener número de Personas

Caso de Uso:	Decodificar Señal	Actor:	Decodificador IR
Resumen:	Los códigos emitidos por el mando a distancia serán interpretados por el decodificador IR, obteniendo de esta manera todos los códigos correspondientes a cada tecla pulsada desde el mando a distancia, con la finalidad de tener disponibles cada uno de ellos para posteriormente emplearlos de acuerdo a lo dispuesto en el agente deliberativo.		
Objetivos:	<ul style="list-style-type: none"> • Determinar los códigos hexadecimales del mando a distancia. • Identificar datos. 		

Precondiciones:	<ul style="list-style-type: none"> ● Sensor Infrarrojo conectado. ● Placa Arduino conectada.
Poscondiciones:	<ul style="list-style-type: none"> ● Códigos hexadecimales detectados.

Tabla 14: Descripción caso de uso Decodificar Señal

Caso de Uso:	Fijar Temperatura	Actor:	Led IR
Resumen:	Los datos provenientes del agente deliberativo se procesaran en la placa Arduino para enviar la señal hacia el aire acondicionado, de acuerdo a las reglas previamente establecidas en las BDI del agente deliberativo dando como resultado el control automático del aire acondicionado.		
Objetivos:	<ul style="list-style-type: none"> ● Conectarse con el aire acondicionado a través del infrarrojo. ● Controlar aire acondicionado mediante el actuador Led IR. ● Fijar datos. 		
Precondiciones:	<ul style="list-style-type: none"> ● Placa conectada al circuito. ● Datos existentes en el agente deliberativo. 		
Poscondiciones:	<ul style="list-style-type: none"> ● Temperatura adecuada. 		

Tabla 15: Descripción caso de uso fijar temperatura

El sistema contará con dos agentes, respectivamente como se muestra en el diagrama 3.

Agente Reactivo.

Este agente obtendrá a través de sus percepciones los datos arrojados por los actores en este caso el sensor de temperatura y la webcam.

Agente Deliberativo.

Este agente deliberativo consulta la información obtenida del agente reactivo para posteriormente de acuerdo a las reglas definidas en las BDI ejecutar una acción correspondiente en este caso enviar las señales a la placa Arduino para controlar el aire acondicionado con esta.

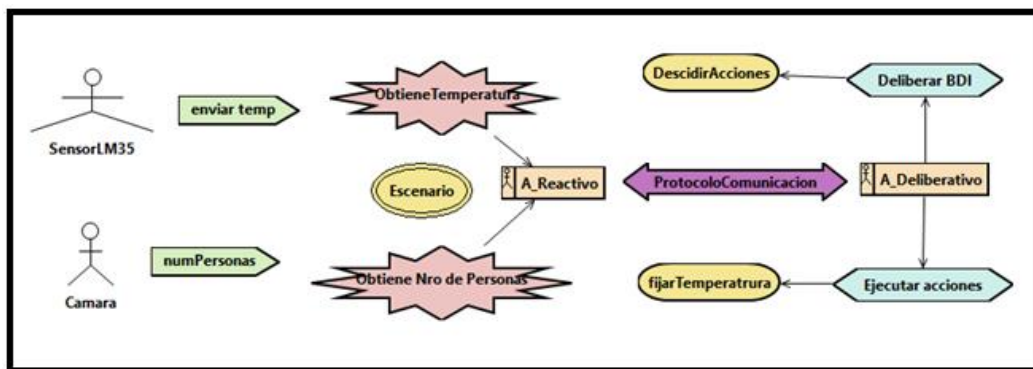


Diagrama 3: Especificación del sistema

En la diagrama 3 se muestra las especificaciones del sistema que nos permiten identificar las funcionalidades básicas del mismo, los actores emiten datos de entrada que serán procesados a través de los agentes inteligentes y enviados a la placa Arduino para controlar la temperatura ambiente.

6.4.2 Diseño Arquitectónico.

Como se muestra en el diagrama 4 los actores cámara y sensor LM35 son quienes proveen de los datos de entrada al sistema, ya que el agente reactivo a través de sus percepciones obtiene estos datos los mismos que se enviaran al agente deliberativo para que cumpla con la acción de activar el aire acondicionado mediante la placa Arduino.

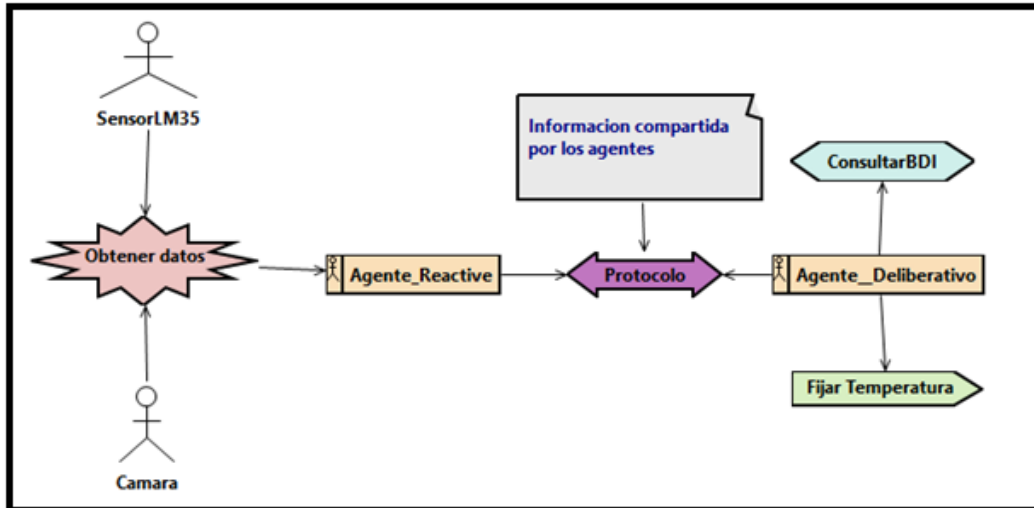


Diagrama 4: Diseño Arquitectónico

La activación del aire acondicionado se efectuará de acuerdo a las reglas que se ajustan en las creencias del agente deliberativo, de este depende fijar una temperatura ambiente estable, ya que, éste agente es el encargado de deliberar de acuerdo a los datos obtenidos por el agente reactivo a través de los actores cámara y sensor de temperatura como se muestra en el diagrama 4.

6.4.3 Desarrollo Detallado.

El agente deliberativo tendrá un comportamiento basado en las BDI, obtendrá los datos provistos por el agente reactivo a través de los actores, luego consulta su tabla de creencias donde se define las reglas de la temperatura ambiente previamente establecida, basada en normativas y resultados de encuestas realizadas.

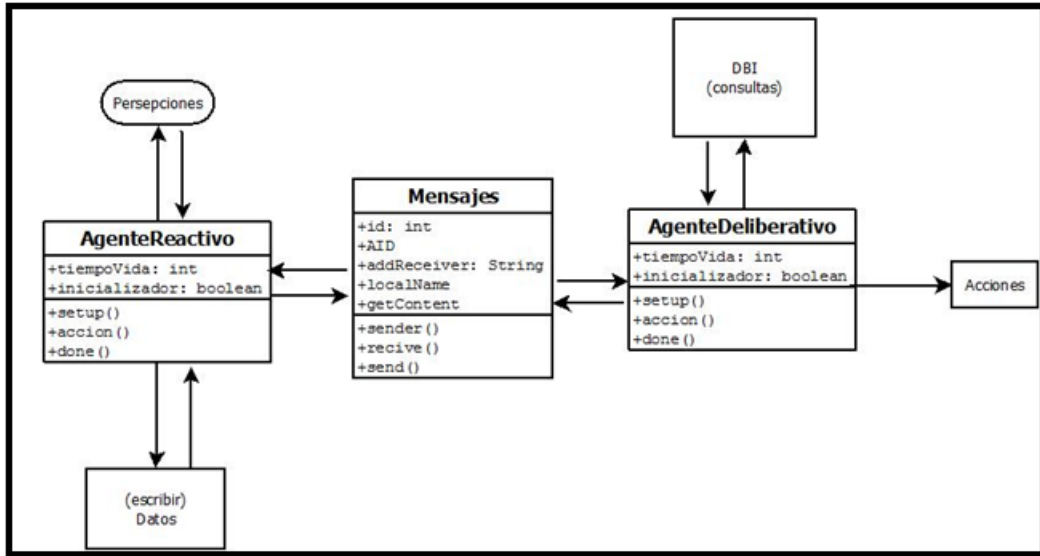


Diagrama 5: Diseño detallado

Por otro lado el agente deliberativo ejecutará las acciones recibidas del agente reactivo y a su vez enviará como resultado la temperatura ambiente conveniente a la placa Arduino para que esta pueda activar el aire acondicionado de acuerdo a la nueva temperatura.

6.5 Diagramas de componentes

6.5.1 Diagrama de componentes integración del sistema

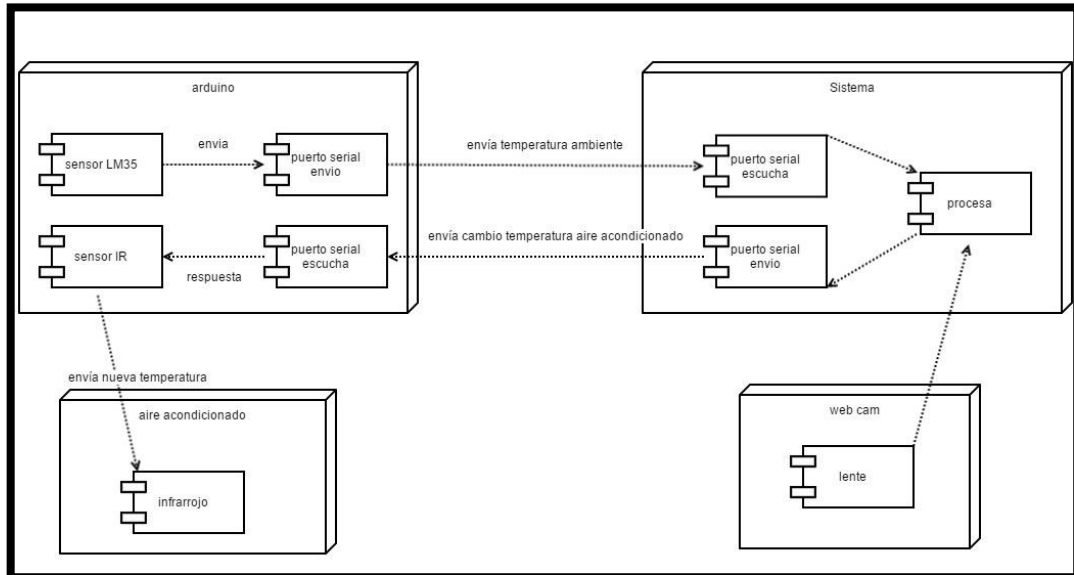


Diagrama 6: Diagrama de componentes integración del sistema

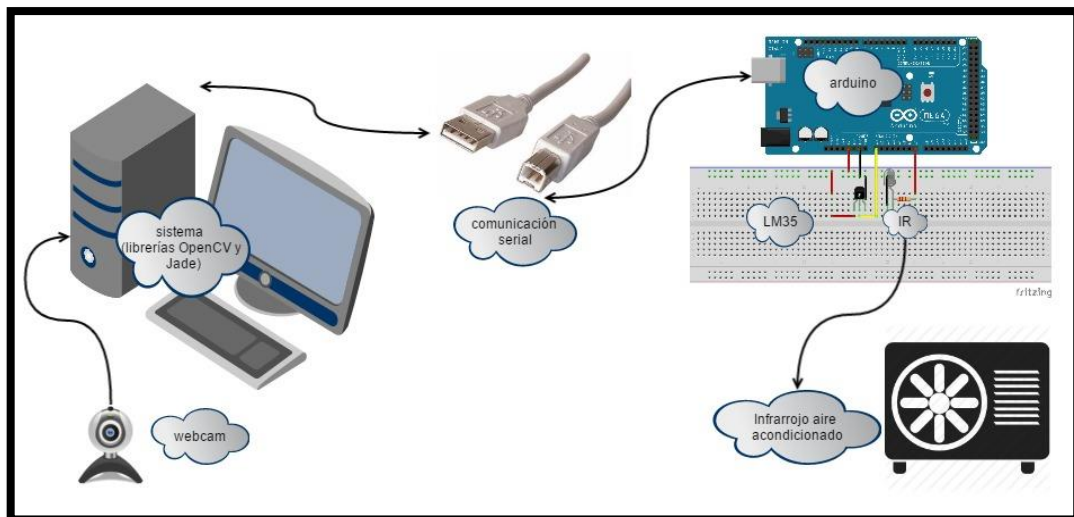


Figura 16: Descripción diagrama de componentes integración del sistema

6.6 Diagramas de flujo

6.6.1 Diagrama de flujo captura de la imagen.

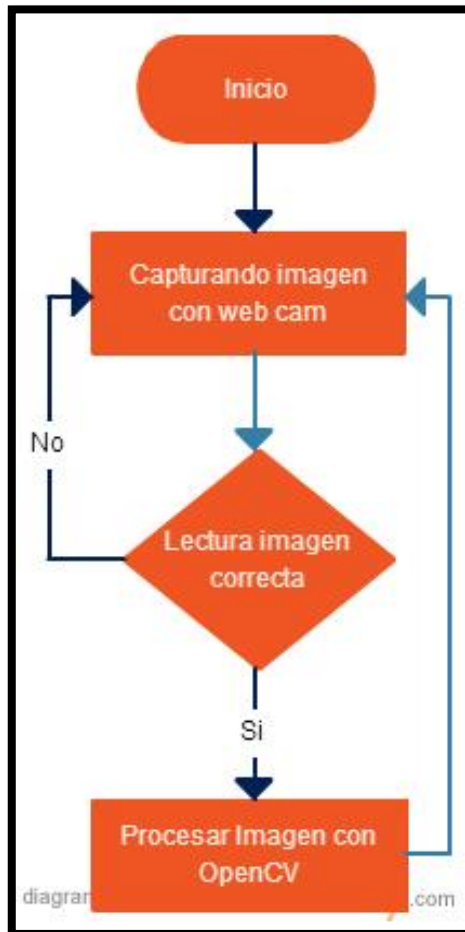


Diagrama 7: Diagrama de flujo captura de la imagen

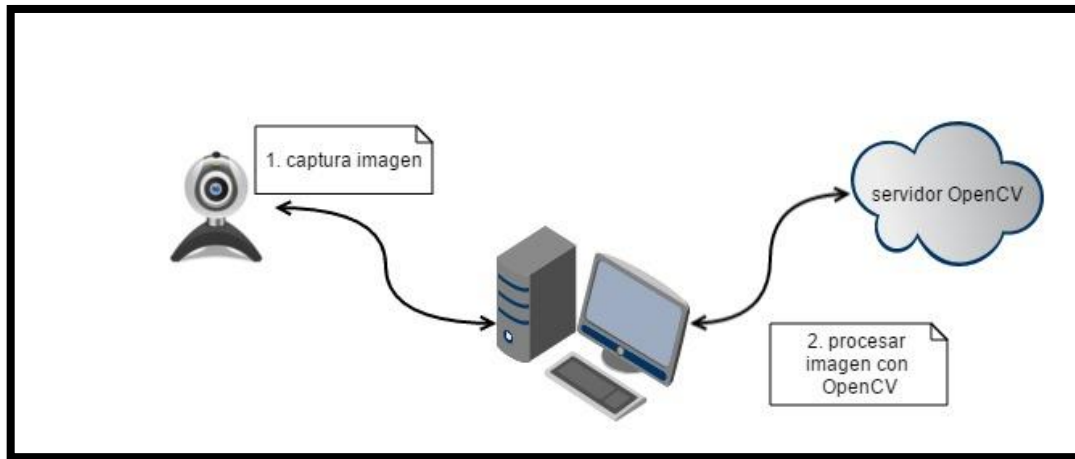


Figura 17: Descripción diagrama de flujo captura de la imagen.

