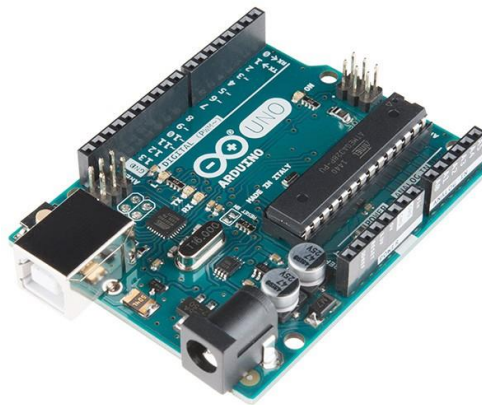




**Universidad Nacional Autónoma de México**  
**Facultad de Ingeniería**

**Detector de ritmo con ARDUINO**



**Alumna:** Ruiz Flores Laura Andrea

**Asignatura:** Fundamentos de programación

**Profesor:** Marco Antonio Martínez Quintana

**Semestre:** 2021 – 1

**Fecha de elaboración:** Enero 2021

## Resumen

En este documento se encuentra el desarrollo de un proyecto con ARDUINO, el manejo del programa y materiales, así como costos, disponibilidad de tiempo, etc. También algunos comentarios para complementar la explicación del procedimiento o en modo de comentario.

En la introducción se refiere a la motivación detrás de la realización de este proyecto y cómo la situación actual ha influido en esto. Se menciona también cómo el usuario puede interactuar y modificarlo a su conveniencia, para hacerlo más didáctico. También se menciona un enfoque empresarial que puede tener este proyecto, así como su importancia hoy en día para el usuario.

Para el desarrollo de este proyecto se cuenta con el algoritmo y el diagrama de flujo, los cuales indican los pasos para realizar el circuito y el programa a ejecutar. El código fuente necesario para hacer que el circuito realice la acción que requerimos, en este caso que las luces se enciendan al ritmo de una canción o sonido. También hay una imagen que muestra cómo es el circuito y la distribución que puede tener, así como notas respecto a la ejecución. Están el enlace que dirige a un video en el cual hay una explicación rápida de cómo funciona el programa y el circuito juntos, y un enlace que dirige a un repositorio en el que se puede acceder al código y al circuito de manera más clara.

Incluye la comparación de costos respecto a lo contemplado al principio del semestre y cuánto fue en realidad, los materiales con los que se pensaba se contarían y los que se requirieron al final. También un diagrama de Gantt para indicar cómo fue el flujo de la realización del proyecto.

Al final cuenta con una guía para el usuario que desee realizar este proyecto, que contiene los materiales a utilizar, los pasos y enlaces donde puede adquirir el código y la imagen del circuito a montar.

En las conclusiones se menciona cómo el proyecto y el entorno donde está desarrollado pueden hacer un impacto en nuestra sociedad y cómo influye en aspectos relacionados con iniciativas tecnológicas.

También se incluye la experiencia al realizar un proyecto de este tipo, que tanto el software como el hardware son esenciales para su funcionamiento, y las complicaciones que hubo a lo largo del proceso.

## Introducción

Hoy en día el uso de tecnologías para la mayoría de la población es primordial, para cada acción que realicen hay un dispositivo con el que pueden realizarla. Esto lleva a que también haya gran cantidad de productos que son sólo por entretenimiento o para decoración, y en esto está basado este proyecto. Se parte del principio de aquellos dispositivos que para funcionar necesitan un estímulo auditivo, como las luces que se encienden con una palmada.

La importancia que tiene nuestro entorno hoy en día ha escalado de gran manera, derivado de que tenemos que pasar mucho tiempo en él por la situación de la pandemia que nos rodea, el hacer que nuestro espacio de trabajo sea menos monótono es algo que nos motiva a que lo que realicemos sea de mayor calidad, por sentirnos en un lugar cómodo.

Últimamente todo lo que tiene que ver con luces se ha popularizado, es algo muy llamativo para gran cantidad de gente, y qué más que estas luces vayan al ritmo de alguna canción o sonido que reproduzcas, esto hace que la persona se sienta envuelta en lo visual y lo auditivo.

La idea detrás de esto es aportar al usuario algo que lo acompañe en su espacio personal, que no ocupe demasiado espacio y pueda tener a la mano en caso de requerirlo.

Se decidió que se desarrollaría en Arduino porque es un software que cualquier usuario con una computadora puede acceder, el principal problema es que requiere de una inversión monetaria para obtener el hardware requerido para ejecutar el programa o para que se observe de mejor manera el resultado. Una ventaja es que este mismo hardware puede ser utilizado para realizar muchos más proyectos, por lo que no es necesario gastar dinero siempre que se quiera desarrollar un proyecto en este software.

También la intención es que la persona que quiera realizar el proyecto tenga acceso al código de forma fácil y en caso de que quiera tomarlo como base para un proyecto más grande pueda hacerlo. Con esto va que se indican los materiales específicos para realizarlo, para que el usuario que tenga la posibilidad de adquirirlos sepa qué buscar y la cantidad que requiere.

El proyecto es personalizable, es decir, se puede cambiar algunos valores para que el patrón de encendido cambie o reemplazar las luces LED por otros colores que el usuario desee, esto para mejorar la experiencia en el uso del dispositivo.

Es un buen starter para aquel que quiera involucrarse en proyectos con Arduino, ya que incluye la interacción entre los componentes y las acciones del usuario, el que pueda probar el dispositivo de diferentes maneras y experimentar con el código para poder realizar proyectos más complejos o entender mejor cómo funciona cada una de las líneas de código.

## Enfoque de emprendimiento

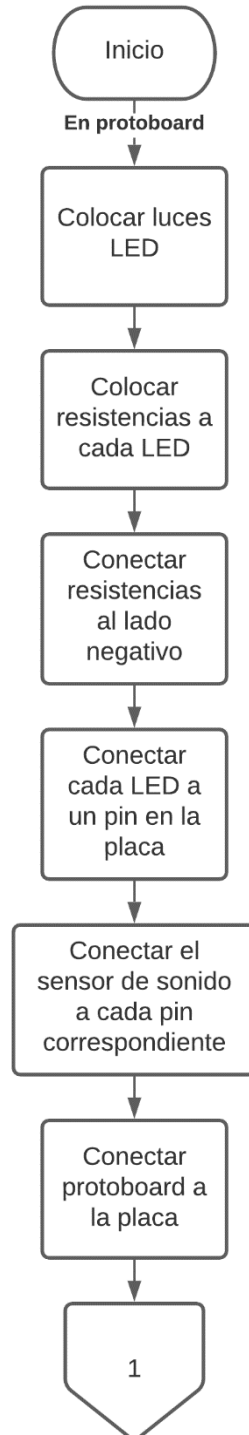
Este programa, como una idea con enfoque de emprendimiento, se basa mucho en cómo la gente ha querido personalizar sus espacios de trabajo, se podría vender sólo como un dispositivo independiente o venderse la idea a una marca de dispositivos electrónicos para que utilicen su marca en el producto. Puede venderse en cualquier comercio que venda productos electrónicos o cadenas de supermercados. En cuanto el programa aplicaría venderlo a una empresa en concreto para que diseñen el circuito que funcione con él de manera adecuada y precisa.

