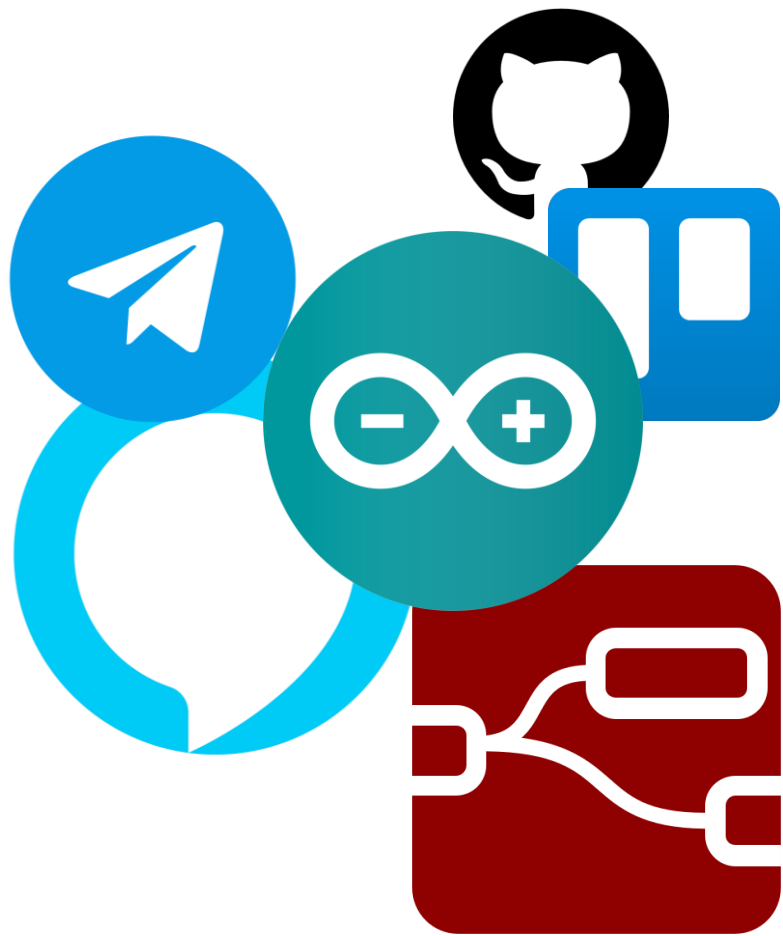


**20 DE ENERO DE 2022**



# **PROYECTO IoT**

**CURSO:**  
4º Grado en Ingeniería Electrónica, Robótica y Mecatrónica

**ASIGNATURA:**  
Informática Industrial

**AUTORES:**  
Luis Calderón Robustillo – 07274306G  
David Rodríguez Onieva – 51211340T  
Rubén González Navarro - 77493894V  
Pedro Antonio Peña Puente - 77228494V



## ÍNDICE

1.- INTRODUCCIÓN .....	1
1.1.- Descripción del Proyecto .....	1
1.2.- Elementos Utilizados .....	1
1.3.- Esquema General .....	2
2.- SOFTWARES DE ADMINISTRACIÓN.....	3
2.1.- TRELLO .....	3
2.2.- GIT-HUB .....	3
3.- ELEMETOS Y HERRAMIENTAS IMPLEMENTADOS .....	3
3.1.- Telegram .....	3
3.2.- FOTA.....	3
3.3.- Sensores .....	4
3.3.1.- Interruptor Táctil TTP223B .....	4
3.3.2.- Sensor de Movimiento HC-SR501 .....	4
3.3.3.- Sensor de temperatura y humedad DHT11 .....	5
3.3.4.- Fotorresistencia LDR .....	6
3.3.5.- Receptor de infrarrojo VS1838B .....	7
3.3.5.- Diodo LED IR.....	8
3.4.- Altavoz “Echo Dot de 3ra generación” .....	8
3.5.- Tiras Led RGB Multicolor con Control Remoto (WEILY).....	9
ANEXO I.....	10

## 1.- INTRODUCCIÓN

Se va a desarrollar un proyecto *IoT* (*Internet of Things*) que a rasgos generales será un proyecto de asistencia al usuario en iluminación, temperatura y audio en el hogar.