La webcam es la encargada de la captura de imágenes dentro de la sala de conferencias, estas imágenes son obtenidas por el sistema y procesadas con OpenCV.

6.6.2 Diagrama de flujo procesamiento de la imagen.



Diagrama 8: Diagrama de flujo procesamiento de la imagen.

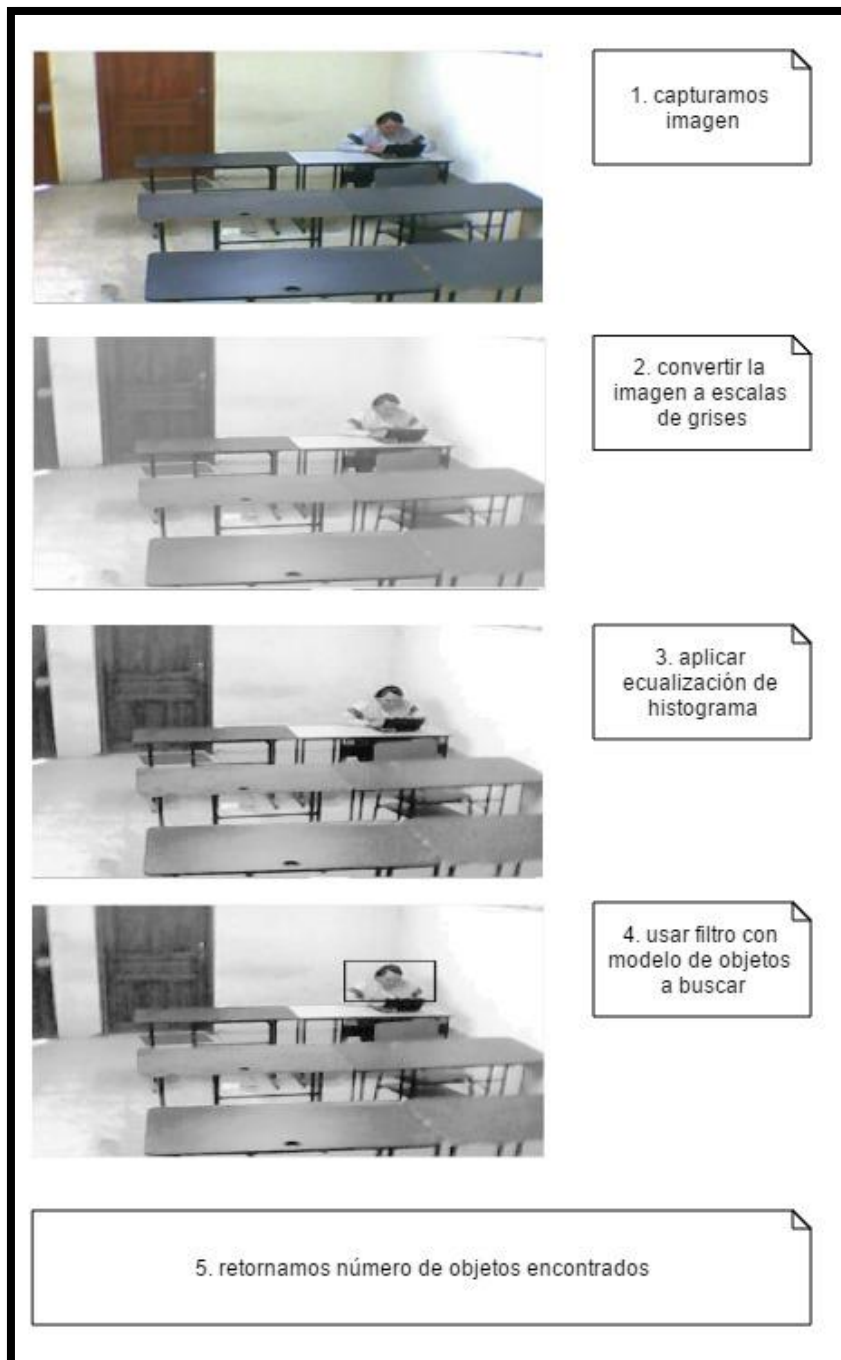


Figura 18: Descripción diagrama de flujo procesamiento de la imagen.

Convertimos la imagen a escala de grises, para que OpenCV procese menos información pues las imágenes a color poseen mayor número de píxeles. Lo siguiente que debemos hacer es aplicar ecualización de histograma a la imagen, esto con el fin de estandarizar el contraste y brillo de la misma, para que las distintas condiciones de iluminación no afecten la detección de objetos en la imagen.

Con la imagen procesada ahora debemos cargar el clasificador que deseamos utilizar, los archivos .xml que debemos cargar se encuentran en C:\Opencv\data aquí se encuentran varias carpetas que contienen los distintos clasificadores. En este caso usaremos upperbody.xml el mismo que clasifica personas considerando la parte superior del cuerpo y por último retornamos el número de objetos encontrados por el clasificador.

6.6.3 Diagrama de flujo comunicación Sistema con Arduino

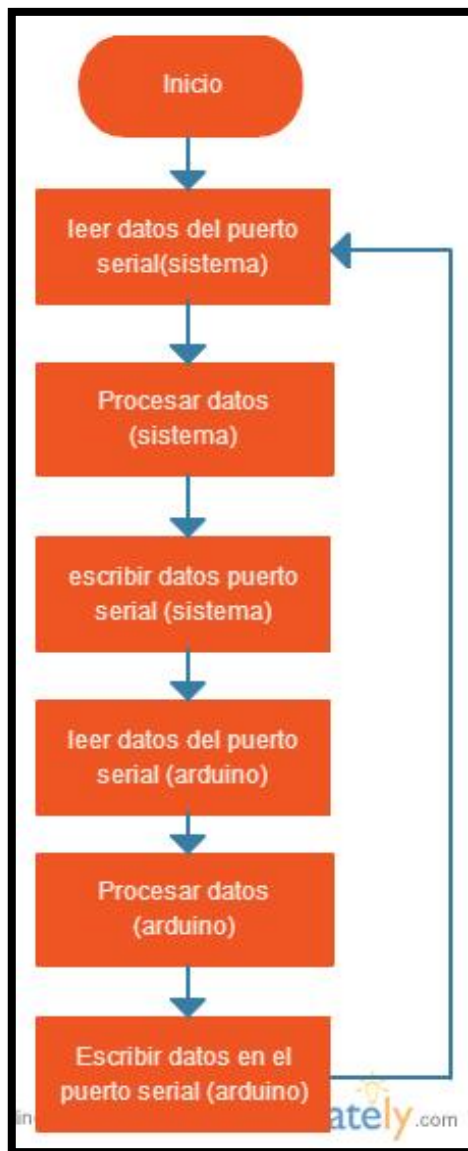


Diagrama 9: Diagrama de flujo comunicación Sistema con Arduino

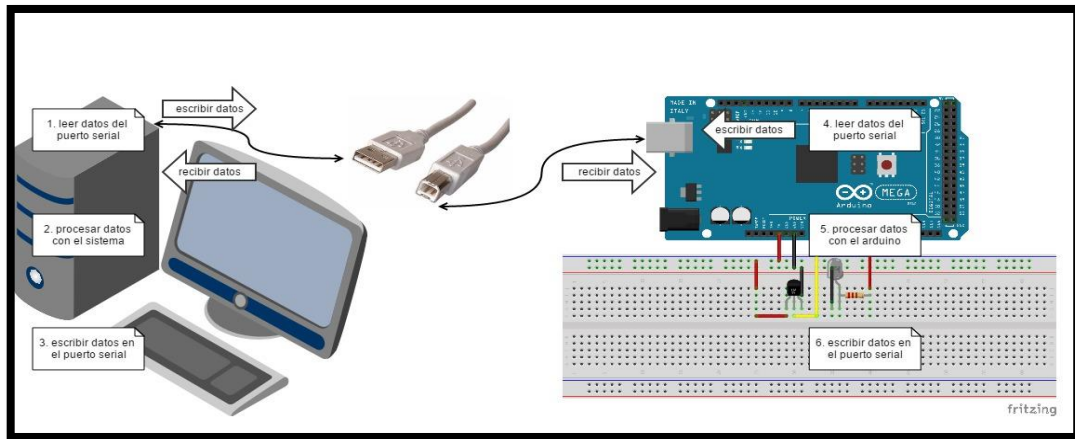


Figura 19: Descripción diagrama de flujo comunicación Sistema con Arduino

Arduino posee la habilidad de comunicación serial. El puerto serie ha quedado un poco en desuso a favor de la tecnología USB, Arduino cuenta con un convertidor de Serial a USB que permite a nuestra placa ser reconocida por nuestra computadora como un dispositivo conectado a un puerto COM aun cuando la conexión física sea mediante USB, lo que permite una comunicación limpia con nuestro ordenador. Para poder conectarnos con la placa Arduino es necesario instalar la biblioteca RXTX, esta biblioteca posee los jar necesarios para poder establecer comunicación vía serial con la placa.

6.6.4 Diagrama de flujo programación del Arduino

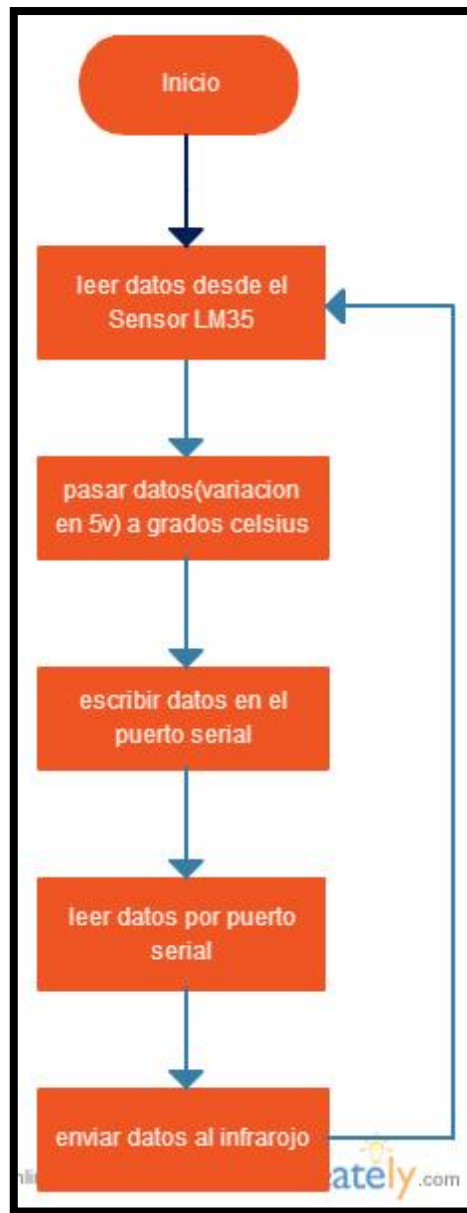


Diagrama 10: Diagrama de flujo programación del Arduino

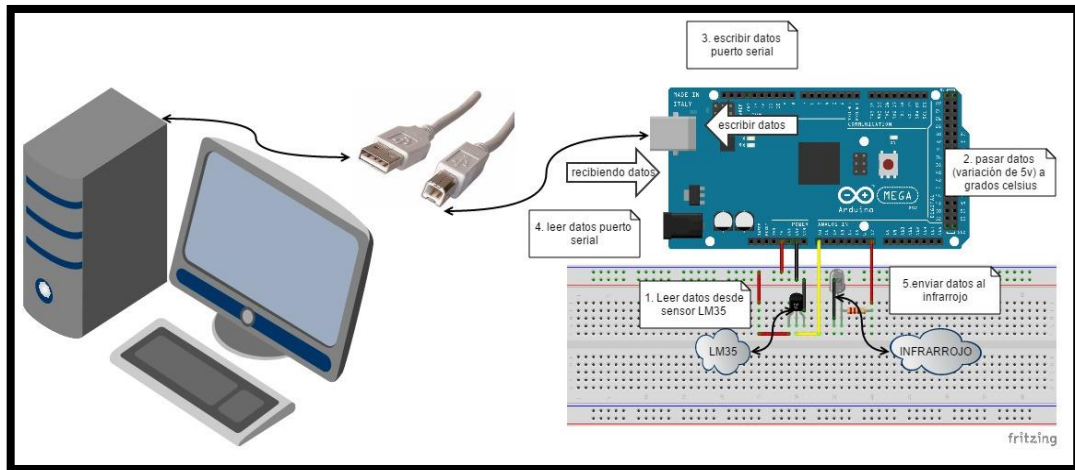


Figura 20: Descripción diagrama de flujo programación del Arduino

6.7 Interfaces de usuario

Las interfaces en el sistema permite la configuración sencilla y puesta en ejecución del mismo sin tener que poseer conocimientos de un experto informático, ya que el sistema como tal es muy intuitivo.

En la figura 21 se aprecia la interfaz principal la misma que se muestra a iniciar el sistema.