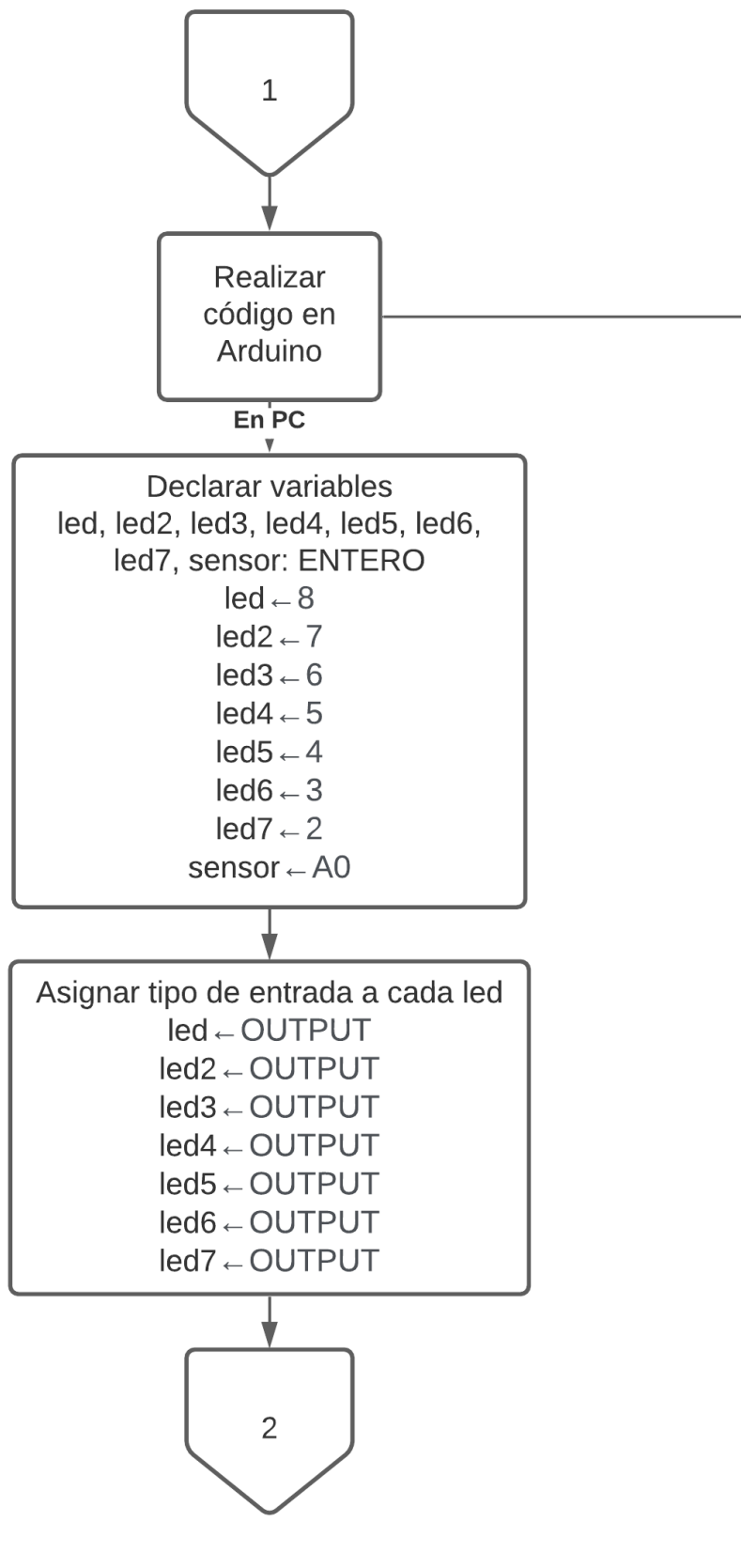
## Algoritmo

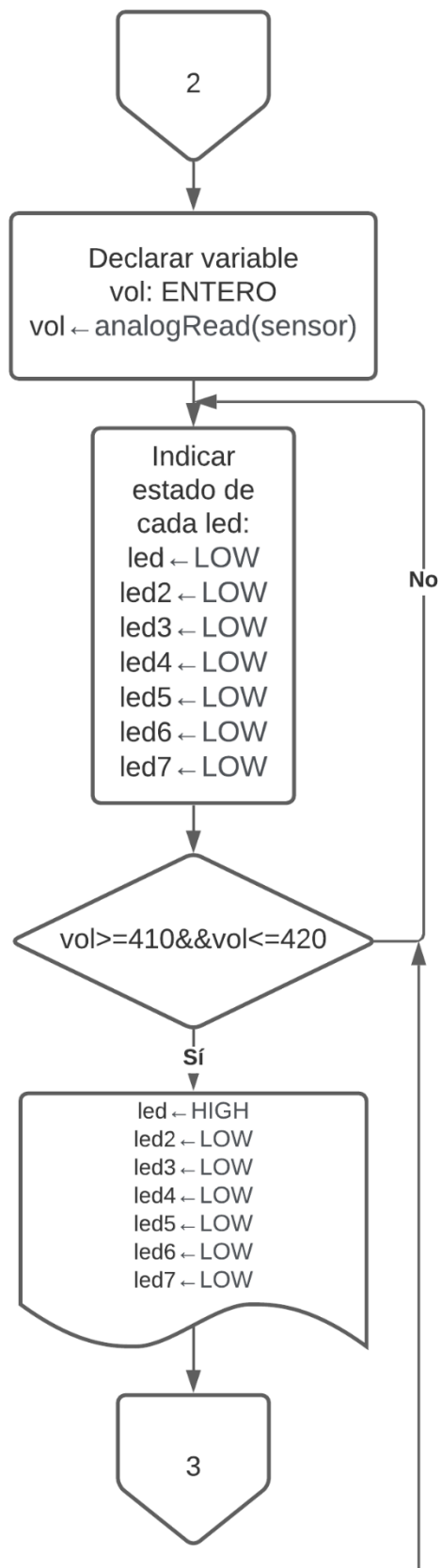
1. Colocar las luces LED en la protoboard.
2. Poner las resistencias para cada LED.
3. Poner un cable que conecte cada resistencia al lado negativo de la protoboard.
4. Conectar cada LED a un pin en la placa.
5. Tomar el sensor de sonido y conectar cada pin en su lugar correspondiente.
6. Colocar los cables que conecten a la protoboard a la placa.
7. Realizar el código en Arduino.
  - 7.1. Declarar variables para cada LED y una para el volumen que va a leer el sensor de sonido.
  - 7.2. Indicar que cada variable que corresponde a un LED es de tipo salida.
  - 7.3. Declarar una variable que tome los valores que la variable del volumen lea.
  - 7.4. Iniciar cada LED en apagado.
  - 7.5. Colocar estructuras de tipo condicional para que a cada LED se le asigne un rango del volumen.
    - 7.5.1. Si el volumen está dentro del rango uno, el primer led se encenderá.
    - 7.5.2. Si el volumen está dentro del rango dos, el segundo led y su antecesor se encenderá.
    - 7.5.3. Si el volumen está dentro del rango tres, el tercer led y sus antecesores se encenderán.
    - 7.5.4. Si el volumen está dentro del rango cuatro, el cuarto led y sus antecesores se encenderán.
    - 7.5.5. Si el volumen está dentro del rango cinco, el quinto led y sus antecesores se encenderán.
    - 7.5.6. Si el volumen está dentro del rango seis, el sexto led y sus antecesores se encenderán.
    - 7.5.7. Si el volumen está dentro del rango siete, el séptimo led y sus antecesores se encenderán.
8. Compilar el programa.
9. Conectar placa a entrada USB.
10. Subir el programa a la placa.
11. Reproducir canción en un altavoz.
12. Acercar el altavoz al sensor de sonido.