### 1.1.- Descripción del Proyecto

La ESP-8266 es un módulo para Arduino de bajo costo que permite la conexión Wi-Fi de la placa en cuestión con la red. Con esta, conseguimos que todos los proyectos que hasta ahora requerían conexión física pudiesen trabajar a distancia, controlar desde una puerta de garaje, hasta una alarma casera o construir un controlador LED. En nuestro caso, nos adaptaremos a la comodidad del usuario y el ahorro energético, específicamente en el hogar. El proyecto ofrecerá un servicio de variación de luminosidad de la estancia en función de la cantidad de luz incidente por la ventana, otro de estabilización de la temperatura en una referencia agradable y un servicio de sugerencias de música. El sistema incluirá el uso de la aplicación de *Telegram* como interfaz de interacción con el usuario. En ella el usuario podrá elegir entre utilizar las escenas de música, temperatura y luminosidad sugerida o utilizar una personalizada. Las canciones se reproducirán mediante el asistente virtual *Alexa*, para la temperatura se usará un aire acondicionado de la marca MITSUBISHI y para la luminosidad una cinta de LEDs con sensor infrarrojo.

El algoritmo que determinará las sugerencias ofrecidas al usuario se basará en gran medida en la meteorología del exterior. En especial con la medida de la luminosidad exterior que llega a la habitación y la temperatura que hace en la propia habitación. Adicionalmente, el proyecto contará con un sistema de activación mediante sensor táctil, y un sistema de renovación de la sesión mediante un sensor de movimiento. Es decir, el sistema permanecerá apagado hasta que el usuario interaccione con el sensor táctil. El sistema, una vez pasado un tiempo, se apagará suponiendo que nadie lo está usando (sensor de presencia no detecta nada). No obstante, una vez activado no será estrictamente necesario renovar la sesión mediante el sensor táctil, sino que el sistema contará con un sensor de movimiento que detectará el movimiento del usuario por la habitación. De forma adicional, el usuario podrá activar o desactivar manualmente el sistema mediante *Telegram*. Por ejemplo, activamos y utilizamos el sistema, y posteriormente queremos apagar el sistema sin tener que esperar el tiempo en el que se apaga automáticamente por no detectar presencia. En este caso acudiríamos a *Telegram* y finalizaríamos la sesión manualmente.

### 1.2.- Elementos Utilizados

Como elementos utilizados para el desarrollo del proyecto, se han usado cuatro módulos ESP-8266, sensores de temperatura y humedad DH11, sensores de movimiento HC-SR501, sensores de luminosidad GL55, resistencias de 10K $\Omega$  y de 100 $\Omega$ , un interruptor táctil TTP223B, un aire acondicionado de la compañía Mitsubishi, un diodo LED IR, un receptor infrarrojo VS1838B, un altavoz Echo Dot de 3ª Generación, el asistente virtual “*Alexa*” y una cinta de iluminación LED.

En cuanto al software utilizado para el interconexionado de todos los elementos, se ha hecho uso de *Node-Red*, *Arduino IDE*, *Telegram*, MQTT Broker Mosquitto, ESP-NOW, Mongo-DB y *Amazon Alexa*.

Se hará un pequeño estudio de todos estos elementos más adelante.

### 1.3.- Esquema General

Como podemos ver en la *Ilustración 1*, disponemos de 3 placas que actuarán de master y una de esclavo. Cada placa master le envía datos a la placa esclava por ESP-Now, dos mandan datos meteorológicos y la otra manda el estado del sistema.