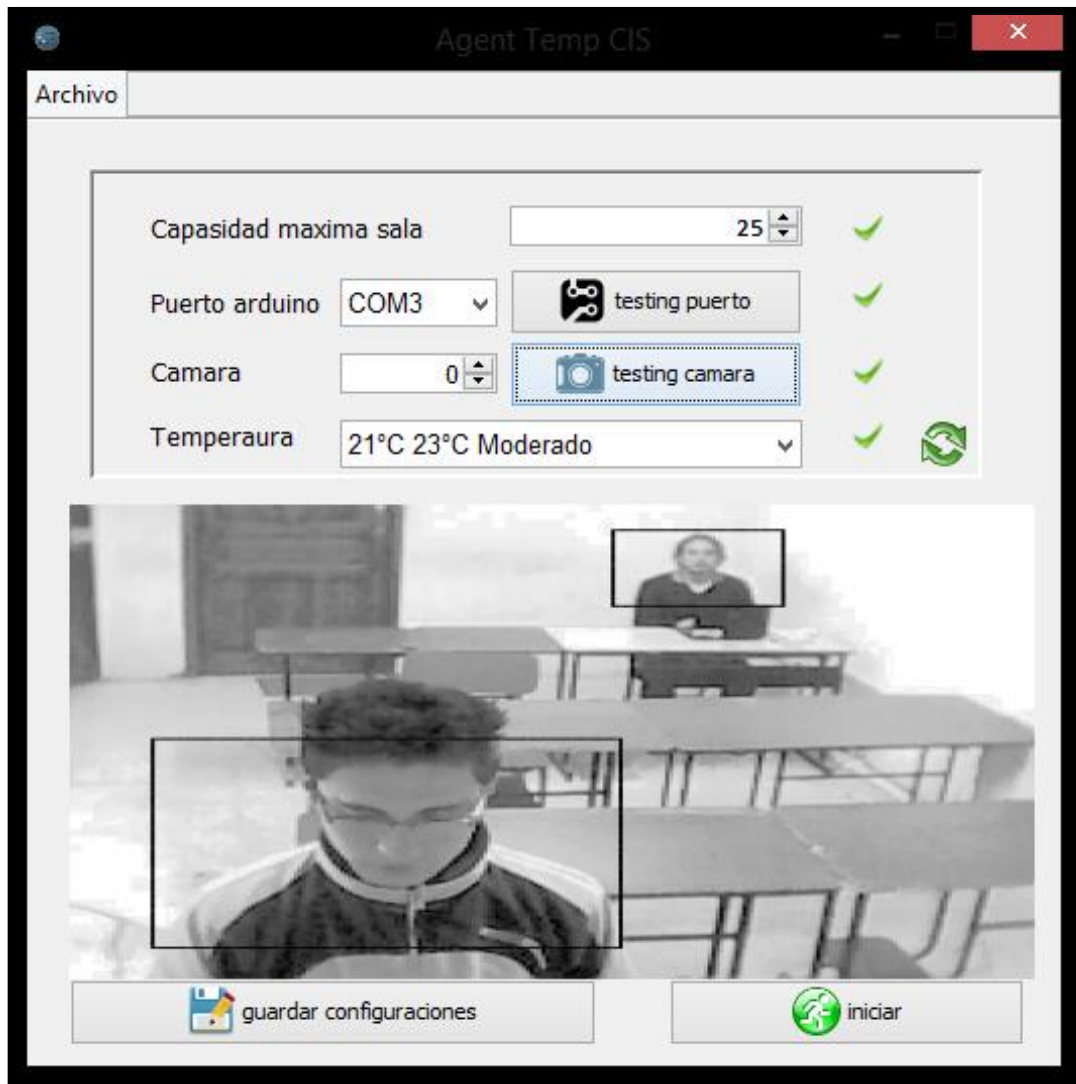


Figura 21: Interfaz principal, configuraciones del sistema.

En esta pantalla se puede configurar el sistema y dar paso a la ejecución del mismo, el sistema carga valores por defecto, los cuales suelen ser suficientes para la ejecución del sistema, Al iniciar la ejecución revisa si la placa Arduino está conectado al computador, además de revisar si existe una cámara disponible, una vez hecho esto se pasa a iniciar la ejecución del sistema; y, mostrará la pantalla de ejecución, la misma que se muestra en la figura 22.

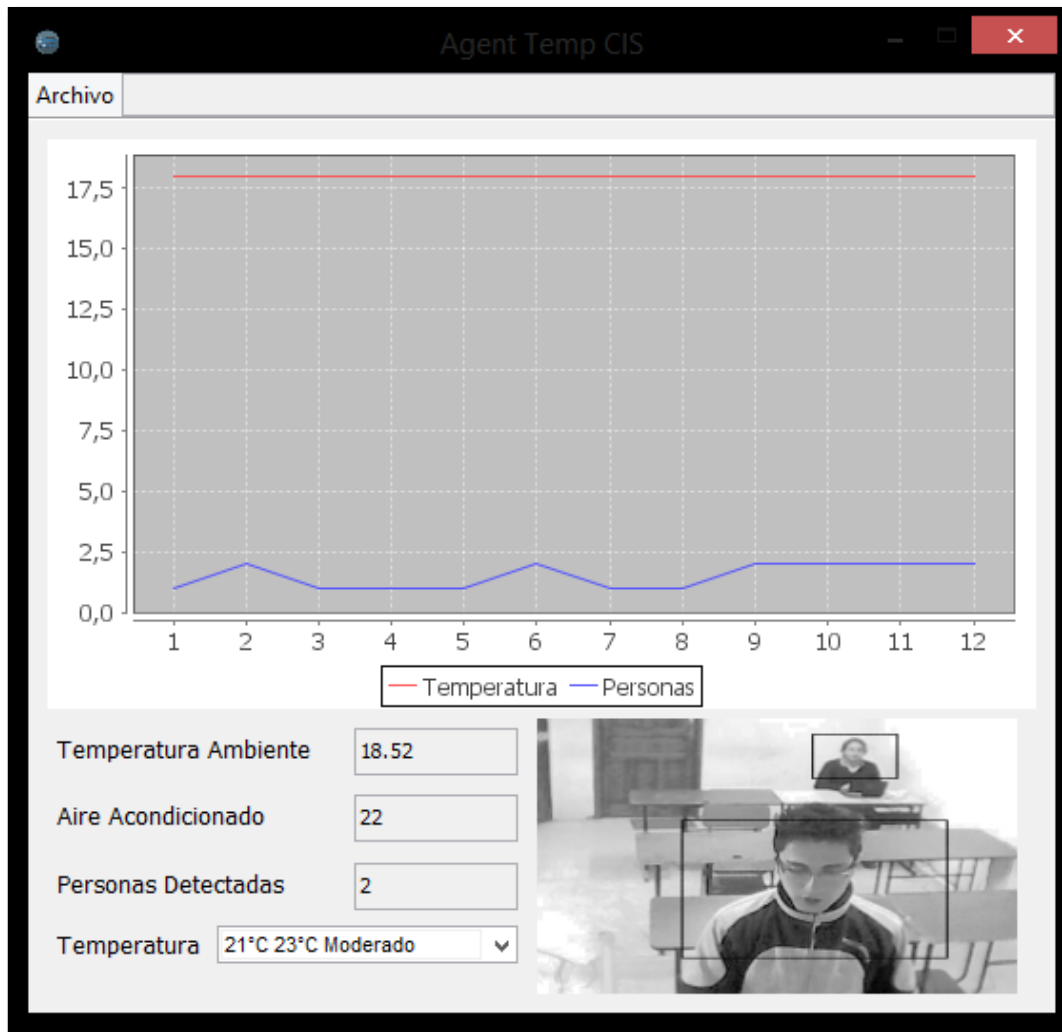


Figura 22: Interfaz ejecución del sistema.

Es aquí en donde el sistema empezará a trabajar por sí solo, mostrándonos la temperatura a la cual se encuentra la sala de conferencias (Temperatura Ambiente), la temperatura a la cual está trabajando el aire acondicionado (Aire Acondicionado), y las personas que ha detectado nuestra cámara (Personas Detectadas), además de permitir cambiar la temperatura confort (Temperatura).

En la parte derecha inferior se muestra la captura de la cámara en la sala de conferencias.

En la parte superior se muestra una tabla de resultados, la cual muestra las variantes de temperatura en color rojo y la variante de personas que ha registrado el sistema en color azul.

6.8 Plan de pruebas

A continuación se describe el plan de pruebas que fue ejecutado:

6.8.1 Alcance

Se plantea validar el correcto funcionamiento de los aspectos fundamentales del software realizado, como lo son: la integración del sistema con la plataforma Aduino, el reconocimiento de objetos con la librería OpenCV, la integración de la plataforma JADE para la gestión de agentes, validar que los comportamientos de los agentes creados funcionarán de forma correcta y por último las funcionalidades de la interfaz y control del aire acondicionado. Lo anterior basado en pruebas de caja negra, de caja blanca, funcionales y de sistema.

6.8.2 Requerimientos

Para ejecutar el plan de pruebas se requiere contar con las plataformas ARDUINO, JADE correctamente instaladas y en ejecución, del mismo modo se requiere la librería OpenCV correctamente agregada al proyecto, tener instalado el lenguaje de programación JAVA además del IDE eclipse en ejecución.

6.8.3 Procedimiento

Para validar el éxito de la prueba, se ejecutó los procedimientos por consola, además se escribió algunos mensajes entre algunas líneas de código de los métodos a probar, esto con el fin de confirmar si se llegó a un punto específico de forma satisfactoria y/o si la información, hasta ese punto, es correcta.

En la (Tabla 16 Resultados de prueba) se muestra la información adicional de las pruebas realizadas, así como de los resultados obtenidos.

6.8.4 Pruebas de Caja Negra

Interfaz de Configuración:

Pantalla principal: Permite seleccionar el puerto para la webcam y del mismo modo agregar el puerto serial para la comunicación con Arduino.

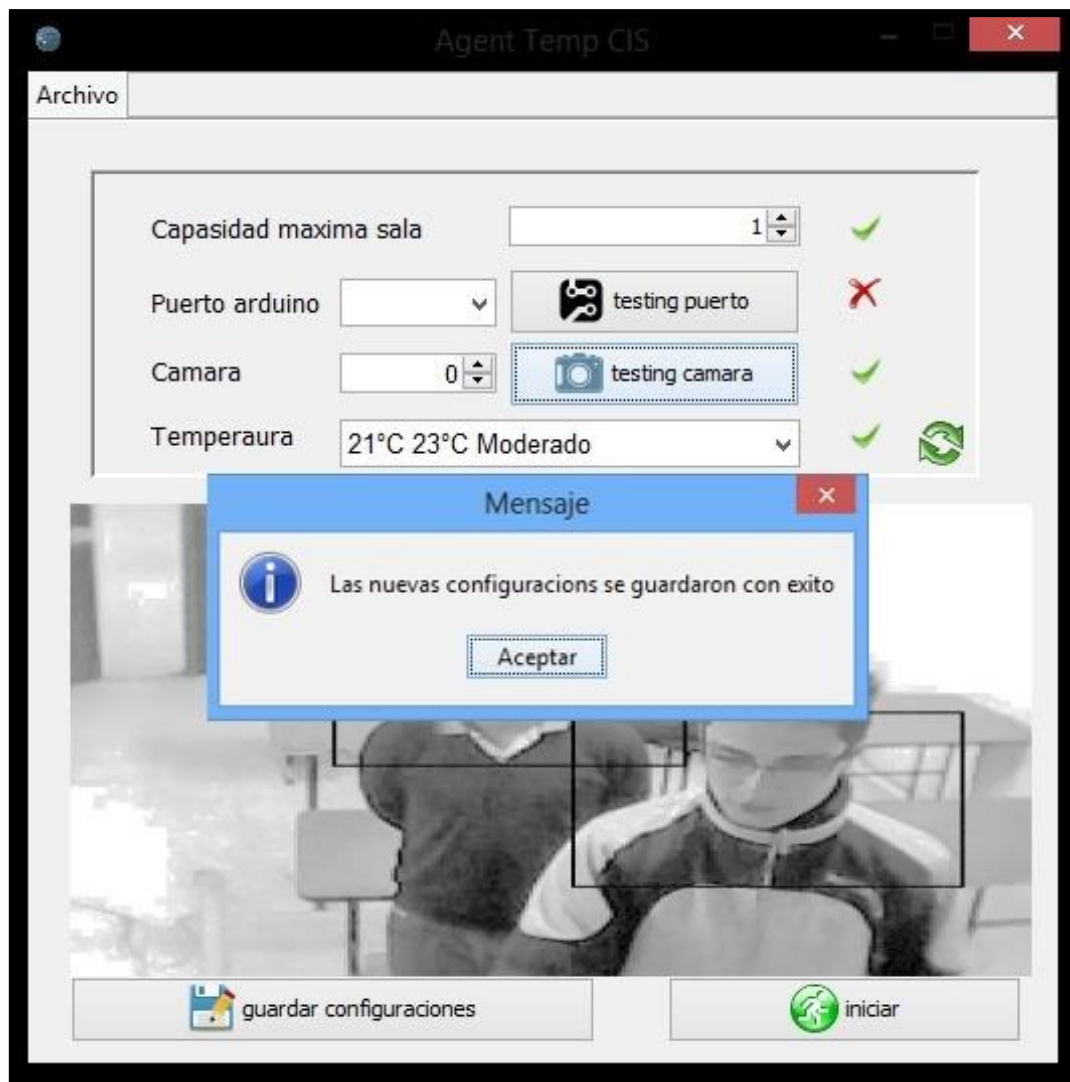


Figura 23: Parámetros de configuración.

Testeo de parámetros de Configuración: Se comprueba que los puertos tanto como para la cámara y el Arduino estén correctamente conectados y en funcionamiento para continuar con la ejecución del sistema caso contrario no se ejecutara el sistema.

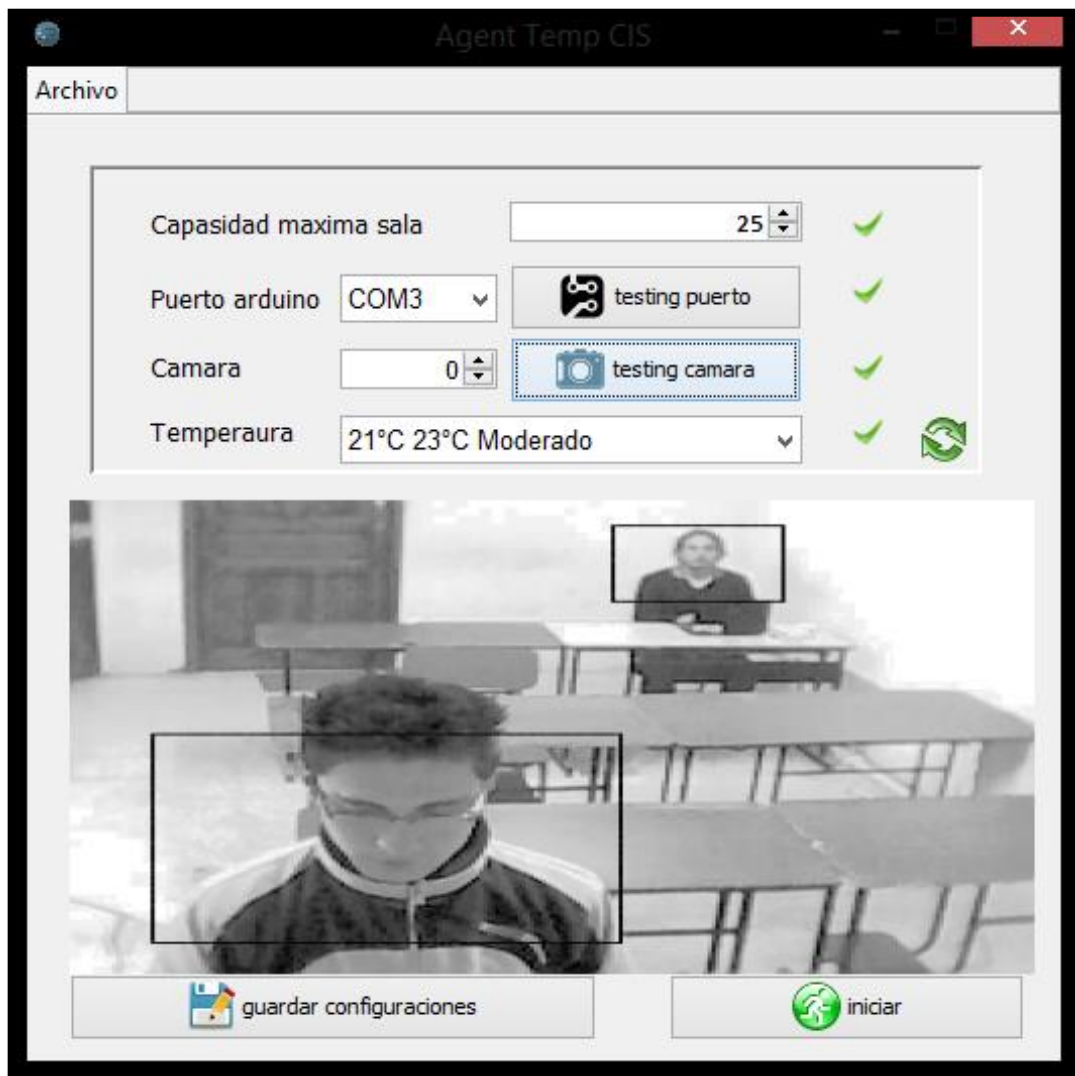


Figura 24: Testing de parámetros de configuración.