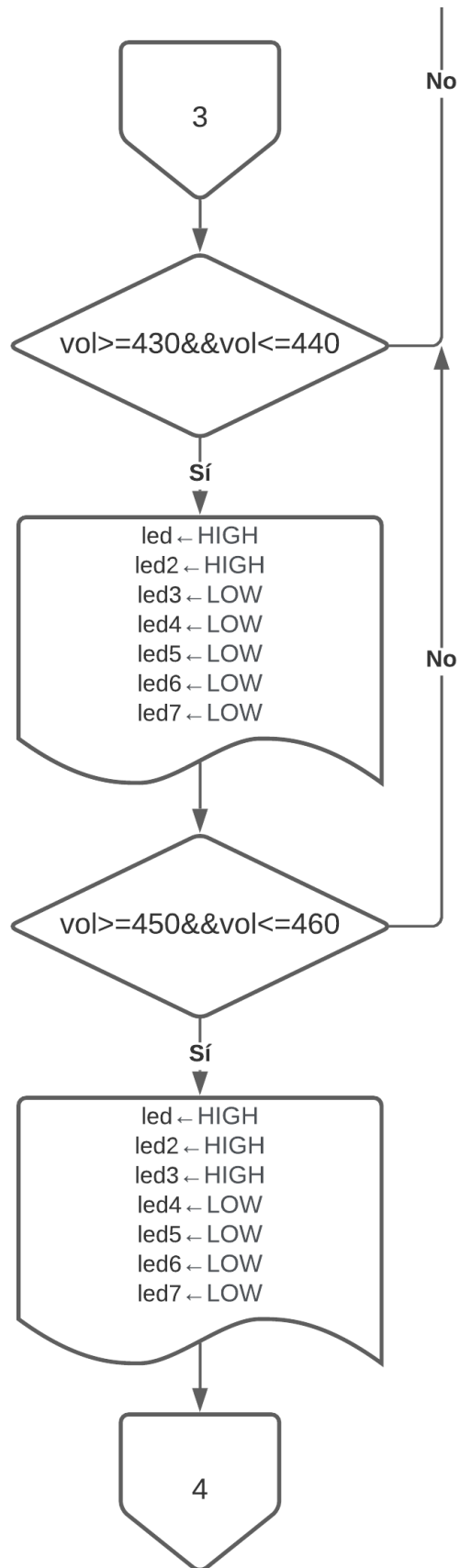
13. Observar cómo se enciende y se apagan las luces LED.