La placa esclava manda por MQTT mediante 2 topics estos datos y nos suscribimos a ellos en *Node-Red*. También manda mediante un sensor infrarrojo una señal que controla el aire acondicionado en función de la temperatura.

A fin de poder realizar una mejor recomendación de escena (música a escuchar + color de las luces) se ha optado por hacer uso de una base de datos para ir almacenando la lectura de los sensores (Topic Meteorología) y cuando se vaya a hacer una recomendación al usuario el resultado proporcionado sea más acertado.

Como se ha comentado en el apartado anterior, la activación del sistema se realizará mediante un sensor de presencia y/o un sensor táctil. También está la opción de encendido mediante el bot de *Telegram*. Una vez que tengamos iniciado el sistema en función de los datos meteorológicos más recientes, se recomendará por *Telegram* una *playlist* y una iluminación. El usuario será el encargado de responder si desea esta iluminación y música o ser él mismo el que elija otras mediante comandos mandados a *Telegram*.

Una vez que el usuario ha respondido y tenemos esa información y los demás datos, a través de *Node-Red* se enviarán ciertos comandos a *Alexa* que reproducirá la música en el *Echo Dot* y pondrá las luces del color e intensidad correspondiente.

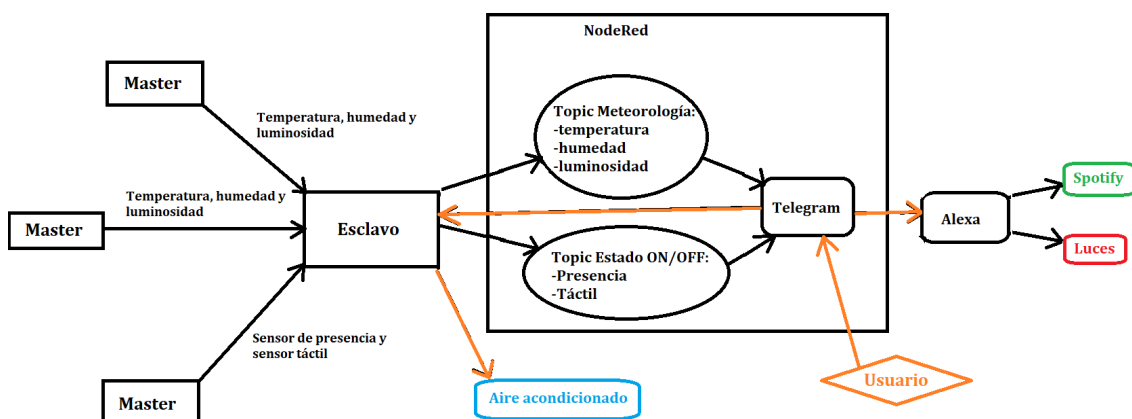


Ilustración 1. Esquema General del proyecto

## 2.- SOFTWARES DE ADMINISTRACIÓN

### 2.1.- TRELLO

Hemos utilizado esta herramienta para crear espacio de trabajo para organizar la realización del proyecto. Empezamos desarrollando en el espacio los puntos que pensábamos convenientes abordar en el proyecto y los clasificamos en distintos labels tales como Investigación, Hardware, Documentación, Programación Arduino y Flujo *Node-Red*.

A medida que íbamos completando tareas íbamos añadiéndolas al bloque de “Done” o al bloque de “Testing” si las habíamos terminado, pero no probado (en caso de ser programación). Este método nos ha permitido de forma eficiente hacer un seguimiento del trabajo realizado y sobre todo tener un control y una visión constante del estado de nuestro trabajo. Ha ayudado a recordar y dividir las tareas pendientes. Además, hemos tenido en todo momento dividido el proyecto en los distintos labels y cada tarea con su pequeña descripción. Así se ha perdido mucho tiempo a la hora de volver al trabajo después de varios días de desconexión.