Pantalla de Información de temperatura: Se muestra los datos obtenidos por medio del sensor de temperatura y la webcam, además de la temperatura ambiente que es proporcionada a la sala.

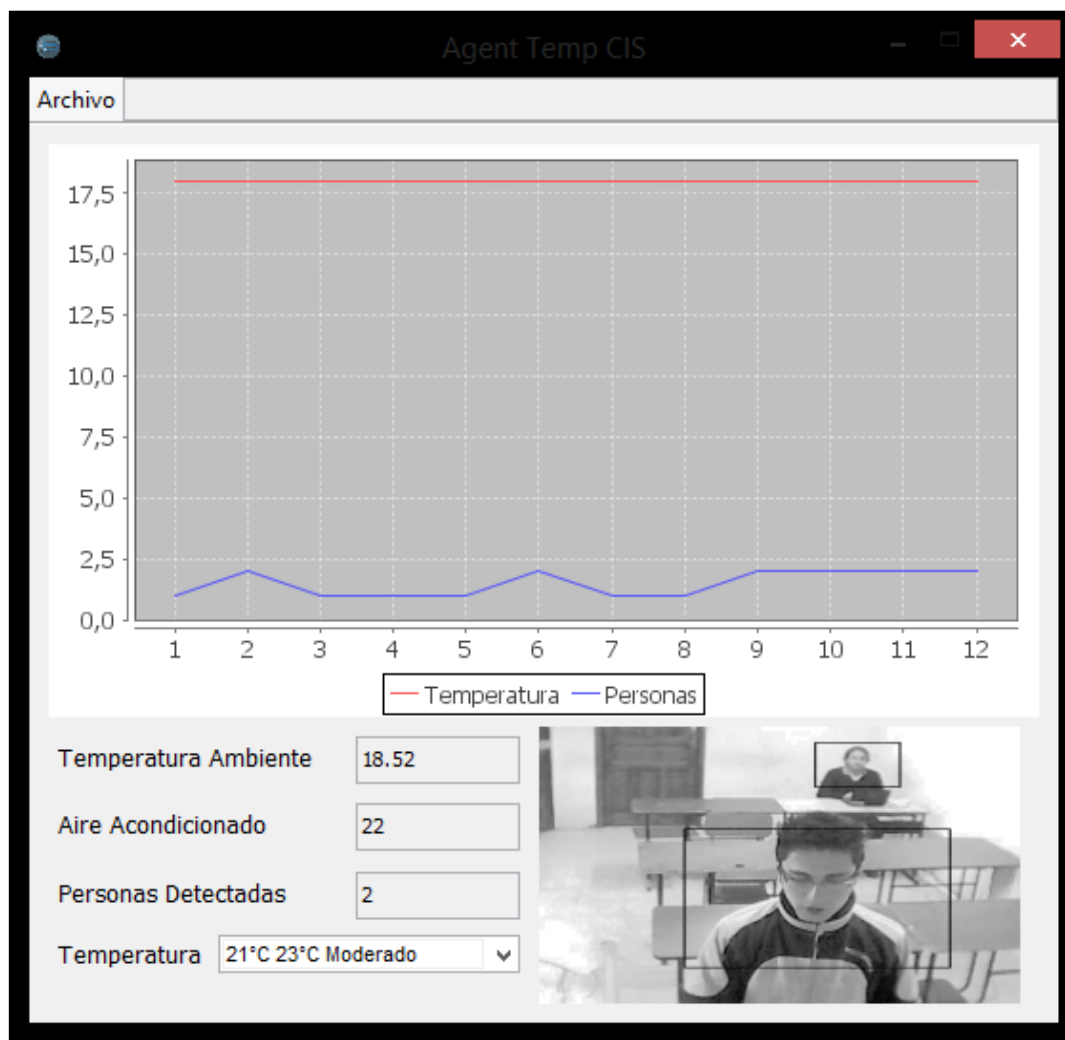



Figura 25: Pantalla de Información.

6.8.5 Pruebas de Caja Blanca

Creación de los Agentes:

AgentControlTempEnviroment: Creación del agente deliberativo encargado de ejecutar las acciones correspondientes para controlar el aire acondicionado.

AgentDetectionPersonsAndTemp: Creación del agente reactivo encargado de obtener los datos del sensor de temperatura y de la webcam. (Ver figura 26)



```
public void iniciarAgente()
{
    crearMainContainer();
    try
    {
        agentController = agentContainer.createNewAgent (
            "AgentDetectionPersonsAndTemp",
            AgentDetectionPersonsAndTemp.class.getName(), null);

        System.out.println("AgentDetectionPersonsAndTemp creado correc
        agentController.start();

        agentController = agentContainer.createNewAgent (
            "AgentControlTempEnviroment",
            AgentControlTempEnviroment.class.getName(), null);

        System.out.println("AgentControlTempEnviroment creado correcta
        agentController.start();

    } catch (StaleProxyException e)
    {
        System.out.println(this.getClass().getName() + ">>" + e.getCau
            + " " + e.getMessage());
    }
}
```

```
jun 26, 2015 1:05:17 PM jade.incp.ieap.LEAPInitManage
INFORMACIÓN: Listening for intra-platform commands on
- jicp://10.20.51.169:1099

jun 26, 2015 1:05:17 PM jade.core.BaseService init
INFORMACIÓN: Service jade.core.management.AgentManage
jun 26, 2015 1:05:17 PM jade.core.BaseService init
INFORMACIÓN: Service jade.core.messaging.Messaging in
jun 26, 2015 1:05:17 PM jade.core.BaseService init
INFORMACIÓN: Service jade.core.resource.ResourceManag
jun 26, 2015 1:05:17 PM jade.core.BaseService init
INFORMACIÓN: Service jade.core.mobility.AgentMobility
jun 26, 2015 1:05:17 PM jade.core.BaseService init
INFORMACIÓN: Service jade.core.event.Notification ini
jun 26, 2015 1:05:17 PM jade.mtp.http.HTTPServer <ini
INFORMACIÓN: HTTP-MTP Using XML parser com.sun.org.ap
jun 26, 2015 1:05:17 PM jade.core.messaging.Messaging
INFORMACIÓN: MTP addresses:
http://DesertPC:7778/acc
jun 26, 2015 1:05:17 PM jade.core.AgentContainerImpl
INFORMACIÓN: -----
Agent container Main-Container@10.20.51.169 is ready.
-----
AgentDetectionPersonsAndTemp creado correctamente
AgentControlTempEnviroment creado correctamente
```

Figura 26: Creación de los agentes

Integración del Sistema con la placa Arduino

ComunicacionArduino: Se realiza con los tres atributos importantes como son:

- serialPort, input, output.
- writeData: Método que permite escribir los datos en la placa Arduino
- serialEvent: Método que permite la lectura de los datos de los sensores.

<pre> public synchronized void enviarDatosAlArduino(String data) { System.out.println("Enviando datos de arduino"); try { output.write(data.getBytes()); } catch (Exception e) { System.out.println(this.getClass().getName() + ">>" + e.getCau + " " + e.getMessage() + " " + e.getLocalizedMessage()); } } /** * Cada que el puerto serial reciba datos este método los leerá y envi * esa información a la variable temperatura. */ @Override public synchronized void serialEvent(SerialPortEvent onEvent) { System.out.println("Recibiendo datos de arduino"); if (onEvent.getEventType() == SerialPortEvent.DATA_AVAILABLE) { try { String lecturaArduino = input.readLine(); temperatura = Integer.parseInt(lecturaArduino.substring(0, lecturaArduino.indexOf("."))); } catch (Exception e) { } } } </pre>	<pre> jun 26, 2015 1:03:13 PM jade.imtp.leap.LEAPI INFORMACIÓN: Listening for intra-platform co - jicp://10.20.51.169:1099 jun 26, 2015 1:03:13 PM jade.core.BaseServic INFORMACIÓN: Service jade.core.management.Ag jun 26, 2015 1:03:13 PM jade.core.BaseServic INFORMACIÓN: Service jade.core.messaging.Mes jun 26, 2015 1:03:13 PM jade.core.BaseServic INFORMACIÓN: Service jade.core.resource.Reso jun 26, 2015 1:03:13 PM jade.core.BaseServic INFORMACIÓN: Service jade.core.mobility.Agen jun 26, 2015 1:03:13 PM jade.core.BaseServic INFORMACIÓN: Service jade.core.event.Notific jun 26, 2015 1:03:13 PM jade.mtp.http.HTTPSe INFORMACIÓN: HTP-MTP Using XML parser com.s jun 26, 2015 1:03:13 PM jade.core.messaging. INFORMACIÓN: MTP addresses: http://DesertPC:7778/acc jun 26, 2015 1:03:13 PM jade.core.AgentConta INFORMACIÓN: ----- Agent container Main-Container@10.20.51.169 ----- Recibiendo datos de arduino Enviando datos de arduino Recibiendo datos de arduino Enviando datos de arduino Recibiendo datos de arduino Enviando datos de arduino Recibiendo datos de arduino Enviando datos de arduino Recibiendo datos de arduino </pre>
---	--

Figura 27: Comunicación Arduino.

Comportamientos de los agentes.

Se verifica que los comportamientos asignados a cada agente estén operando de forma correcta debe ver en pantalla toda la información de los parámetros que va a manejar el comportamiento, como los datos obtenidos a través del sensor de temperatura y el número de personas obtenido a través de la webcam.

Funcionamiento de los agentes.

AgentDetectionPersonsAndTemp: Durante la ejecución de este agente se obtendrá la temperatura ambiente proveniente del sensor de temperatura y se obtendrá el número de personas detectadas en la sala de conferencias a través de la webcam.

AgentControlTempEnviroment: Durante la ejecución de este agente se procesara las acciones correspondientes a través de la placa Arduino que surgirán efecto en el control del aire acondicionado. (Ver figura 28)

```

private ACLMessage mensaje;

@Override
public void action()
{
    try
    {
        Thread.sleep(1000);
    } catch (InterruptedException e)
    {
    }

    informationSensors.setNumeroPersonas(sensorCamara
        .obtenerNumeroPersonasDetectadas());
    informationSensors.setTemperaturaAmbiente(sensorTemperatura
        .verTemperatura());

    System.out.println(informationSensors);
    try
    {
        mensaje.setContentObject(informationSensors);
    } catch (IOException e)
    {
    }

    this.agent.send(mensaje);
}

```

```

INFORMACION: Service jade.core.mobility.AgentMobility initial
jun 26, 2015 12:50:25 PM jade.core.BaseService init
INFORMACION: Service jade.core.event.Notification initialized
jun 26, 2015 12:50:25 PM jade.mtp.http.HTTPServer <init>
INFORMACION: HTTP-MTP Using XML parser com.sun.org.apache.xer
jun 26, 2015 12:50:25 PM jade.core.messaging.MessagingService
INFORMACION: MTP addresses:
http://DesertPC:7778/acc
jun 26, 2015 12:50:25 PM jade.core.AgentContainerImpl joinPla
INFORMACION: -----
Agent container Main-Container@192.168.1.124 is ready.
-----
InformationSensors [numeroPersonas=0, temperaturaAmbiente=17]
InformationSensors [numeroPersonas=0, temperaturaAmbiente=17]
InformationSensors [numeroPersonas=0, temperaturaAmbiente=17]
InformationSensors [numeroPersonas=0, temperaturaAmbiente=17]
InformationSensors [numeroPersonas=0, temperaturaAmbiente=17]
InformationSensors [numeroPersonas=0, temperaturaAmbiente=17]
InformationSensors [numeroPersonas=0, temperaturaAmbiente=16]
InformationSensors [numeroPersonas=1, temperaturaAmbiente=17]
InformationSensors [numeroPersonas=1, temperaturaAmbiente=17]
InformationSensors [numeroPersonas=0, temperaturaAmbiente=17]
InformationSensors [numeroPersonas=0, temperaturaAmbiente=17]
InformationSensors [numeroPersonas=1, temperaturaAmbiente=17]
InformationSensors [numeroPersonas=2, temperaturaAmbiente=17]
InformationSensors [numeroPersonas=0, temperaturaAmbiente=17]
InformationSensors [numeroPersonas=1, temperaturaAmbiente=17]
InformationSensors [numeroPersonas=2, temperaturaAmbiente=17]
InformationSensors [numeroPersonas=4, temperaturaAmbiente=17]
InformationSensors [numeroPersonas=1, temperaturaAmbiente=17]

```

Figura 28: Funcionamiento de los agentes

6.8.6 Resultados de pruebas de caja blanca y caja negra

PRUEBA	RESULTADO	OBSERVACIONES
Parámetros de Configuración	Exitoso	Se agrega los parámetros correspondientes para el inicio del funcionamiento del sistema.
Testeo de los parámetros de Configuración	Exitoso	Se comprueban que los puertos para la webcam y el Arduino estén disponibles para su correcta ejecución.
Pantalla de Información	Exitoso	Muestra los datos obtenidos tanto del sensor como de la cámara.
Integración del sistema con la placa Arduino	Exitoso	A través de la clase ComunicacionArduino se leen y se escriben datos desde el sistema hacia la placa Arduino.