### Diagrama de flujo

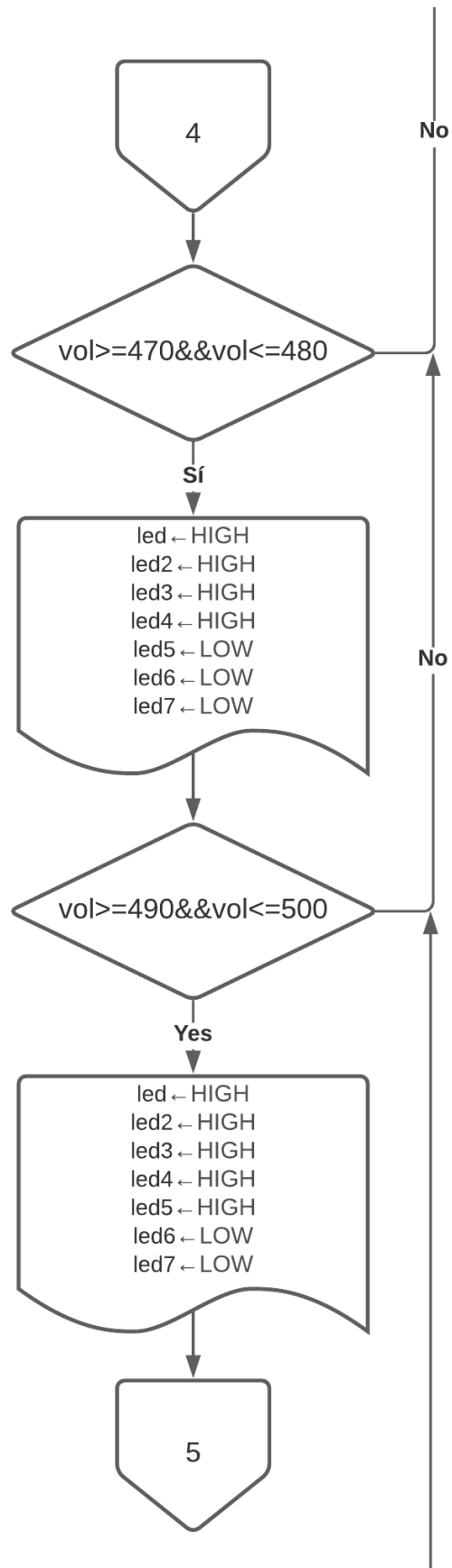


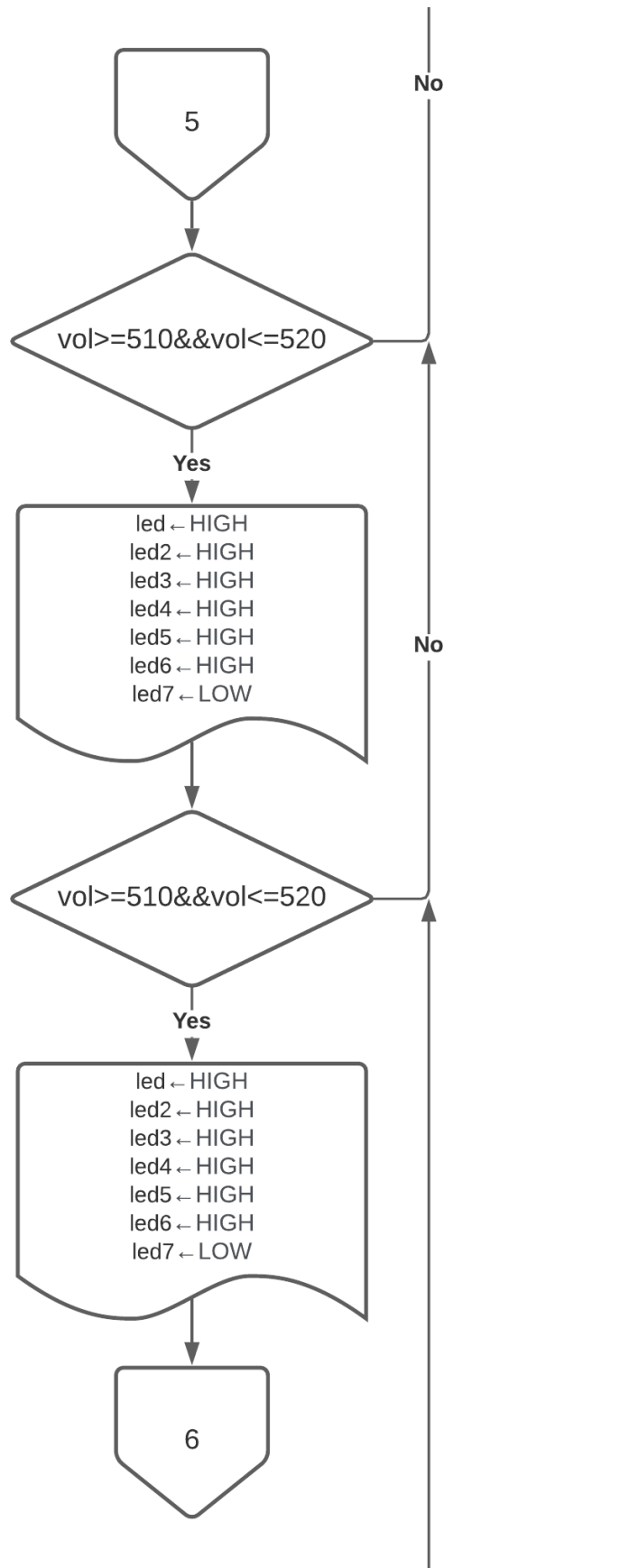


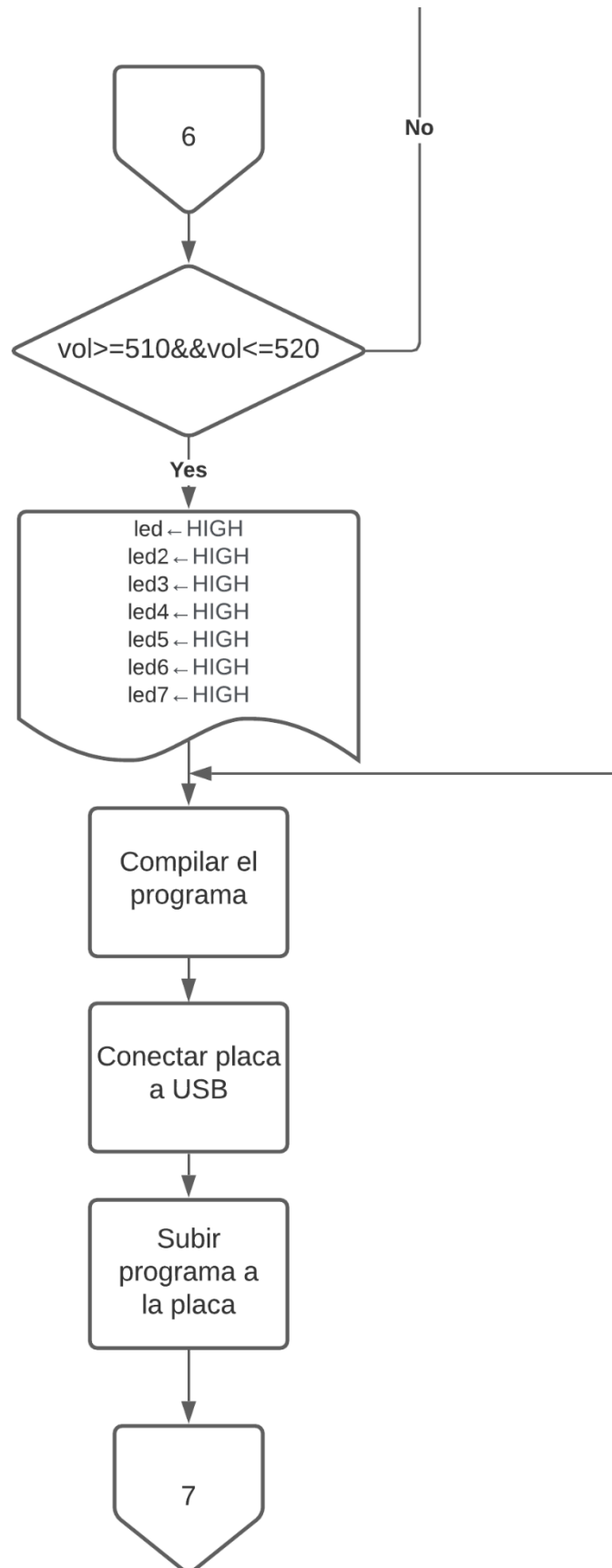


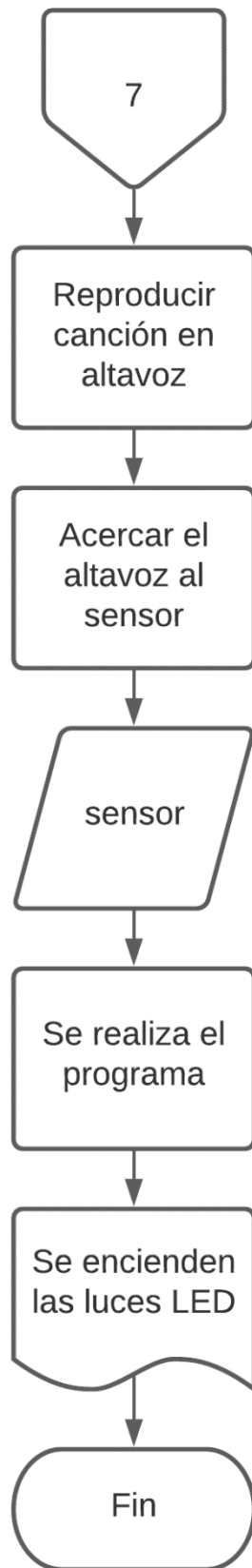












## Pseudocódigo

Nota: Se referirá al pseudocódigo del código escrito en Arduino.