### 2.2.- GIT-HUB

Este software online, muy utilizado por toda la comunidad, lo hemos utilizado con el fin de organizar nuestro proyecto y tener de manera online archivos en común al que tengamos acceso todos los miembros de grupo y los profesores. En un primer momento creamos un main principal como punto de partida sobre el que trabajar. Posteriormente se han ido abriendo “Branches” con el contenido de todas las pequeñas partes del trabajo que se han ido realizando y probando por separado. Estas solo contienen material que luego ha sido unido al proyecto principal y por lo tanto se han quedado como “Branches” auxiliares. Una vez realizado todo el trabajo por separado se ha unido para realizar los programas de cada una de las placas, los flujos de *Node-Red* y los documentos del proyecto final.

## 3.- ELEMETOS Y HERRAMIENTAS IMPLEMENTADOS

### 3.1.- Telegram

Se ha creado un bot llamado “*PrealexaInfind\_bot*” con el cuál interactuaremos para realizar una serie de acciones determinadas por el comando que le mandemos. Los comandos son: */alegre*, */bailar*, */blanco\_solo*, */chill*, */concentración*, */continua*, */inicio*, */start* y */triste*.

Como hemos explicado previamente, tendremos un comando para seguir la recomendación acorde con el tiempo, un comando para el inicio manual sin tener que estar en la habitación y otros comandos para elegir tu propio estilo de música e iluminación.

### 3.2.- FOTA

Al igual que con *Telegram*, reutilizamos tanto el código como el flujo de *Node-Red* hechos previamente en la asignatura. Implementar esto nos permite actualizar el código directamente desde la nube sin tener que conectar las placas al ordenador.

Esto es muy útil ya que si pusiéramos las placas en la pared atornilladas por ejemplo no tendríamos que estar desatornillando y atornillando cada vez que queramos modificar el código ya sea para arreglar bugs o para modificar la sensibilidad de los sensores.

De esta manera cada vez que queramos modificar el código de alguna de las placas podemos crear un archivo binario y subirlo a la plataforma. Una vez hecho esto, resetearemos la

placa en cuestión y se actualizará automáticamente. Esto es aplicable a las cuatro placas utilizadas en el proyecto

### 3.3.- Sensores

En este proyecto hemos usado múltiples sensores:

#### 3.3.1.- Interruptor Táctil TTP223B

Un sensor táctil capacitivo es un dispositivo que presenta un comportamiento similar a un pulsador, pero puede ser activado con poca o ninguna presión.

Por default la salida del módulo se encuentra en 0 lógico y bajo consumo de corriente, cuando nuestro dedo u otro componente “toca” la posición correspondiente a la salida del módulo se activa el 1 lógico, notando este cambio lógico a través del led. Si no se toca el módulo en 13 segundos, este vuelve a modo bajo de energía.

La variación de la capacitancia es la clave en el funcionamiento. La placa censora y el cuerpo humano actúan como condensador y, por tanto, forman un sistema que almacena una carga electricidad.

Al reducir la distancia la capacitancia aumenta y el sistema almacena una carga superior. Esta acumulación de carga puede ser detectada en la placa censora y generar una señal digital cuando supere un cierto valor. Esta señal de disparo puede a ser vez ser capturada con una entrada digital de Arduino.

En este proyecto, el uso de este sensor se ha destinado a la activación del sistema. Cuando el usuario active el sensor, se enviará una señal que habilite todo el sistema y a su vez se comenzará una cuenta atrás para la posterior desactivación del sistema.



*Ilustración 2. Interruptor Táctil TTP223B*

Se conectará la patilla IO al pin 5 (D1) y asimismo las patillas VCC y GND a los pines 3.3V y GND respectivamente.

#### 3.3.2.- Sensor de Movimiento HC-SR501

Los sensores de movimiento son sensores infrarrojos pasivos (PIR). Debido al bajo costo, consumo y fácil uso se ha optado para usarlo en nuestra aplicación domótica.