Creación de los Agentes	Exitoso	Se crean correctamente cada uno con su propio controlador.
Ejecución de los Agentes	Exitoso	Se ejecutan en diferentes contenedores, lo cual no afecta el funcionamiento.
Ejecución de comportamientos de los Agentes	Exitoso	Cada agente cumple con su propio comportamiento.

Tabla 16: Resultados de Pruebas.

6.9 Pruebas Funcionales

Las pruebas funcionales se llevaron a cabo en una sala de conferencias en la cual nos permitió mostrar la funcionalidad y eficacia del sistema, a continuación se describen los procesos llevados a cabo durante la ejecución del sistema.

Escenario de pruebas.

La figura 29 muestra el escenario de pruebas con los elementos que se utilizaron para llevar a cabo las pruebas funcionales del sistema. Esto con el fin de obtener los resultados esperados del sistema.

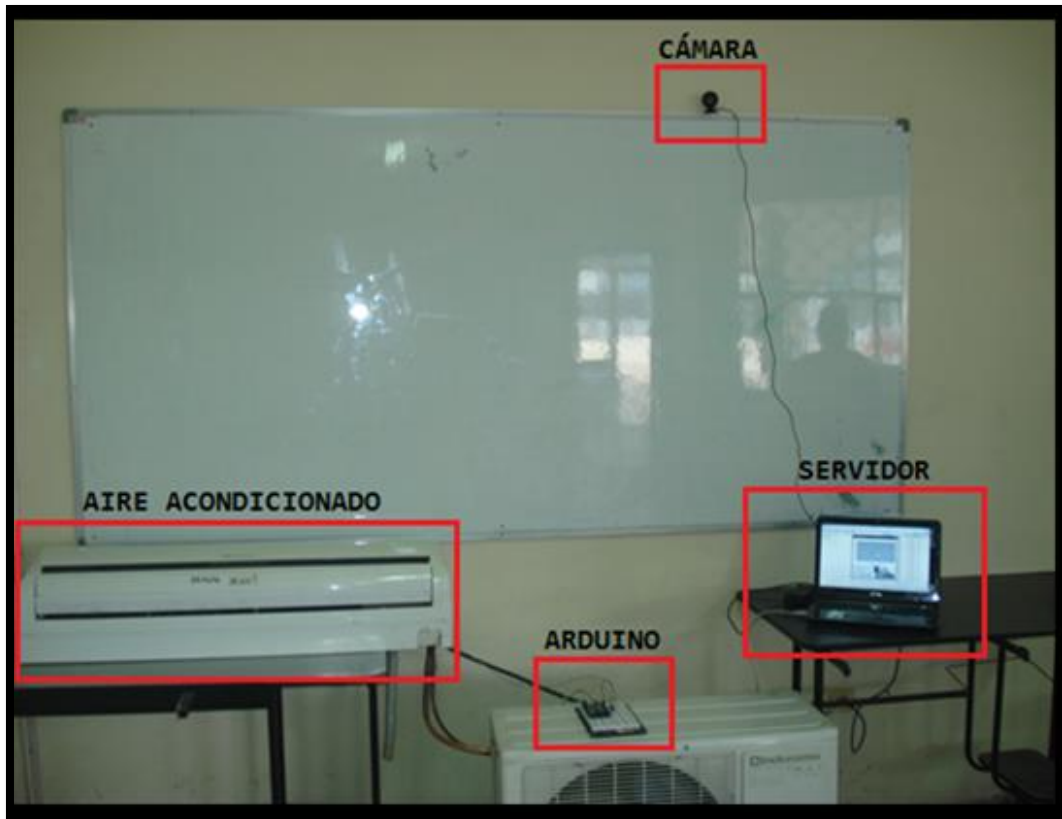


Figura 29: Escenario de Pruebas Funcionales

Para empezar se hizo una prueba de la webcam (ver figura 30) la cual se la colocó en un lugar estratégico donde tenga una vista panorámica de toda la sala de conferencias, capaz de que pueda detectar todos las personas que se encuentren en dicha sala.



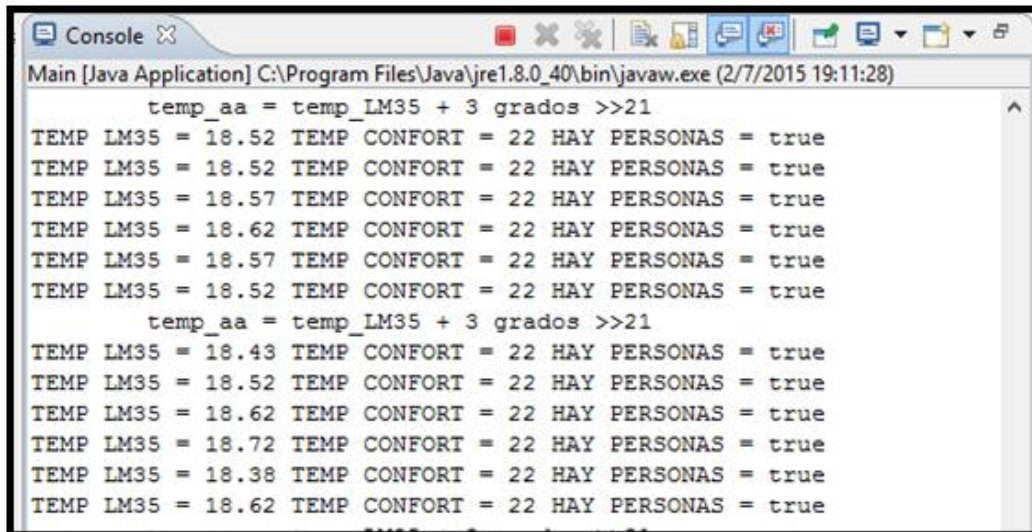
Figura 30: Webcam



Figura 31: Personas Detectadas

Dando como resultado la detección correcta de las personas presentes en la sala como se muestra en la figura 31.

El siguiente paso es comprobar que la información es correctamente procesada por el servidor, para ello se capturó la pantalla (ver figura 32) con los datos obtenidos a través de la cámara y el sensor de temperatura.



```
Console
Main [Java Application] C:\Program Files\Java\jre1.8.0_40\bin\javaw.exe (2/7/2015 19:11:28)
    temp_aa = temp_LM35 + 3 grados >>21
TEMP LM35 = 18.52 TEMP CONFORT = 22 HAY PERSONAS = true
TEMP LM35 = 18.52 TEMP CONFORT = 22 HAY PERSONAS = true
TEMP LM35 = 18.57 TEMP CONFORT = 22 HAY PERSONAS = true
TEMP LM35 = 18.62 TEMP CONFORT = 22 HAY PERSONAS = true
TEMP LM35 = 18.57 TEMP CONFORT = 22 HAY PERSONAS = true
TEMP LM35 = 18.52 TEMP CONFORT = 22 HAY PERSONAS = true
    temp_aa = temp_LM35 + 3 grados >>21
TEMP LM35 = 18.43 TEMP CONFORT = 22 HAY PERSONAS = true
TEMP LM35 = 18.52 TEMP CONFORT = 22 HAY PERSONAS = true
TEMP LM35 = 18.62 TEMP CONFORT = 22 HAY PERSONAS = true
TEMP LM35 = 18.72 TEMP CONFORT = 22 HAY PERSONAS = true
TEMP LM35 = 18.38 TEMP CONFORT = 22 HAY PERSONAS = true
TEMP LM35 = 18.62 TEMP CONFORT = 22 HAY PERSONAS = true
```

Figura 32: Datos del Servidor

Luego de esta prueba se verificó que el diseño del circuito en la placa Arduino esté correctamente conectado para poder enviar los datos al aire acondicionado como se muestra en la figura 33.

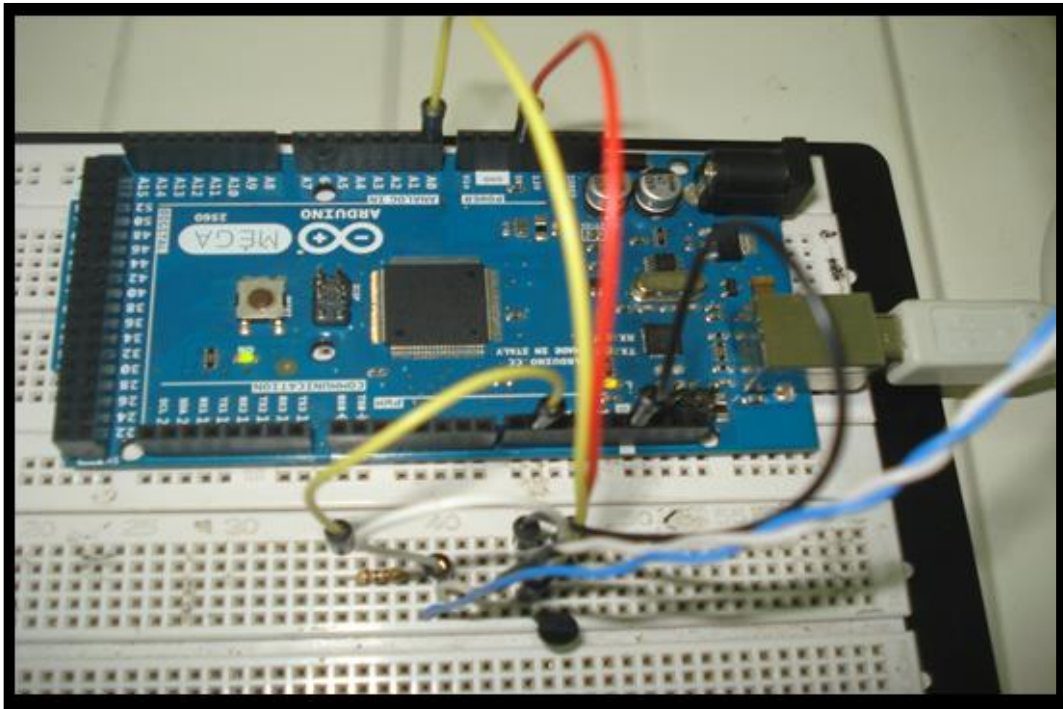


Figura 33: Circuito en la placa Arduino

Por último se comprobó la funcionalidad del aire acondicionado enviando datos desde la placa Arduino a través del LED IR el cual fijó una temperatura estable como se puede ver en la figura 34.



Figura 34: Temperatura fijada en el Aire Acondicionado

6.10 Pruebas del Sistema

Finalmente se han realizado las pruebas del sistema con el fin de tener integradas todas las características del sistema y comprobar su funcionalidad cumpliendo con los requerimientos planteados.

La figura 35 muestra la pantalla de inicio del sistema donde se configura los datos iniciales con los que se va a ejecutar el sistema.

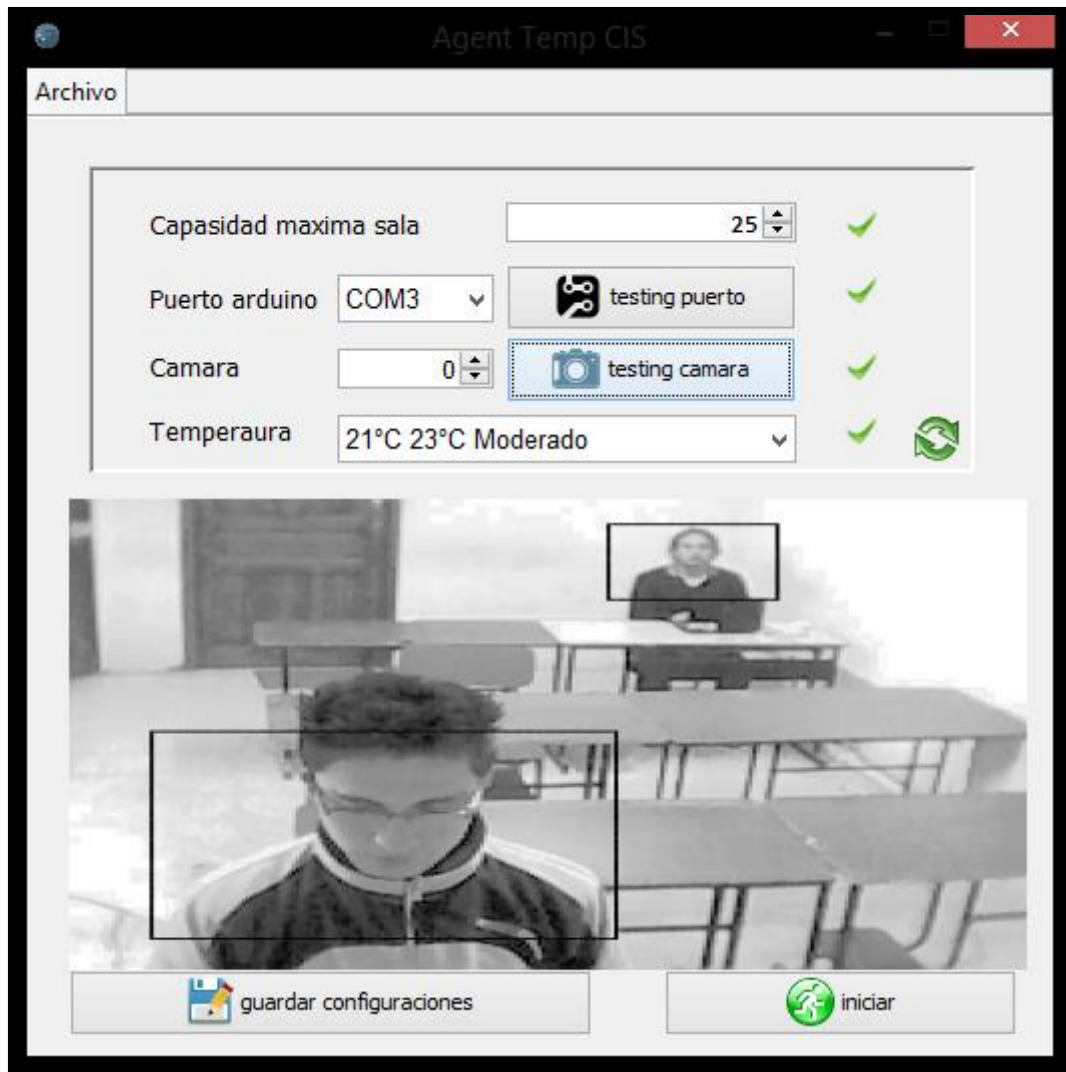


Figura 35: Inicio del sistema.

Para cumplir los requerimientos funcionales se determinó que el usuario puede modificar los parámetros iniciales, como la capacidad máxima de personas en la sala, la temperatura confort deseada y los puertos en los cuales van a funcionar la webcam y la placa Arduino.

La Figura 36 muestra las configuraciones establecidas por el usuario, comprobando cada campo que esté correctamente seleccionado y testeando que se encuentren conectados la webcam y la placa Arduino, para luego guardar las nuevas configuraciones y dar paso a la ejecución del sistema, de lo contrario el sistema no iniciará.