### INICIO

led, led2, led3, led4, led5, led6, led7, sensor, vol: ENTERO

led = 8

led2 = 7

led3 = 6

led4 = 5

led5 = 4

led6 = 3

led7 = 2

sensor = A0

LEER sensor

Vol = sensor

SI vol $\geq$ 410 y vol $\leq$ 420

ESCRIBIR led

SI vol $\geq$ 430 y vol $\leq$ 440

ESCRIBIR led

ESCRIBIR led2

SI vol $\geq$ 450 y vol $\leq$ 460

ESCRIBIR led

ESCRIBIR led2

ESCRIBIR led3

SI vol $\geq$ 470 y vol $\leq$ 480

ESCRIBIR led

ESCRIBIR led2

ESCRIBIR led3

ESCRIBIR led4

SI  $\text{vol} \geq 490$  y  $\text{vol} \leq 500$

ESCRIBIR led

ESCRIBIR led2

ESCRIBIR led3

ESCRIBIR led4

ESCRIBIR led5

SI  $\text{vol} \geq 510$  y  $\text{vol} \leq 520$

ESCRIBIR led

ESCRIBIR led2

ESCRIBIR led3

ESCRIBIR led4

ESCRIBIR led5

ESCRIBIR led6

SI  $\text{vol} \geq 530$

ESCRIBIR led

ESCRIBIR led2

ESCRIBIR led3

ESCRIBIR led4

ESCRIBIR led5

ESCRIBIR led6

ESCRIBIR led7

FIN

## Código fuente

```
1 int sensor = A0; //Lee los datos dados por el sensor en la entrada analógica .
2 int led = 8; //Es el valor que tiene la entrada digital para el primer LED.
3 int led2 = 7; //Es el valor que tiene la entrada digital para el segundo LED.
4 int led3 = 6; //Es el valor que tiene la entrada digital para el tercer LED.
5 int led4 = 5; //Es el valor que tiene la entrada digital para el cuarto LED.
6 int led5 = 4; //Es el valor que tiene la entrada digital para el quinto LED.
7 int led6 = 3; //Es el valor que tiene la entrada digital para el sexto LED.
8 int led7 = 2; //Es el valor que tiene la entrada digital para el séptimo LED.
9
10 //Se indica qué configuración va a tener cada variable.
11 void setup() {
12 //Se indica que cada led es de tipo salida.
13   pinMode(led, OUTPUT);
14   pinMode(led2, OUTPUT);
15   pinMode(led3, OUTPUT);
16   pinMode(led4, OUTPUT);
17   pinMode(led5, OUTPUT);
18   pinMode(led6, OUTPUT);
19   pinMode(led7, OUTPUT);
20   Serial.begin(9600); //Es la velocidad con que se establece la comunicación entre la placa y la computadora.
21 }
22
23 //Se inicia el ciclo que repetirá
24 void loop() {
25
26   int vol = analogRead(sensor); //Variable que toma los datos dados por el sensor en la entrada analógica.
27   Serial.println(vol); //Muestra los valores que toma la variable vol, lo que nos permite ajustar el sensor y los rangos.
28
29 //Iniciamos los LEDs en apagado.
30   digitalWrite(led, LOW);
31   digitalWrite(led2, LOW);
32   digitalWrite(led3, LOW);
33   digitalWrite(led4, LOW);
34   digitalWrite(led5, LOW);
35   digitalWrite(led6, LOW);
36   digitalWrite(led7, LOW);
37
38 //Colocamos los condicionales que indican la acción que hará cada LED en cada rango determinado.
39 //Asignamos el primar valor de 410 ya que es un valor que detecta de buena forma sonidos cercanos y no se perturba por sonidos lejanos.
40   if (vol >= 410 && vol <= 420) {
41     digitalWrite(led, HIGH);
42     digitalWrite(led2, LOW);
43     digitalWrite(led3, LOW);
44     digitalWrite(led4, LOW);
45
46     digitalWrite(led5, LOW);
47     digitalWrite(led6, LOW);
48     digitalWrite(led7, LOW);
49     delay(20); //Los delay se colocan para que se pueda observar por mayor tiempo el led encendido, a mayor tiempo, mayor tiempo encendido.
50   }
51   if (vol >= 430 && vol <= 440) {
52     digitalWrite(led, HIGH);
53     digitalWrite(led2, HIGH);
54     digitalWrite(led3, LOW);
55     digitalWrite(led4, LOW);
56     digitalWrite(led5, LOW);
57     digitalWrite(led6, LOW);
58     digitalWrite(led7, LOW);
59     delay(20);
60   }
61   if (vol >= 450 && vol <= 460) {
62     digitalWrite(led, HIGH);
63     digitalWrite(led2, HIGH);
64     digitalWrite(led3, HIGH);
65     digitalWrite(led4, LOW);
66     digitalWrite(led5, LOW);
67     digitalWrite(led6, LOW);
```

```

67     digitalWrite(led7, LOW);
68     delay(20);
69 }
70 if (vol>=470&&vol<=480) {
71     digitalWrite(led, HIGH);
72     digitalWrite(led2, HIGH);
73     digitalWrite(led3, HIGH);
74     digitalWrite(led4, HIGH);
75     digitalWrite(led5, LOW);
76     digitalWrite(led6, LOW);
77     digitalWrite(led7, LOW);
78     delay(20);
79 }
80 if (vol>=490&&vol<=500) {
81     digitalWrite(led, HIGH);
82     digitalWrite(led2, HIGH);
83     digitalWrite(led3, HIGH);
84     digitalWrite(led4, HIGH);
85     digitalWrite(led5, HIGH);
86     digitalWrite(led6, LOW);
87     digitalWrite(led7, LOW);
88     delay(20);
89 }
90 if (vol>=510&&vol<=520) {
91     digitalWrite(led, HIGH);
92     digitalWrite(led2, HIGH);
93     digitalWrite(led3, HIGH);
94     digitalWrite(led4, HIGH);
95     digitalWrite(led5, HIGH);
96     digitalWrite(led6, HIGH);
97     digitalWrite(led7, LOW);
98     delay(20);
99 }
100 if (vol>=530) {
101     digitalWrite(led, HIGH);
102     digitalWrite(led2, HIGH);
103     digitalWrite(led3, HIGH);
104     digitalWrite(led4, HIGH);
105     digitalWrite(led5, HIGH);
106     digitalWrite(led6, HIGH);
107     digitalWrite(led7, HIGH);
108     delay(20);
109 }
110 }

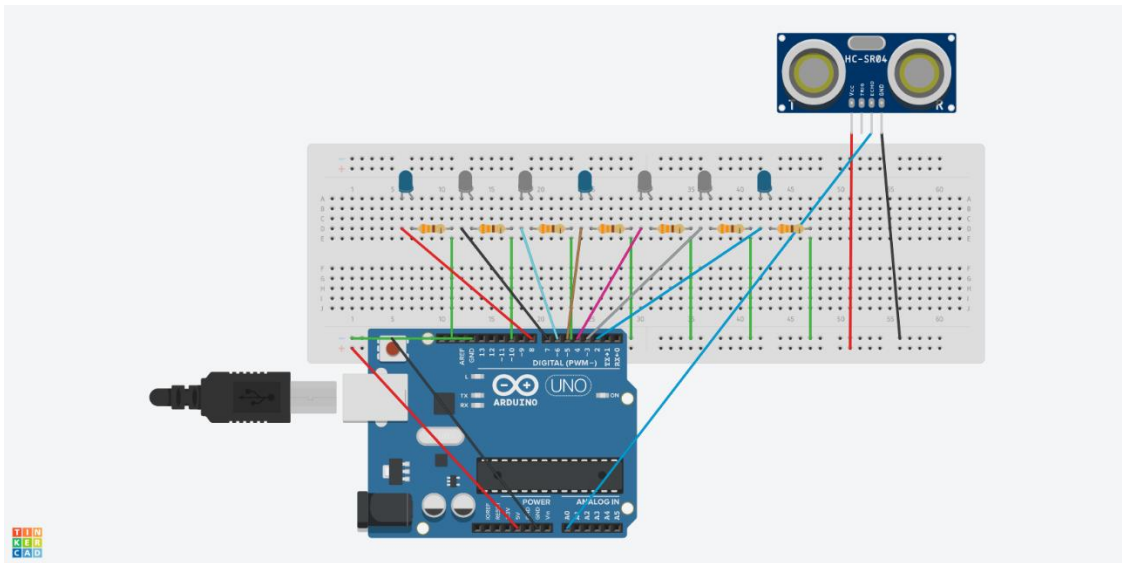
```

En cuanto al sensor, se calibra dependiendo de la sensibilidad con la que va a detectar los sonidos, en este caso está en un valor entre 380 y 390, para que no reaccione tan fácilmente a un sonido y sea un volumen moderado el que requiera para funcionar.