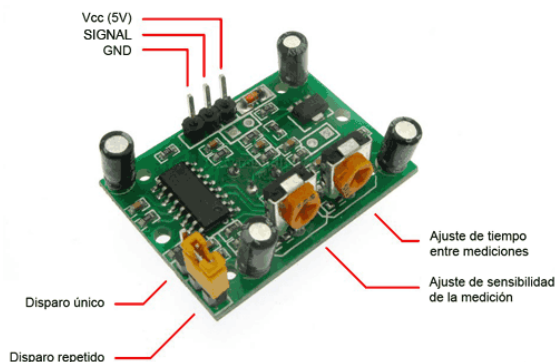
Este sensor detecta el movimiento de personas hasta 6 metros de distancia mediante el uso de una lente de Fresnel y el elemento sensible al infrarrojo para detectar cambios en los patrones infrarrojos emitidos por los objetos en su cercanía.

En nuestro caso, se ha usado el sensor para reestablecer la cuenta atrás de la desactivación del sistema cada vez que detecte movimiento. Esto se hace por un lado para hacer un uso eficiente de la energía y para la comodidad del usuario ya que no tiene que acercarse a activar el sensor táctil cada cierto tiempo evitando interrumpir sus actividades.



*Ilustración 3. Sensor de Movimiento HC-SR501*

En cuanto al conexionado de este sensor nos hemos basado en el siguiente esquema:



*Ilustración 4. Sensor de temperatura y humedad DHT11*

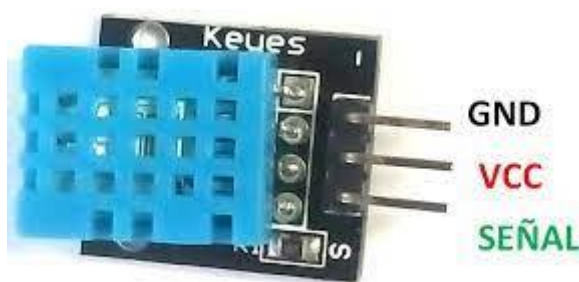
Se conectará la patilla SIGNAL al pin 14 (D5) y asimismo las patillas VCC y GND a los pines 3.3V y GND respectivamente.

### 3.3.3.- Sensor de temperatura y humedad DHT11

Hemos reutilizado este sensor con el cual hemos estado trabajando durante todo el curso. Integra un sensor capacitivo de humedad y un termistor para medir el aire circundante, y muestra los datos de temperatura y humedad mediante una señal digital en el pin de datos.

Este sensor es una parte fundamental de nuestro proyecto ya que junto al sensor de luminosidad nos permite ofrecer una recomendación acorde con el tiempo de un estado de ánimo para posteriormente poner la música e iluminación adecuadas.





*Ilustración 5. Sensor de temperatura y humedad DHT11*

Se conectará la patilla SEÑAL al pin A0 y asimismo las patillas VCC y GND a los pines 3.3V y GND respectivamente.

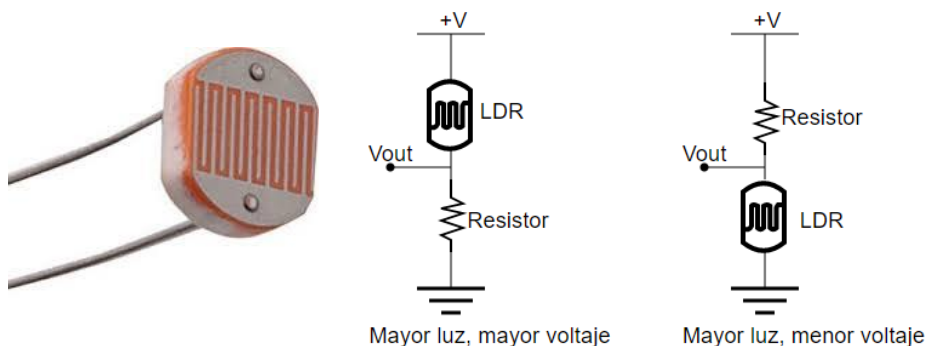
### 3.3.4.- Fotorresistencia LDR

Hemos usado esta para detectar cambios de luz ya que, cuanto más luz entre menos luz te artificial tendremos que consumir (los leds tendrán menos intensidad).