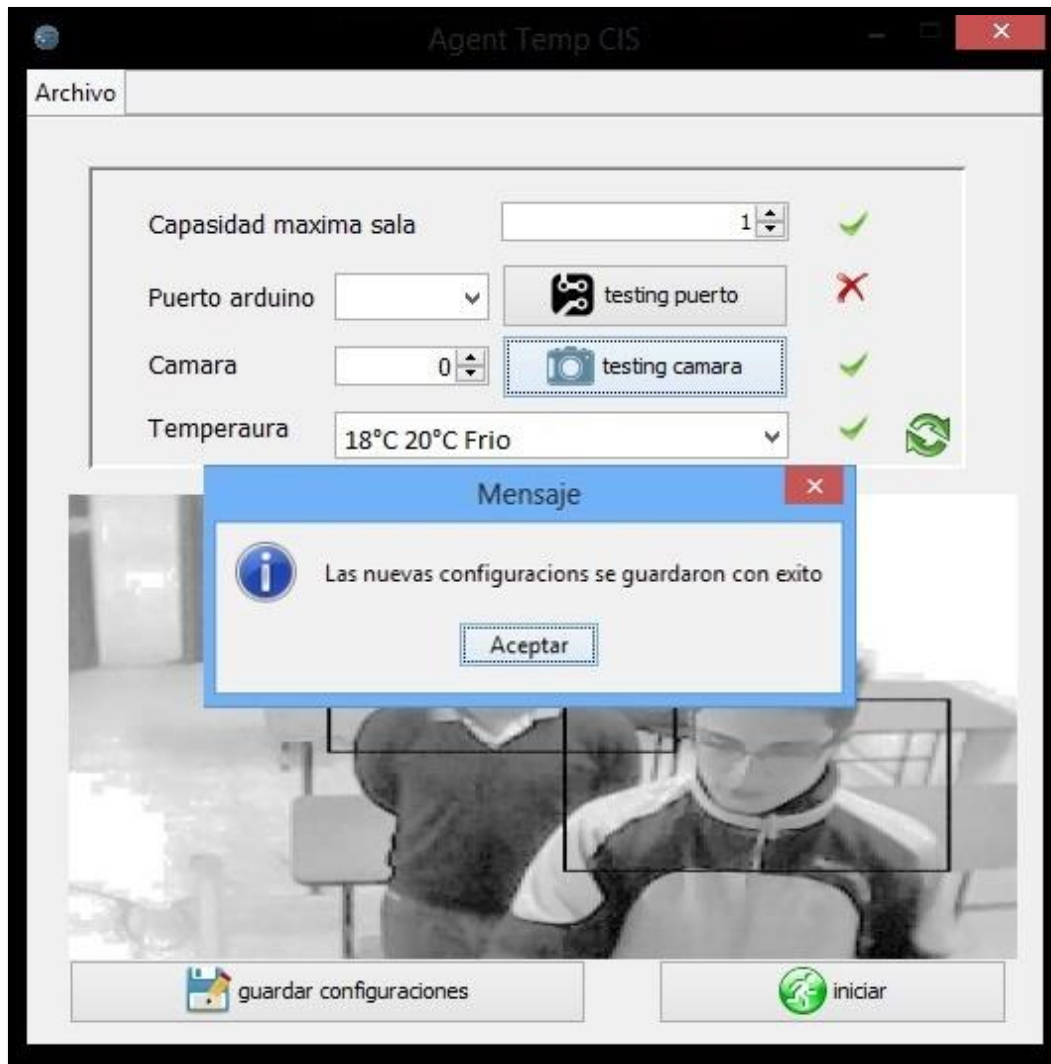


Figura 36: Guardar nuevas configuraciones del sistema.

Como se puede observar la nueva temperatura fijada es de 18°C a 20°C y se hecha a correr el sistema, estos datos son obtenidos y procesados por el sistema como se muestra en la figura 37 determinando si la temperatura que se obtiene a través del sensor LM35 es mayor o menor a la establecida y determinando si existen o no personas en la sala, para de acuerdo a ello fijar la nueva temperatura confort.

```
Console
Main [Java Application] C:\Program Files\Java\jre1.8.0_40\bin\javaw.exe (2/7/2015 19:11:28)
temp_aa = temp_LM35 - 3 grados >> 19
TEMP LM35 = 22.52 TEMP CONFORT = 19 HAY PERSONAS = true
TEMP LM35 = 22.52 TEMP CONFORT = 19 HAY PERSONAS = true
TEMP LM35 = 22.57 TEMP CONFORT = 19 HAY PERSONAS = true
TEMP LM35 = 22.62 TEMP CONFORT = 19 HAY PERSONAS = true
TEMP LM35 = 22.57 TEMP CONFORT = 19 HAY PERSONAS = true
TEMP LM35 = 22.52 TEMP CONFORT = 19 HAY PERSONAS = true
```

Figura 37: Datos procesados por el sistema.

En la figura 38 se muestra la temperatura inicial obtenida por el sensor LM35 de 22.5°C y al mismo tiempo el sistema detecta personas en la sala, entonces se fija la temperatura confort de 19°C en el aire acondicionado. (Ver figura 39)

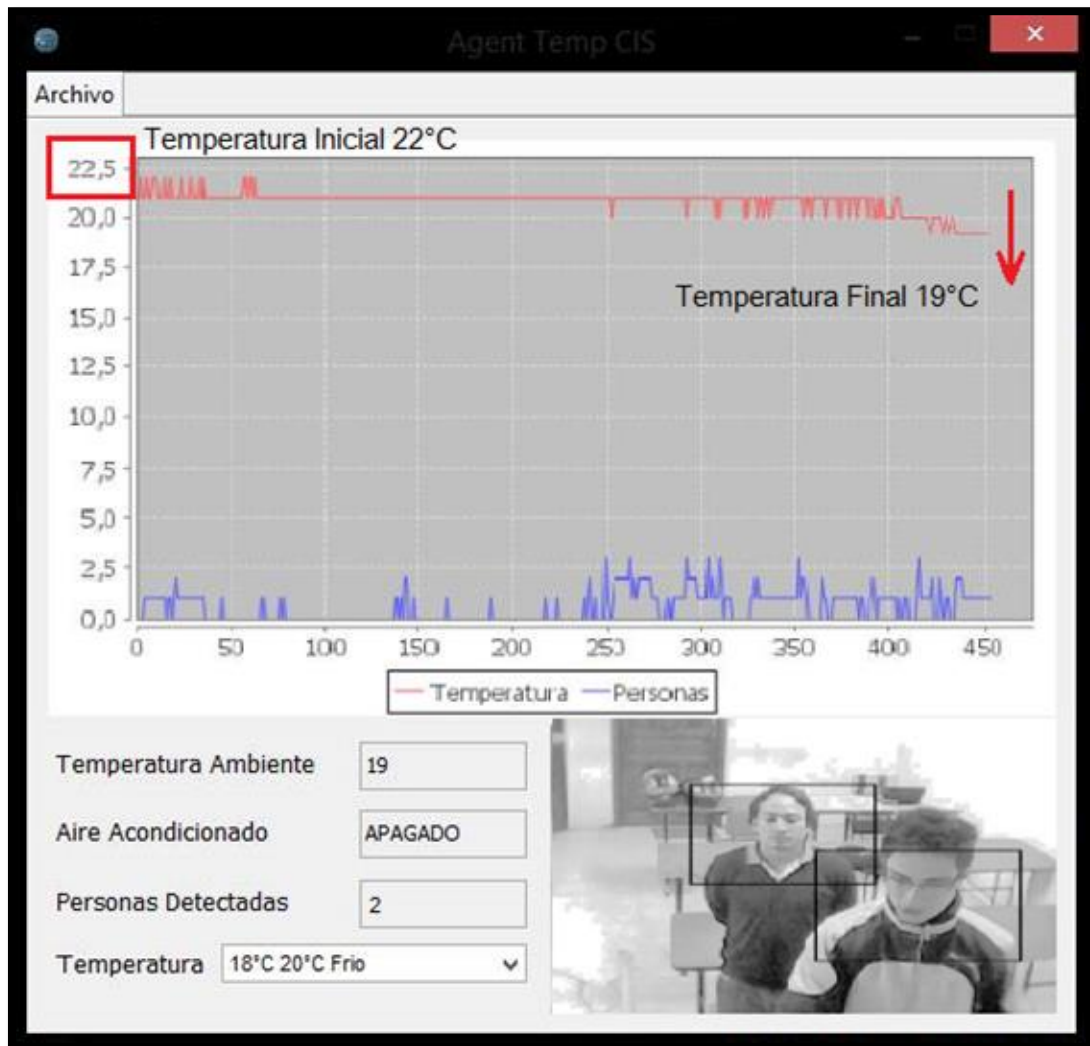


Figura 38: Pantalla de información

Después de haber fijado la temperatura confort a 19°C el sistema a través del sensor LM35 nuevamente obtendrá la temperatura ambiente, en este caso es ya de 19°C, entonces el sistema apagará el aire acondicionado debido a que la sala ya se encuentra en la temperatura requerida.

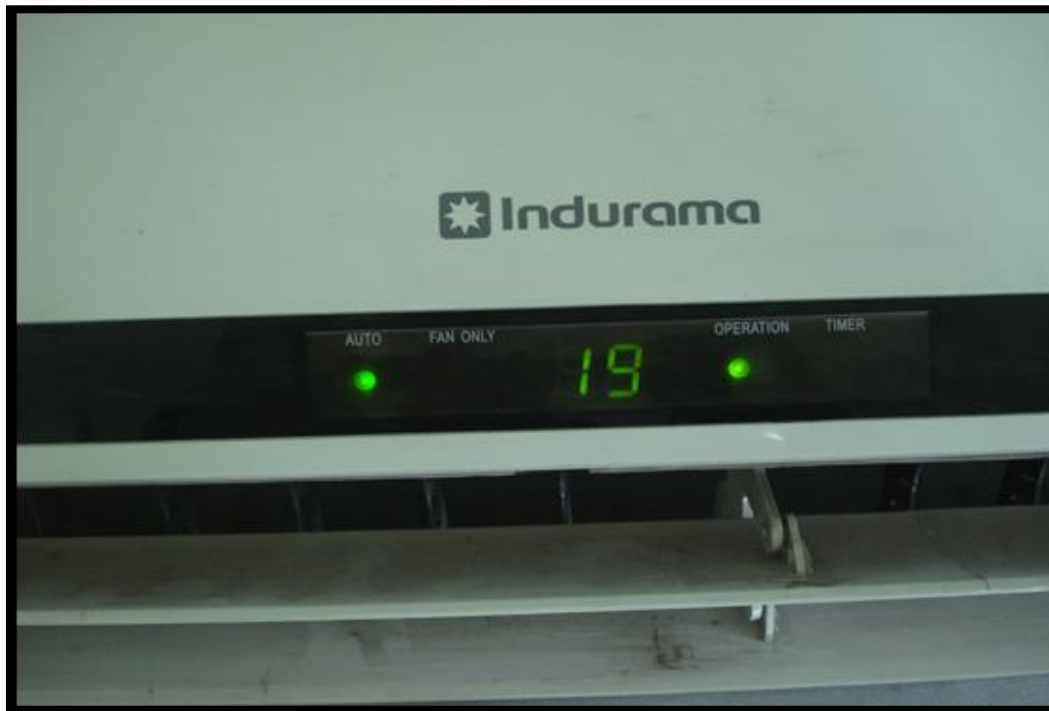


Figura 39: Temperatura del Aire acondicionado

7. Discusión

Los resultados obtenidos muestran el control automático del aire acondicionado a través de la placa Arduino activando o desactivando el mismo, además de modificar la temperatura que este expelle de acuerdo a la información obtenida y procesada a través del sensor de temperatura y la webcam.

Para obtener el número de personas que se encuentran en la sala de conferencia se hizo uso de la librería OpenCV para el procesamiento de las imágenes capturadas por la webcam.

El uso de agentes inteligentes permitió interactuar de una mejor manera con el ambiente a trabajar el cual es la sala de conferencias, y el contexto del sistema se da por él, las personas que se encuentran en la misma y la temperatura que ésta posee, por ende, el uso de un agente reactivo encargado de la detección de personas y la captura de temperatura ambiente.

El agente deliberativo es el encargado del control del aire acondicionado el cual conoce los siguientes aspectos.

El reglamento de seguridad y salud de los trabajadores y mejoramiento del medio ambiente del trabajo de Ecuador dice que la temperatura en trabajos sedentarios deberá ser de entre los 17° C y 27° C. Con esto consideramos que una sala de conferencias se lo puede catalogar como trabajo sedentario, y como el rango posee una variación de 9 grados Celsius se ha considerado clasificar el mismo en temperaturas diferentes las cuales son frío, templado y cálido. Como se muestra en la tabla temperaturas agente_temp_cis.

Nombre	Rango	Confort
Frio	18° C a 20° C	19° C
Templado	21° C a 23° C	22° C
Cálido	24° C a 26° C	25° C

Tabla 16: Temperaturas agente_temp_cis

El reglamento de seguridad y salud de los trabajadores y mejoramiento del medio ambiente del trabajo de Ecuador nos dice que los cambios bruscos de temperatura pueden provocar daños a la salud, estos cambios bruscos son considerados cambios de temperatura que varían entre los 5 grados Celsius. Contemplando esto decidimos hacer los cambios de temperatura con una variación máxima de tres grados en la salida del aire acondicionado, suponiendo que cualquier persona podrá estar demasiado cerca del mismo.

Otro conocimiento de nuestro agente deliberativo es el de los tres tipos de temperaturas que deberá manejar: la temperatura del LM35 (temperatura de la sala de conferencias), temperatura del aire acondicionado (temperatura de salida del aire acondicionado) y temperatura confort (temperatura que queremos obtener en la sala de conferencias).

Con el fin de ahorrar energía consideramos que existen dos motivos para encender el aire acondicionado, la primera: si en la sala de conferencias está siendo ocupada y la segunda: si la temperatura ambiente es menor o mayor a la temperatura confort establecida previamente.

Las acciones a ejecutar por este agente son las siguientes que se detallan en el siguiente algoritmo:

Si la temperatura del LM35 es igual a la temperatura confort, entonces el aire acondicionado deberá ser o permanecer apagado.

Si la temperatura del LM35 es igual a la temperatura confort más un grado o menos un grado, entonces la temperatura del aire acondicionado será igual a la temperatura confort.

Si la temperatura del LM35 es menor a la temperatura confort, entonces la temperatura del aire acondicionado será igual a la temperatura confort más 3 grados.

Por último si la temperatura del LM35 es mayor a la temperatura confort, entonces la temperatura del aire acondicionado será igual a la temperatura confort menos tres grados.

Estas condiciones nos aseguran una cobertura de todas las temperaturas posibles en el ambiente sin salirnos del grado de cambio brusco citado anteriormente.

7.1 Desarrollo de la propuesta alternativa

Los objetivos específicos que a continuación se detallan así como el objetivo general fueron cumplidos en su totalidad gracias al correcto análisis de información y al práctico desarrollo de las correspondientes fases planteadas desde un inicio.