El “delay” no debe tener un valor alto, ya que puede ocasionar que vaya desincronizado con el audio a reproducir.



## Circuito electrónico

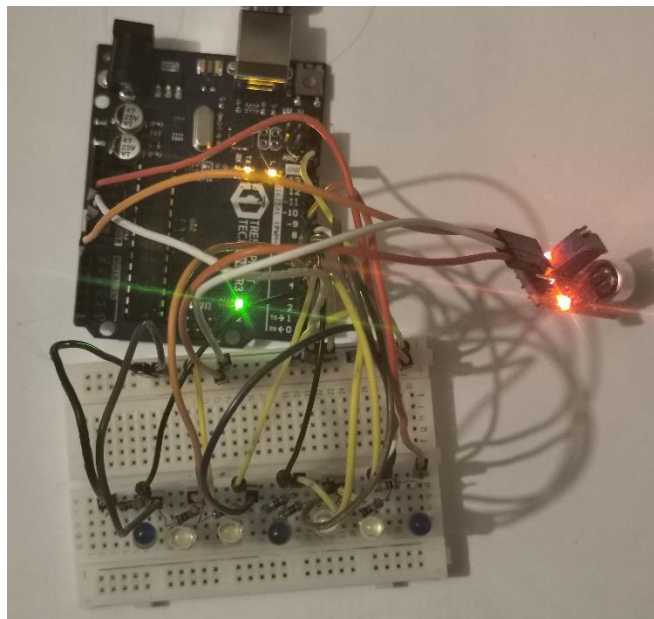


Circuito realizado con Tinkercad

Nota: Se ha optado por utilizar un sensor similar en cuanto pines, debido a que el simulador no incluye el simulador con el que se trabaja. El pin “VCC” se refiere al pin “+”, el pin “GND” es el mismo es el sensor de sonido, y el pin “ECHO” se refiere al pin “A0”, que va conectado a la entrada analógica 0.

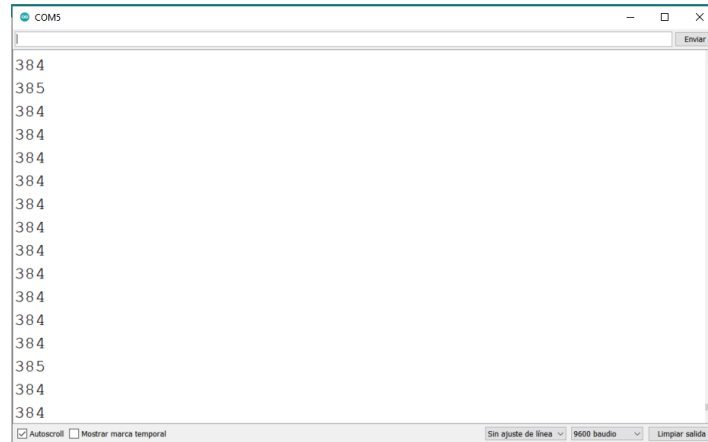
## Resultados

El circuito se ve de la siguiente manera:



Cuando conectamos el USB al puerto de la computadora, podemos abrir el Serial Plotter y el Monitor Serie, que nos indican los valores analógicos que registra el sensor, conectado en la entrada analógica 0.

Salida del Monitor Serie:



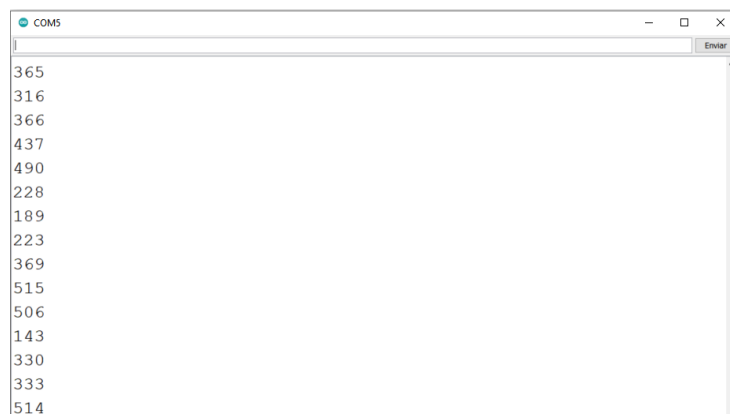
Salida del Serial Plotter:



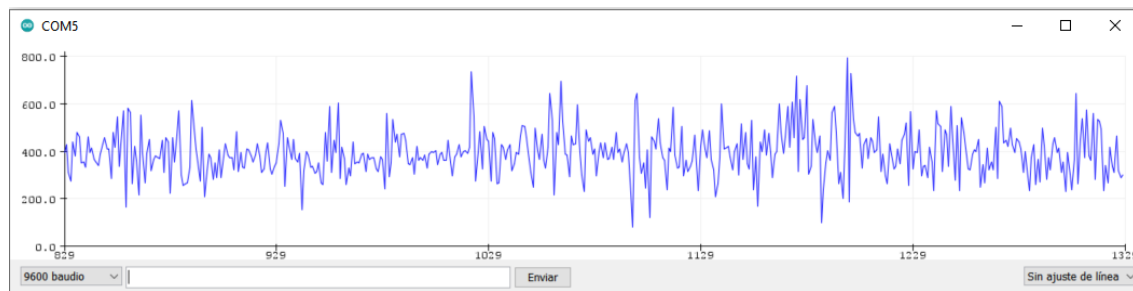
Estos valores son los que el sensor lee al no estar reproduciendo algún sonido cerca, como se ve hay más movimiento en el Serial Plotter.