Su funcionamiento es básico, cuando el LDR no está expuesto a radiaciones luminosas, los electrones están firmemente unidos en los átomos que lo conforman, pero cuando sobre él inciden radiaciones luminosas, esta energía libera electrones con lo cual el material se hace más conductor, y de esta manera disminuye su resistencia. Las resistencias LDR solamente reducen su resistencia con una radiación luminosa situada dentro de una determinada banda de longitudes de onda.

Existen dos formas básicas para la conexión de nuestra LDR, pueden ser utilizadas dependiendo del fin deseado. Si disponemos de un controlador es posible modificar los resultados mediante programación.

- Mayor luz, mayor voltaje: Al conectar la fotorresistencia al nodo positivo de nuestra fuente de voltaje tendremos que, al incidir una mayor cantidad de luz provocará una menor caída de voltaje o diferencial de potencial entre la fuente y el pin de referencia ( $V_{out}$ ), por lo tanto, se tendrá una lectura mayor.
- Mayor luz, menor voltaje: En pocas palabras la fotorresistencia se conecta al nodo de GND y provocará un comportamiento opuesto al punto 1.



*Ilustración 6. Fotorresistencia LDR*



Cuyo montaje quedaría del siguiente modo:

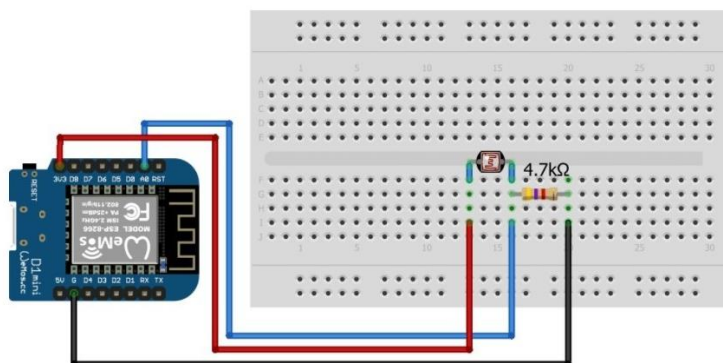


Ilustración 7. Montaje fotorresistencia LDR

### 3.3.5.- Receptor de infrarrojo VS1838B

Como su nombre lo indica, es un receptor de infrarrojos utilizado principalmente en los controles remotos. Utiliza la codificación NEC.

Hemos utilizado este receptor para recoger la señal que envía un mando de aire acondicionado cuando se pulsa el botón de encender, apagar, aumentar la temperatura y disminuir la temperatura. Una vez recogida esta señal la hemos mostrado por pantalla y anotado el valor de este dato para poder utilizarla posteriormente.

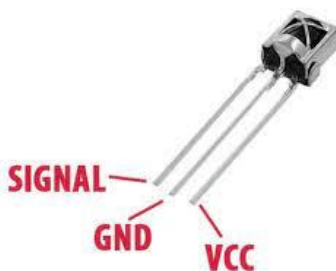


Ilustración 8. Receptor Infrarrojo VS1838B

Y su montaje quedaría del siguiente modo:

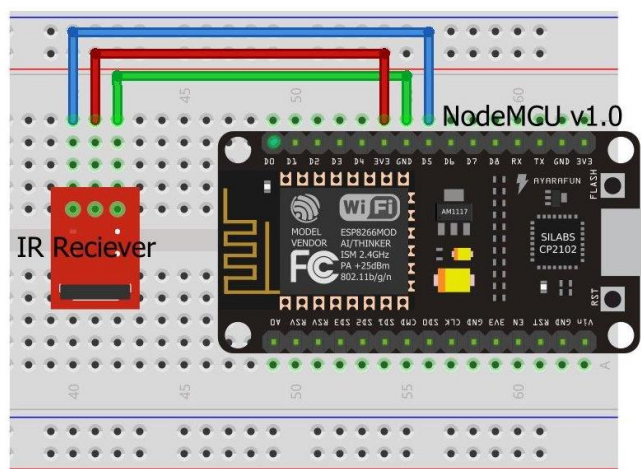


Ilustración 9. Montaje receptor IR.

### 3.3.5.- Diodo LED IR

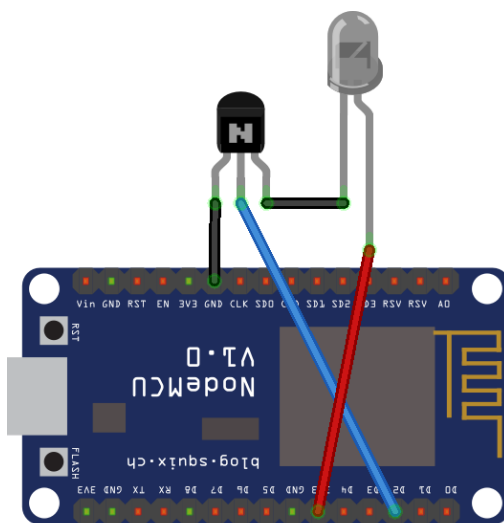
Un diodo emisor de luz infrarroja (LED) es un tipo de dispositivo electrónico que emite luz infrarroja no visible a simple vista. Un LED de infrarrojos (IR) funciona como un LED normal, pero puede utilizar diferentes materiales para producir luz infrarroja. Esta luz infrarroja puede usarse para un control remoto, para transferir datos entre dispositivos, para proporcionar iluminación para equipos de visión nocturna o para una variedad de otros propósitos.

En nuestro caso, lo hemos utilizado para mandar a un aire acondicionado la señal que hemos recogido previamente del mando. Esta señal es del valor al pulsar el botón de *On*, al pulsar *Off* o al cambiar la temperatura. Estos valores serán enviados según la temperatura que haga en la habitación (recogida por el sensor de temperatura) pudiendo así automatizar el aire y provocar una temperatura agradable en la habitación



*Ilustración 10. Diodo LED IR*

Cuyo montaje quedaría del siguiente modo:



*Ilustración 51. Montaje Diodo LED IR*

### 3.4.- Altavoz “Echo Dot de 3ra generación”

Este dispositivo dispone del asistente virtual “Alexa”. Para poder mandarle comandos de acción hemos tenido que instalar un ejecutable de *GitHub* en la máquina virtual. Con este (tras iniciar sesión rellenando los datos) podemos simular mediante un comando escrito lo que sería un comando de voz. Gracias a esto hemos automatizado las escenas (poner los leds de un color específico y la música de un estilo concreto) según el estado pedido por el usuario mediante *Telegram*.

La música se escucha a través de este a un volumen que hemos estimado normal. Si por algún casual el usuario quisiera subir el volumen puede hacerlo mediante un comando de voz ya que este dispositivo tiene un reconocimiento de voz. El comando para modificar el volumen sería “Alexa, sube el volumen” o en su defecto “Alexa, baja el volumen”. No obstante, no es necesario modificar puesto que el volumen que hemos puesto de base es adecuado.

Al principio se había implementado *Spotify* como servicio de música utilizado, no obstante, empezó a dar ciertos problemas así que terminamos sustituyéndolo por el servicio de Amazon Music.