7.1.1 Objetivo Específico 1: “Diseñar y programar el módulo de visión artificial, encargado de la detección y conteo de personas en una habitación.”

EL cumplimiento de este objetivo se lo consigue gracias a la librería OpenCV, ya que la misma posee los algoritmos de reconocimiento, en primera instancia se planteó el uso de los dos filtros para el reconocimiento de personas en una sala de conferencias, los cuales fueron el filtro de reconocimiento facial de frente y el de perfil de una persona, esto fue analizado ya que en una sala de conferencias las personas tienen su mirada puesta en el expositor, y su ángulo mayor de giro se lo rescataría con el filtro de detección de perfil que opencv brinda, esto en la práctica no se cumplió con el objetivo como esperábamos, lo cual incentivo a la búsqueda de un nuevo filtro.

OpenCV además de los filtros mencionados, posee una lista limitada de ellos por fortuna uno de estos filtros dio solución a este problema el cual detecta personas considerando la parte superior del cuerpo humano. La figura 40 muestra la detención de personas en la sala prueba, usando filtros de detección de perfil y de frente. Mientras que en la figura 41 se muestra la detección de personas en la sala prueba, usando el filtro de detección de la parte superior del cuerpo.



Figura 40: Filtro de perfil y de frente

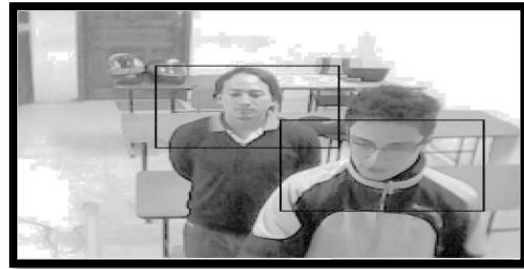


Figura 41: Filtro parte superior cuerpo

A simple vista se observa que los filtros de perfil y de frente son muy exigentes pues para que estos funcionen correctamente se debe ver a la webcam, en cambio el filtro de detección de la parte superior del cuerpo, es todo lo contrario, pues no se requiere ver a la webcam para que este funcione correctamente. Además de permitirnos un mayor alcance de detección como muestran las figuras 42 y 43 en la primera figura usando el filtro de la parte superior del cuerpo y la siguiente usando el filtro de perfil y de frente de una persona.



Figura 42: Distancia filtro parte superior cuerpo



Figura 43: Distancia filtro de perfil y de frente

EL filtro de detección de la parte superior del cuerpo además de detectar personas eficientemente evitó el doble procesamiento que en primera instancia pretendíamos realizar al pasar dos filtros.

7.1.2 Objetivo Específico 2: “Diseñar e integrar un circuito con sensores y actuadores en la placa Arduino encargada de la detección de temperatura ambiente y activación del aire acondicionado.”

El diseño del circuito que contará con un led de infrarrojo para la activación del aire acondicionado y el sensor de temperatura LM35, se realizó en el software libre fritzing. Esto permitió tener una idea clara de cómo conectar correctamente estos componentes, ya que no se puede hacerlo directamente en la placa Arduino, pues esta puede llegar a recibir una descarga y ser estropeada.

Como el diseño en el software fritzing es muy claro pasarlo al físico, es aún más sencillo, esto ya, para poder testear si estamos recibiendo y enviando datos del LM35 y del LED de infrarrojo respectivamente.

Como el LED de infrarrojo envía señales en hexadecimal, debimos buscar la forma de decodificar las mismas que el control remoto del aire acondicionado usa, en internet no se pudo encontrar los códigos del control remoto que estamos usando, por lo que, la solución a este inconveniente fue el diseño de un decodificador de infrarrojos.

El componente encargado de decodificar las señales infrarrojas es el IR Receiver, el cual es muy sencillo conectar al Arduino, la librería IRemote.h nos brindó en su documentación el código necesario para decodificar las señales de cualquier control remoto usando este componente. Después de implementar este código lo siguiente es subirlo al Arduino. Una vez hecho esto se habilita el monitor serial en el ID Arduino, acercamos el control remoto al IR Receiver presionamos la tecla la cual deseamos conocer su hexadecimal, y en la pantalla que habíamos habilitado anteriormente se mostrará el código, y el número de bits que utiliza, como se muestra en la figura 44.

```
COM3 (Arduino Mega or Mega 2560)
1
For IR Scope:
+4450 -4200 +650 -1500 +650 -450 +600 -1500 +650 -1450 +650 -450 +650 -450 +600 -1500 +650 -450 +650 -450 +600 -1500 +600 -500 +600 -450 +650 -1500 +600 -1500 +600 -500 +600 -1500 +600 -
For Arduino sketch:
unsigned int raw[100] = {4450,4200,650,1500,650,450,600,1500,650,1450,650,450,650,450,600,1500,650,450,650,450,600,1500,600,500,600,450,650,1500,600,1500,600,500,650,145
I2send.sendRaw(raw,100,38);
2
For IR Scope:
+4450 -4200 +650 -1500 +650 -450 +600 -1500 +650 -1450 +650 -450 +650 -450 +600 -1500 +650 -450 +650 -400 +650 -1500 +650 -1450 +650 -450 +650 -1450 +650 -
For Arduino sketch:
unsigned int raw[200] = {4450,4200,650,1500,650,450,600,1500,650,1450,650,450,650,450,600,1500,650,450,650,450,600,1500,650,450,650,450,600,1500,650,1450,650,1500,600,1450,650,1500,600,145
I2send.sendRaw(raw,200,38);
3
For IR Scope:
+4450 -4250 +700 -1450 +650 -400 +700 -1450 +650 -1450 +650 -450 +650 -450 +600 -1500 +650 -450 +650 -400 +700 -1450 +650 -400 +650 -450 +650 -1500 +600 -1500 +650 -450 +650 -1450 +650 -
For Arduino sketch:
unsigned int raw[100] = {4450,4250,700,1450,650,400,700,1450,650,1450,650,450,650,450,600,1500,650,450,650,450,600,1450,650,400,650,450,650,1500,600,1500,650,450,650,1450,650,1450,650,1450,650,145
I2send.sendRaw(raw,100,38);
4
For IR Scope:
+4450 -4200 +650 -1450 +650 -450 +650 -1450 +650 -1500 +650 -400 +700 -400 +650 -1500 +600 -450 +650 -450 +650 -1450 +650 -450 +650 -450 +650 -1450 +650 -1500 +600 -450 +650 -1500 +600 -
For Arduino sketch:
unsigned int raw[200] = {4450,4200,650,1450,650,450,650,1450,650,1500,650,400,700,400,650,1500,600,450,650,450,650,1450,650,450,650,450,650,1450,650,1500,600,450,650,1450,650,1500,600,1500,650,145
I2send.sendRaw(raw,200,38);
5
For IR Scope:
+4400 -4250 +650 -1450 +650 -450 +650 -1450 +650 -1500 +650 -400 +650 -450 +650 -1450 +700 -400 +650 -450 +650 -1450 +650 -450 +650 -450 +650 -1450 +650 -1450 +650 -450 +650 -1450 +650 -
For Arduino sketch:
unsigned int raw[200] = {4400,4250,650,1450,650,450,650,1450,650,1500,650,400,650,450,650,1450,700,400,650,450,650,1450,650,450,650,450,650,1450,650,1450,650,450,650,1450,650,1500,650,1450,650,1450,650,145
I2send.sendRaw(raw,200,38);
Autoscroll
Nueva línea
9600 baudo
```

Figura 44: Monitor serial ID Arduino

Con estos datos ahora conocidos el sistema ya está en capacidad de enviar señales desde su Led Infrarrojo para controlar el aire acondicionado.

7.1.3 Objetivo Específico 3: “Integrar de módulo de visión artificial con el circuito en la placa Arduino.”

Para el cumplimiento de este objetivo se lo realizó mediante la comunicación con el puerto serial, esto gracias a la librería RXTX la misma que posee los jar y dll necesarios para la este tipo de comunicación desde JAVA.

Configurada la librería continuamos con nuestro sistema y damos paso a la creación de dos variables una para la escritura input y otro para la lectura output, las mismas que nos facilitan el manejo de los bytes escritos y leídos en el puerto serial. Ahora nuestra placa Arduino se la conecto al ordenador en el puerto USB ya que Arduino cuenta con un convertidor de serial USB. Por último se programa Arduino para que esté a la escucha de lo que se escriba en el puerto serial, además de que el mismo escriba los datos obtenidos por el LM35.

7.1.4 Objetivo Específico 4: “Realizar pruebas de control en una habitación a escala para probar su funcionalidad y fiabilidad.”

Para el cumplimiento de este objetivo se debió armar una sala de prueba la cual se muestra en la figura 45, en donde se aprecia el aire acondicionado, la placa arduino, la webcam y el servidor del sistema y en la figura 46 se muestra el campo de captura de la webcam.



Figura 45: Sala de prueba



Figura 46: Campo captura webcam

En la figura 47 se muestra como la temperatura de la sala empezó con 22° C, el objetivo fue llegar a los 19° C temperatura confort asignada al sistema; en la gráfica de la figura 47 se aprecia como al cabo de 7 minutos la temperatura se estabilizó llegando

al objetivo previsto, una vez obtenida esta temperatura el sistema toma la decisión de apagar el aire acondicionado con el propósito de ahorrar energía.

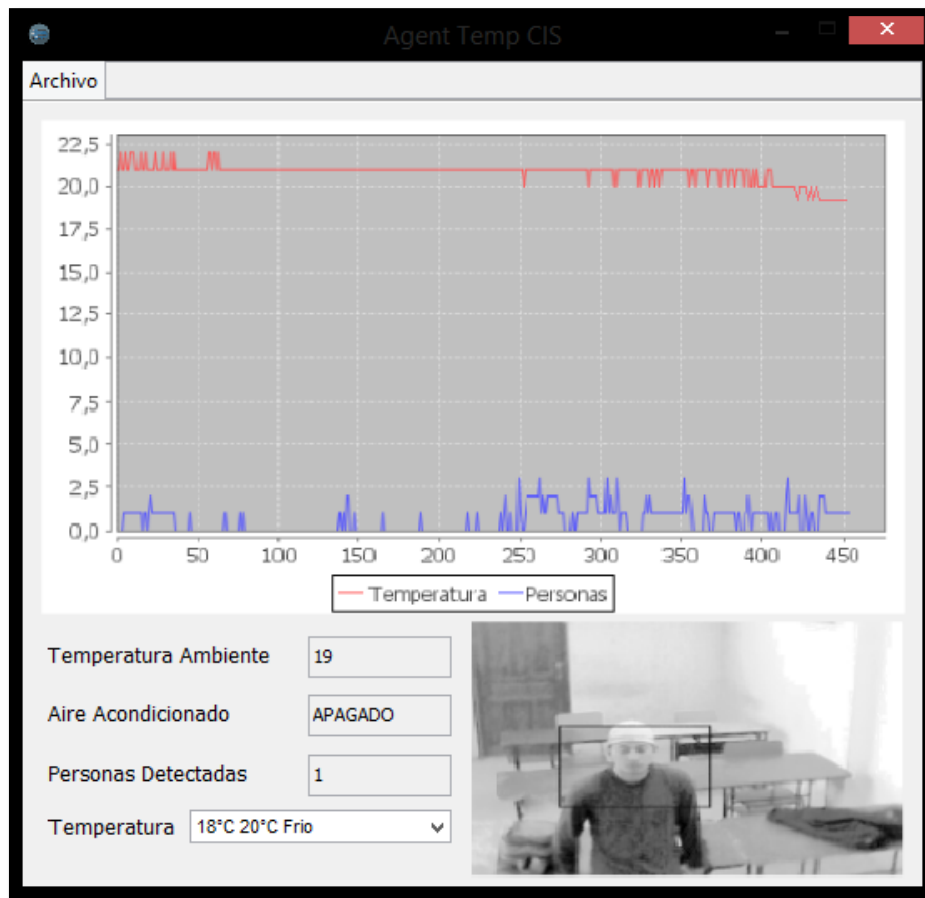


Figura 47: Prueba de funcionalidad y fiabilidad

La gráfica en la pantalla de ejecución del sistema (Figura 47), fue un elemento indispensable para poder probar la fiabilidad y funcionalidad del sistema, pues es, en ella que se visualizan los cambios de temperatura por los que ha pasado la sala. Además de mostrar la captura de la webcam con el fin de ubicar que zona de las sala de conferencia se está visualizando.

7.2 Valoración técnica económica ambiental

7.2.1 Valoración técnica económica

El presente proyecto se ha desarrollado con el uso de diferentes recursos como son: humanos, materiales y técnicos-tecnológicos. A continuación en la tabla 17 se detallan los recursos humanos que se intervinieron en el desarrollo del sistema, además de contar con el asesoramiento técnico del tutor, y el técnico encargado de la instalación del aire acondicionado.

Recursos	Número de horas	Costo por hora	Costo total
Juan Pablo Guamán	400	5.00	2000.00
Yandry Ramírez	400	5.00	2000.00
Ing. Tutor Luis Jácome	50	--	--
Técnico Carlos Moreno	6	5.00	30.00
Total			4030.00

Tabla 17: Recursos humanos

En la tabla 18 se indica detalladamente los recursos materiales que fueron necesarios para presentar los diferentes avances, entrevistas y el informe final con sus respectivos manuales del trabajo de titulación.

Descripción	Cantidad	valor/unitario	Valor Total
Webcam VGA	1	16.00	16.00

Aire acondicionado	1	450.00	450.00
Placa Arduino	1	80.00	80.00
Sensor temperatura	1	4.00	4.00
Sensor Infrarrojo	1	5.00	5.00
Impresión del circuito	1	3.00	3.00
Cables	9	0.15	1.35
Protoboard	1	20.00	20.00
LED IR	1	2.00	2.00
Total			581.35

Tabla 18: Recursos materiales

En la tabla 19 se muestra cada uno de los recursos técnicos y tecnológicos que se emplearon en el desarrollo del presente trabajo de titulación, como se puede observar existieron algunos que fueron adquiridos de forma gratuita.

Recursos	Cantidad	Valor unitario	Valor total
Hardware			
Computadora	2	800.00	1600.00

Impresora	1	60	60.00
Software			
Eclipse ID		Gratis	0.00
Arduino ID		Gratis	0.00
Gliffy diagrams		Gratis	0.00
Licencia creative commons		Gratis	0.00
Total			1660.00

Tabla 19: Recursos técnicos y tecnológicos

En la tabla 20 se indica el resumen del presupuesto del proyecto de trabajo de titulación, el cual se realizó en función de la estimación de los recursos humanos, materiales, técnicos y tecnológicos.