Cuando se reproduce una canción cerca del sensor, los valores son los siguientes:

Monitor Serie:



## Serial Plotter:



Los resultados del funcionamiento de las luces LED se encuentra en el video:

<https://youtu.be/pSyvue9JF94>

Ejemplo extra del funcionamiento del dispositivo:

<https://youtu.be/ao0G-QeWymI>

El repositorio con el programa y el diagrama de armado del dispositivo está en la siguiente liga:

<https://github.com/LauraR-3/proyectofinal>

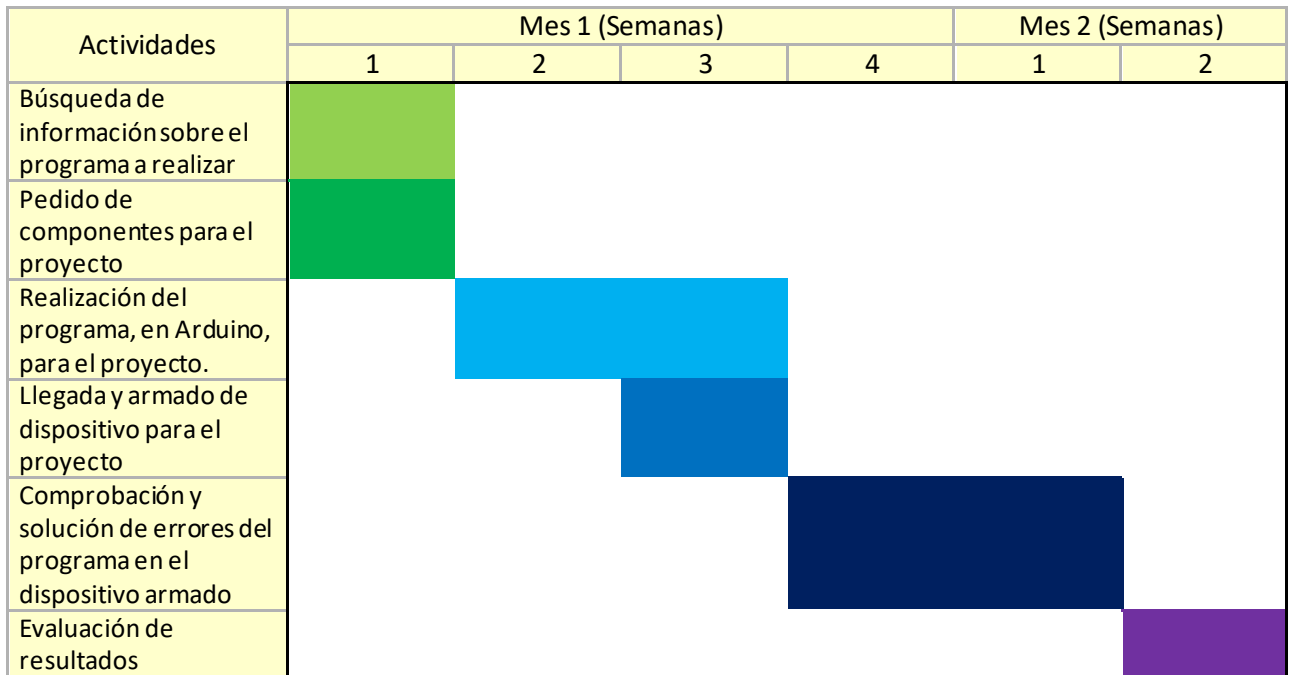
- **Comparativa de recursos informáticos**

	Octubre 2020	Enero 2021
Elementos requeridos (Hardware)	Dos luces LED, placa Arduino UNO, protoboard, dos resistencias, cables conectores, PC	Siete luces LED, placa Arduino UNO, protoboard, siete resistencias, dos sensores de sonido, cables conectores, PC
Elementos requeridos (Software)	Software Arduino y diseñador de diagramas en Arduino	Software Arduino y Tinkercad
Costos estimados	\$500	\$700

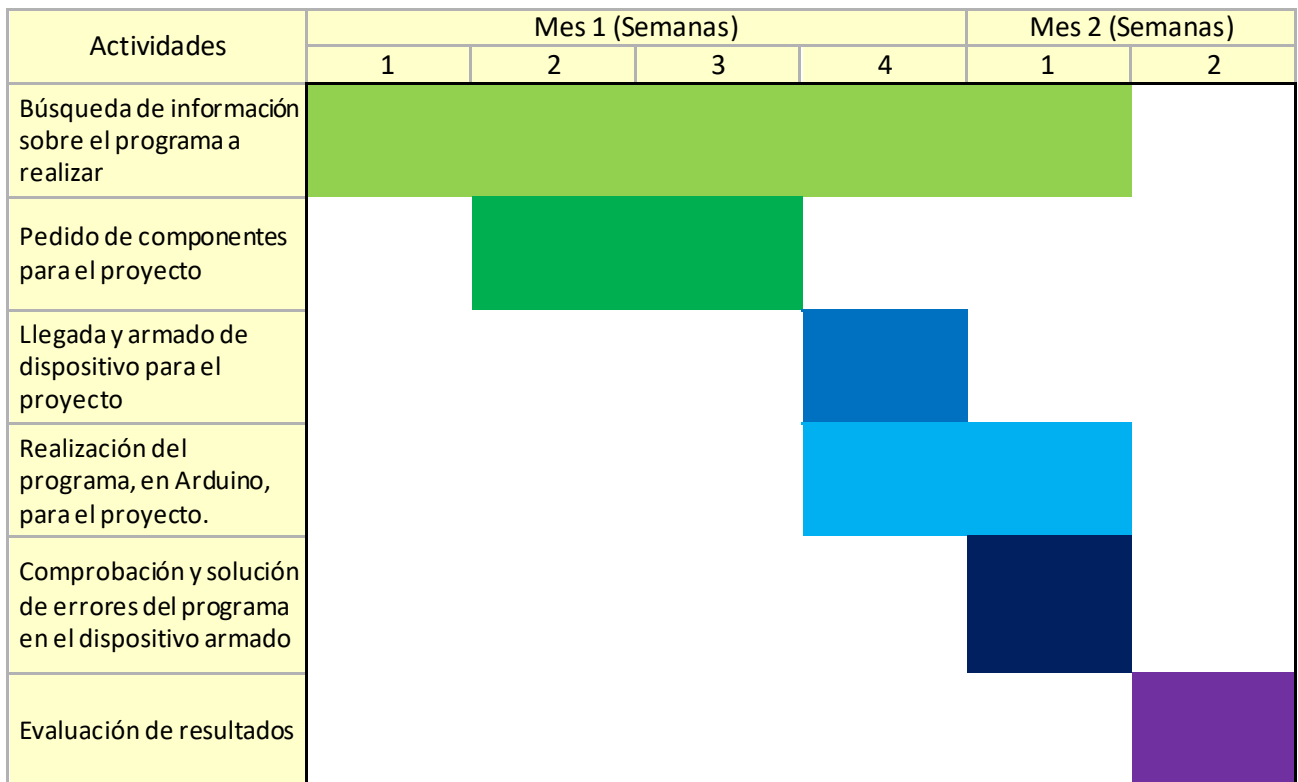
Los costos pueden variar dependiendo del lugar donde se obtengan los componentes, ya que en muchas ocasiones están incluidos en paquetes con más componentes, lo que aumenta su precio. El valor nombrado aquí es un estimado sólo considerando los materiales requeridos.

- Comparativa de diagramas de Gantt

**Octubre 2020:**



**Enero 2021:**



## Guía de usuario

Esta guía indica los pasos para realizar el dispositivo en caso de contar con los materiales requeridos, que son:

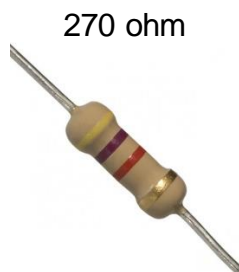
- Placa Arduino o compatible con Arduino. Se puede utilizar cualquier modelo, pero en este caso se utilizó una placa tipo Arduino UNO.