Ilustración 62. Altavoz “Echo Dot de 3ra generación”

### 3.5.- Tiras Led RGB Multicolor con Control Remoto (WEILY)

Es una tira con LEDs los cuales se pueden encender con distintos colores. Tiene hasta 16 opciones de colores, aunque nosotros trabajaremos solo con 5. Su principal tarea es la de iluminar la habitación en función de la cantidad de luz que esté entrando por la ventana. Si hay mucha luz los leds deberán estar apagados o con muy poca intensidad logrando así un ahorro energético. En caso contrario, si no hay luz porque es de noche los leds estarán encendidos a máxima intensidad obteniendo una clara visibilidad de la habitación.



Ilustración 13. Tiras Led RG



# ANEXO I

## *Guía de Usuario*

## GUÍA DE USUARIO

Proyecto de asistencia en iluminación, temperatura y música para un usuario en el hogar.

### Índice de contenidos:

- Configuración Hardware
- Configuración Software
- Instrucciones de uso
- Personalización y Recursos

#### ○ Configuración Hardware

- En esta sección se indicarán los requisitos imprescindibles a configurar de forma previa en los equipos Hardware del sistema para que este funcione correctamente. Estos se realizarán solo la primera vez que se instale el sistema:

- Conexiones de sensores y alimentaciones de la placa (foto de sensor de presencia).
- Conectar Alexa

#### ○ Configuración Software

- En esta sección se indicarán los requisitos que no pueden faltar a la hora de configurar del sistema, estas configuraciones solo habrá que realizarlas la primera vez que lo queramos usar.

- Activar la máquina virtual y la carpeta de OTA en la máquina virtual.
- Configuración de conexión Wi-Fi en cada una de las placas (tanto SSID como password) y subir el nuevo archivo en FOTA.

- Conexión del bot de *Telegram*: en cada nodo de *Telegram* que usamos en *Node-Red* introducimos el nombre de nuestro bot "*PrealexaInfind\_bot*", y para terminar de configurarlo introducimos también el token correspondiente a este bot:

"5121583786:AAEQTfxxUHKg-U4mgAbPM-ZAC-D-uZhyNYM"

- Declaración de *MACs* de cada una de las placas a utilizar en el proyecto que se van a utilizar.

- Configurar *Alexa* en *Node-Red*: tras haber configurado el dispositivo *Alexa* con *Node-Red* (obtención del token y configuración con la cuenta de *Amazon Alexa*) usamos unos nodos llamados '*exed*' para usar el comando que queramos que interprete *Alexa*, pero en vez de introducirlo por voz, lo introducimos por una señal, en este caso por *Telegram*.

#### ○ Instrucciones de uso

- Encender máquina virtual y abrir *Node-Red*.
- Abrir *Dashboard* en caso de querer ver los parámetros del sistema.
- Abrir en *Telegram* conversación con el bot "*Prealexa*". Para empezar la conversación su puede usar tanto el comando */start* como decir cualquier cosa. El bot te responderá saludándote y diciéndote la configuración recomendada según la climatología. Una vez hecha esta recomendación el usuario puede elegir si seguirla o coger otra más acorde con sus necesidades. Para elegir la recomendación el usuario tiene que responder al bot con el comando */continua*. Para elegir cualquier otra configuración el usuario puede elegir entre los siguientes comandos */triste*, */alegre*, */chill*, */bailar* (la configuración recomendada será una de estas 4). Estas configuraciones establecen que suene un tipo de música y que se enciendan leds de un color u otro para iluminar. Todas las configuraciones llevan incluidas un control del aire acondicionado para que el usuario esté lo más cómodo posible.

○ **Personalización y Recursos**

- Podemos cambiar las *playlist* que se recomiendan por defecto se recomiendan con los comandos */triste*, */alegre*, */chill*, */bailar* mediante la aplicación *Amazon Alexa* cambiando las rutinas.
- Podemos personalizar las luces de inicio del programa de igual forma que en el paso anterior.
- Podemos modificar los tiempos de espera hasta que el sistema se apaga tras no detectar presencia en la habitación