Recurso	Subtotal
Recursos humanos	4030.00
Recursos materiales	581.35
Recursos técnicos y tecnológicos	1660.00
Total	6271.35

Tabla 20: Presupuesto del proyecto

7.2.2 Valoración ambiental

El mal uso de los aires acondicionados, la activación permanente de estos, genera gastos a la empresa, consumiendo recursos valiosos, lo cual no solo afecta a la empresa económicamente, si no, lo que es peor, afecta al medio ambiente, para solventar este inconveniente, la utilización del software se considera de gran ayuda con ello evitando el mal uso del mismo lo cual permite disminuir el impacto producido. De esta forma queda justificada la valoración ambiental del presente trabajo de titulación.

8. Conclusiones

- El desarrollo del módulo de visión artificial se realizó con éxito gracias a la implementación de OpenCV; dado que esta librería brindó la posibilidad de un desarrollo limpio, además de ofrecer flexibilidad en el uso de filtros de detección e identificación de objetos.
- La detección de personas fue realizada satisfactoriamente, mediante la incorporación del filtro upperbody que obtuvo resultados positivos en las pruebas realizadas; por otro lado en la práctica se observó que los filtros frontalface y profileface necesitan mayor cantidad de píxeles, además de una mejor iluminación en la sala.
- En la práctica se comprobó que el sensor LM35 y el actuador IR cumplieron con las condiciones del diseño en conjunto con la placa Arduino mega 2560 y por ende la comunicación entre el sistema y el aire acondicionado funcionó de forma óptima sin causar inconveniente alguno.
- La comunicación con el puerto serial que posee Arduino, dio la posibilidad de una integración sencilla y estándar gracias a la librería RXTX la cual permitió al sistema ser multiplataforma.
- La librería JADE fue indispensable en el desarrollo del sistema para la implementación de agentes ya que con ella se dio al sistema una arquitectura clara y legible pues el sistema puede percibir su contexto (temperatura ambiente y detectar personas en una sala) además de ejecutar la acción de cambio de temperatura del aire acondicionado lo que hubiera tomado más tiempo con la programación Orientada a Objetos.
- El funcionamiento integral entre la detección de personas y el sensor de temperatura permitió una regulación precisa de la temperatura ambiente en la sala de conferencias presentando por sus características tiempos de respuesta óptimos y eficaces.

9. Recomendaciones

- Para futuros proyectos se recomienda el uso de cámaras infrarrojas con el objeto de poder controlar y automatizar procesos en un ambiente nocturno, el cual da la posibilidad para un control automático; ya sea de vigilancia o como domótica.
- Para el cálculo de capacidad de un aire acondicionado se recomienda usar la fórmula propuesta por el Ing. Néstor Quadri del libro “Aires acondicionados y calefacción” con el fin de evitar gastos innecesarios en la adquisición de un equipo de este tipo.
- Para futuros proyectos que sigan esta línea se recomienda usar Arduino por su facilidad de uso y excelente información disponible, además es una plataforma libre con múltiples librerías que permiten adaptarse rápidamente a estos tipos de proyectos.
- Para la detección de objetos usando OpenCV, se recomienda primero convertir la imagen a escala de grises, aplicar ecualización de histograma y por último redimensionar la imagen para conseguir una optimización en el procesamiento de la imagen para cuando OpenCV utilice el filtro de detección este no tenga mucha información que procesar.
- En la implementación de este proyecto se recomienda ubicar el sensor de temperatura LM35 al extremo opuesto de donde se encuentra el aire acondicionado evitando colocarlo cerca de ventanas o paredes a las cuales el sol llegue directamente, otra ubicación correcta es la parte superior del aire acondicionado impidiendo que éste quede expuesto a la corriente de aire que expele, se deberá considerar el ángulo de las aspas del aire acondicionado.
- Para un mejor rendimiento y optimización del aire acondicionado se recomienda mantener tanto puertas como ventanas cerradas con el fin de evitar la pérdida de temperatura en la sala.

10. Bibliografía

[1] "Concepto de temperatura", Abril 2015, *WordPress*, [online], Disponible en: <http://definicion.de/temperatura/#ixzz3YZ16xfoq>

[2] "Reglamento de seguridad y salud de los trabajadores y mejoramiento del medio ambiente de trabajo", 1998, [online], Disponible en: <http://www.utm.edu.ec/unidadriesgos/documentos/decreto2393.pdf>

[3] Zona variable de confort térmico, Capítulo 2 "Conceptos generales sobre Ambiente y confort térmico", 2006, [online], Disponible en: http://www.aenor.es/aenor/normas/normas/fichanorma.asp?tipo=N&codigo=N0037517#.VafdSvl_Oko

[4] "Definición de Aire Acondicionado", 2007-2015, Definición ABC, [online], Disponible en: <http://www.definicionabc.com/tecnologia/aire-acondicionado.php>

[5] N. Quadri, "Manual del aire acondicionado y calefacción", Editorial Alsina, Argentina: Buenos Aires, 2007 [online], Disponible en: <http://www.acondicionamiento.com.ar/nueva/wp-content/uploads/2009/02/libro-manual-aa-y-calefaccion.pdf>

[6] N. Quadri, "Manual del aire acondicionado y calefacción", Editorial Alsina, Argentina: Buenos Aires, 2007 [online], Disponible en: <http://www.acondicionamiento.com.ar/nueva/wp-content/uploads/2009/02/libro-manual-aa-y-calefaccion.pdf>

[7] N. Quadri, "Aire acondicionado y Climatización", [online], Disponible en: <http://www.elaireacondicionado.com/calculo.php>

[8] Massimo, David, Tom, Página Oficial, "Getting Started with Arduino", 2015, [online], Disponible en: <http://www.Arduino.cc/en/Guide/Introduction>

[9] Antonio Quirarte, "Microcontroladores", Junio 2014, Disponible en: <http://hacedores.com/que-tarjeta-de-desarrollo-elegir-parte-1/>

[10] Ken Shirriff, "Librería IrRemote", Enero 2010, [online], Disponible en:

<http://www.righto.com/2010/01/using-arbitrary-remotes-with-arduino.html>

- [11] C. H. Tapia Ayala, H.M. Manzano Yupa, Universidad Politécnica Salesiana, "Evaluación de la plataforma Arduino e implementación de un sistema de control de posición horizontal", Octubre 2013, pp 52-53.
- [12] E. Limor, C. O. Johnson, Massachusetts Institute of Technology "SENSORS", 2004 [online], Disponible: <http://www.ladyada.net/wiki/tutorials/learn/sensors/ir.html>
- [13] S. Rocha, "Control universal con arduino. Utilizando un emisor y receptor infrarrojos", Abril 2014, [online], Disponible en: <http://alumnos.informatica.utem.cl/~srocha/control-universal-con-arduino-utilizando-un-emisor-y-receptor-infrarrojos.html>
- [14] A. Castro Domínguez, Universidad Politécnica de Madrid, "Sensores de Temperatura", Enero de 2000, [online], Disponible en: http://oa.upm.es/22322/1/PFC_ALBERTO_CASTRO_DOMINGUEZ.pdf
- [15] Texas Instruments "Sensor de Temperatura LM35", (Precision Centigrade Temperature Sensors), Post Office Box 655303, Dallas, Enero 2015, [online], Disponible en: <http://www.ti.com/lit/ds/symlink/lm35.pdf>
- [16] H. Marchand Rivas, Artículo Tecnología, "Cámara web". Octubre 2010, [online], Disponible en: <http://www.slideshare.net/lalex20/camara-web>
- [17] A. Informática Moderna, "Cámaras Web", Enero 2015, [online], Disponible en: http://www.informaticamoderna.com/Camara_web.htm
- [18] Welcome to opencv documentation!, Febrero 2014, [online], Disponible en: <http://docs.opencv.org/>
- [19] Bradski, Gary. Learning OpenCV Computer Vision with the OpenCV Library, Ed. O' Reilly Media 2008, pps 578
- [20] VIOLA, Paul y JONES, Michael. Robust Real-Time Face Detection. International Journal of Computer Vision, Vol.57, No.2, pp 137-154, 2004. Junio de 2008.
- [21] Citation: zope. (2012, November 23). detection_of_objects. Retrieved June 22, 2015, from UNIA OpenCourseWare, [online], Disponible en línea:

http://ocw.unia.es/ciencias-tecnologicas/tecnologia-del-ocio/materiales-basicos-folder/html/B2_U3/deteccion_de_caras.html/skinless_view

[22] J. R. Valvert Gamboa, G. Sánchez Barrios, Universidad de San Carlos de Guatemala, “Métodos y técnicas de reconocimiento de Rostros en imágenes digitales bidimensionales”, Junio 2006.

[23] L. Chiariglione, P. Marenzoni, Telecom Italia SpA “*Java Agent Development Framework*”, 2015, [online], Disponible en: <http://jade.tilab.com>

[24] T. Araújo Figueiredo, T. Ucha Gándara, “Metodología Prometheus”, Enero 2011, pp 2-6.

11. Anexos

11.1 Glosario de Términos

Termino	Definición
Trifosfato de adenosina (ATP)	Es un nucleótido fundamental en la obtención de energía celular. Es la principal fuente de energía para la mayoría de las funciones celulares.
Metabólico	Es el conjunto de reacciones bioquímicas y procesos físico-químicos que ocurren en una célula y en el organismo. Estos complejos procesos interrelacionados son la base de la vida, y permiten las diversas actividades de las células: crecer, reproducirse, mantener sus estructuras, responder a estímulos etc.
Ritmo circadiano	Conocido también como ritmo biológico, son oscilaciones de las variables biológicas en intervalos regulares de tiempo. Todos los organismos muestran algún tipo de variación rítmica fisiológica (tasa metabólica, producción de calor, floración, etc.) que suele estar asociada con un cambio ambiental rítmico.
Hipotálamo	Parte del encéfalo situada en la zona central de la base del cerebro que controla el funcionamiento del sistema nervioso y la actividad de la hipófisis.
Enfermedad febril	Enfermedad que se produce por el aumento exagerado de temperatura.
Fragosos	Detalles complejos de programación complicados abruptos.

Basado en Wiring	Wiring es una plataforma abierta de prototipos electrónicos compuestos de un entorno de programación, una tarjeta de prototipo, documentación (creada por ingenieros y artistas y una comunidad de expertos), personas con experiencia y nivel intermedio comparten sus ideas, conocimiento y su experiencia colectiva.
Basado en Processing	Es un lenguaje de programación de código abierto, para las personas que quieran crear imágenes, animaciones e interacciones.
AC/DC	Corriente Alterna / Corriente Directa
Fotoeléctrica	Dícese de cualquier fenómeno eléctrico provocado por la intervención de radiaciones luminosas.
Clicker	El clicker es un simple aparato que emite un sonido “click” al momento de presionar un botón del control remoto.
Potencia de disipación	Capacidad de disipar toda la potencia que se producen sin calentar el sensor en exceso.

11.2 Código arduino decodificación señales infrarrojas

```
#include <IRremote.h>
```

```
int RECV_PIN = 11;
```

```
IrReceiver irrecv (RECV_PIN);
```

```
decode results results;
```

```
void setup()
```

```
{
```



```

Serial.begin(9600);

irrecv.enableIRIn(); // Start the receiver
}

void dump(decode results *results) {

int count = results->rawlen;

if (results->decode_type == UNKNOWN) {

    Serial.print("Unknown encoding: ");

}

else if (results->decode_type == NEC) {

    Serial.print("Decoded NEC: ");

}

Else if (results->decode type == SONY) {

    Serial.print("Decoded SONY: ");

}

else if (results->decode type == RC5) {

    Serial.print("Decoded RC5: ");

}

else if (results->decode type == RC6) {

    Serial.print("Decoded RC6: ");

}

else if (results->decode type == PANASONIC) {

```

```

Serial.print("Decoded PANASONIC - Address: ");

Serial.print(results->panasonicAddress, HEX);

Serial.print(" Value: ");

}

else if (results->decode type == JVC) {

    Serial.print("Decoded JVC: ");

}

Serial.print(results->value, HEX);

Serial.print(" (");

Serial.print(results->bits, DEC);

Serial.println(" bits)");

Serial.print("Raw (");

Serial.print(count, DEC);

Serial.print("): ");

for (int i = 0; i < count; i++) {

    if ((i % 2) == 1) {

        Serial.print(results->rawbuf[i]*USECPERTICK, DEC);

    }

    else {

        Serial.print(-(int)results->rawbuf[i]*USECPERTICK, DEC);

    }

}

```

```

    Serial.print(" ");
}

Serial.println("");
}

void loop() {

    if (irrecv.decode(&results)) {

        Serial.println(results.value, HEX);

        dump(&results);

        irrecv.resume();

    }

}

```

11.3 Código arduino lectura de temperatura y envío de señales infrarrojas

```

#include <IRremote.h>

IRsend irsend;

int val = 1;

void setup()

{

    Serial.begin(9600);

}

float temperatura = 0;

```

```

int promedio = 0;

float suma_temperaturas = 0;

void loop() {

    temperatura = (5.0 * analogRead(0) * 100.0) / 1023.0;

    suma_temperaturas = suma_temperaturas + temperatura;

    promedio = promedio + 1;

    if (promedio == 10) {

        Serial.println(suma_temperaturas / 10);

        suma_temperaturas = 0;

        promedio = 0;

        delay(500);

    }

}

void serialEvent()

{

    while (Serial.available())

    {

        val = Serial.parseInt();

        if (val == 0)

        {

            unsigned int rawON[200] = {X}; //X serán los códigos obtenidos por el decodificador

            irsend.sendRaw(rawON, 200, 38); } } }

```

11.4 Entrevista realizada a los hoteles de la ciudad de Loja

Estimados responsables:

El turismo en la ciudad de Loja actualmente es mayoritario debido a la cultura y amabilidad de su gente como también a la confortabilidad que se brinda en los diferentes hoteles de la ciudad específicamente en el Gran Hotel Loja, lo cual nos permite elogiar su capacidad de desarrollo. Sus opiniones serán importantes para valorar el trabajo y responsabilidad con la que se desempeñan en el mismo.

Le pedimos contestar las siguientes preguntas:

CÓDIGO	PREGUNTAS
FUNCIÓN: RESPONSABLES DE LAS SALAS DE CONFERENCIA ÁMBITO 2: ORGANIZACIÓN Y DIRECCIÓN	
P0001	¿Existen equipos que controlen la temperatura ambiente en sala de conferencias?
	Comentarios:
P0002	¿Cuántos aires acondicionados usan por cada sala de conferencias?
	Comentarios:
P0003	¿Cómo toman la decisión de usar más de un aire acondicionado en una sala de conferencia?
	Comentarios:

P0004	¿A qué temperatura inicial se fija el aire acondicionado?
	Comentarios:
P0005	¿En el transcurso de una conferencia la temperatura del aire acondicionado es modificada?
	Comentarios:
P0006	¿Cuándo existe aglomeración de personas se modifica la temperatura del aire acondicionado?
	Comentarios:
P0007	¿En qué momento encienden el aire acondicionado?
	Comentarios:

Loja, 13 de mayo de 2015

.....

Entrevistado (a)

.....

Responsable