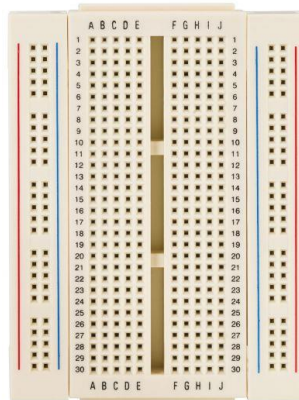
- Siete diodos LED, color indistinto.



- Siete resistencias de 270 o 330 ohm.



- Protoboard

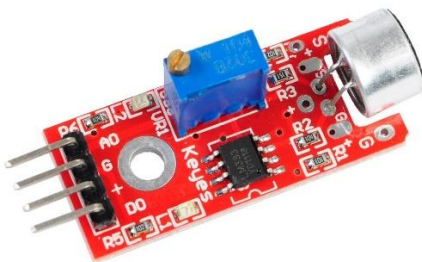


- 19 cables Dupont, 16 macho-macho y 3 macho-hembra.



- Sensor de sonido, modelo KY-037 o KY-038.

Modelo KY-037



Modelo KY-038



- Cable de USB tipo A a USB tipo B.



- Destornillador de punta plana (preferentemente de tamaño pequeño).

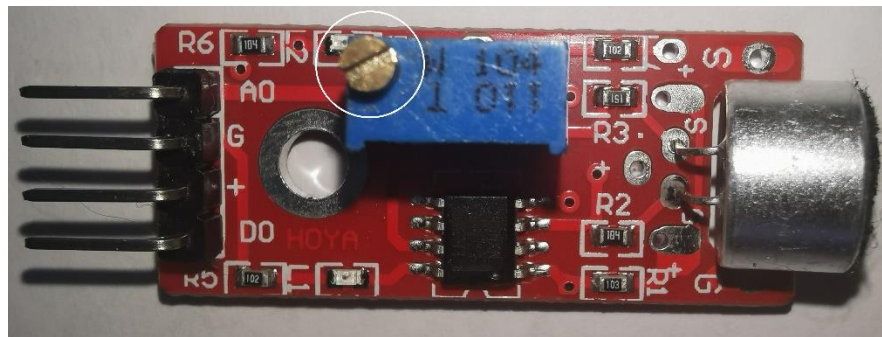


#### Instrucciones:

1. Descargar el software de Arduino de: <https://www.arduino.cc/en/software> y seleccionar la opción que se acomode a su sistema operativo.
2. Una vez instalado, se realiza el código dado anteriormente y se compila.
3. Se realiza el circuito mostrado anteriormente. Nota: Revise con cuidado las conexiones y los polos a los que está conectado, esto puede hacer que su placa no encienda o no llegue electricidad al circuito.
4. Se conecta la placa con el circuito armado al puerto USB de la PC.
5. Se sube el programa a la placa, en caso de que muestre errores intente desconectar y conectar el cable USB o reiniciar el programa.

Para ajustar la sensibilidad del sensor se hace lo siguiente:

1. Se ubica el potenciómetro del sensor y el destornillador se ubica en el siguiente punto:



2. Para aumentar su sensibilidad se gira en sentido contrario a las manecillas del reloj, para disminuirla en sentido de las manecillas del reloj. Al hacerlo procurar no mover demasiado el sensor y agarrarlo de los laterales, ya que esto nos permite tener valores más exactos al calibrarlo.



## Conclusión

Este proyecto es un buen referente de GNU, ya que Arduino es un software de uso libre, que te permite codificar y hacer uso de hardware que puede ser fabricado por ellos u otros fabricantes independientes.

Hablando del software utilizado, es una gran forma de conectar a la población con la tecnología de forma sencilla, lo complicado es que para que haya un impacto realmente de lo realizado en Arduino es necesario adquirir componentes que muchas veces no son accesibles para gran parte de la población.

Con este proyecto la intención es darle al usuario algo para su entretenimiento o incluso que pueda tener un acercamiento básico a lo que es la computación o la robótica e implementar el uso de estos componentes como parte de nuestro día a día, lo cual puede ser programado fácilmente, ya que se puede descargar fácilmente el software en Internet y es un código no complejo.

Los conocimientos no se limitan sólo a detectar un sonido gracias a un sensor, ya que lo aprendido aquí es la estructura que debe tener un programa, cómo se declaran las variables, las acciones que puede realizar una luz LED y cómo debe ser la conexión de estos circuitos. Esto puede orillar a que se desarrollen más y más proyectos en este software y que sean vitales, como para detectar latidos cardiacos, temperatura, movimientos, etc.

Personalmente, este proyecto fue más complicado de lo que esperaba, debido a que en algún momento no conté con el suficiente dinero para conseguir los componentes, lo que retrasó la elaboración del proyecto, pero afortunadamente se pudieron obtener en tiempo para realizarlo.

Quise realizarlo porque es un software que no manejé tanto como hubiera querido, por lo que me hizo descubrir más usos y relacionarlo con algo que siempre me llamó la atención, superé mis expectativas y aunque no sea un proyecto tan complejo o con un código o sentencias muy elaboradas, cumple con el propósito y da resultados satisfactorios.

## Referencias

Cibergrafía:

- (2016), “*PARPADEO DE LEDS AL SON DE LA MÚSICA.*”, consultado en diciembre de 2020 de: <http://www.iescamp.es/miarduino/2016/02/15/parpadeo-de-leds-al-son-de-la-musica/>
- López M., (2013), “*LEDs al ritmo de la música con Arduino*” consultado en enero de 2021 de: <https://miguelangellv.wordpress.com/2013/11/25/leds-al-ritmo-de-la-musica-con-arduino/>



- (2011), “*Vúmetro – Leds al ritmo de la música*”, consultado en enero del 2021 de: <http://playingwitharduino.blogspot.com/2011/12/vumetro-leds-al-ritmo-de-la-musica.html>
- Llamas L., (2016), “*DETECTAR SONIDO CON ARDUINO Y MICRÓFONO KY-038*”, consultados en enero del 2021 de: <https://www.luisllamas.es/detectar-sonido-con-arduino-y-microfono-ky-038/>

Videos de Youtube:

- “*Arduino: Sensor de sonido (KY-037 y KY-038 37in1 kit) | TechKrowd*”, Tech Krowd, 22 de agosto de 2016, consultado en enero de 2021 en: <https://youtu.be/rCB677XbpGA>
- “*COMO FUNCIONA UN SENSOR DE SONIDO ANALÓGICO?|| PROGRAMACIÓN EN ARDUINO*”, ELECTROALL, 20 de junio de 2019, consultado en enero de 2021 en: [https://www.youtube.com/watch?v=he\\_r9p8nw78](https://www.youtube.com/watch?v=he_r9p8nw78)
- “*Arduino desde cero en Español - Capítulo 13 - Sensor de Sonido KY-038 y KY-037 (modo digital)*”, Bitwise Ar, 7 de mayo de 2017, consultado en enero de 2021 en: <https://www.youtube.com/watch?v=wfiSH0CkInE&t=136s>