



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA

CAMPUS DE GANDIA

---

***Proyecto GTI\_2A (GTI\_2A\_20\_TEAM\_12)***

***Murbin - Farolas inteligentes***

***Documento Técnico de Diseño***

***Volumen 2***

## ÍNDICE

<b>1</b>	<b>INTRODUCCIÓN .....</b>	<b>1</b>
1.1	PROPÓSITO .....	1
1.2	ALCANCE.....	1
1.3	EQUIPO DE DESARROLLO E INTERESADOS EN EL PROYECTO .....	4
1.4	DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS .....	6
1.5	ESTRUCTURA DEL DOCUMENTO .....	6
1.6	REPOSITORIO GIT Y LISTADO DOCUMENTAL.....	6
<b>2</b>	<b>SISTEMA.....</b>	<b>7</b>
2.1	CARACTERÍSTICAS DEL SISTEMA.....	7
2.2	ARQUITECTURA DEL SISTEMA .....	9
<b>3</b>	<b>RED DE NODOS .....</b>	<b>11</b>
<b>4</b>	<b>CONCLUSIONES Y LÍNEAS DE MEJORA .....</b>	<b>22</b>
<b>5</b>	<b>REFERENCIAS.....</b>	<b>23</b>

# 1 INTRODUCCIÓN

Murbin (Mobiliario urbano inteligente) plantea un prototipo de alumbrado público inteligente. Más concretamente, en nuestro caso, ser un sistema inteligente implica varias cosas: la conexión a Internet, la capacidad de realizar lecturas de diversas magnitudes físicas, así como la transmisión de éstas a posteriori y la activación automática de comportamientos preestablecidos (actuadores) en función del valor de medidas nominales. Por tanto, se trata de un proyecto que recae sobre el marco del Internet de las cosas (IoT) y requiere de numerosas tecnologías y disciplinas de la ingeniería.

Se recopila información sensible de campo, a fin de monitorizar y optimizar la actividad ambiental (intensidad de la luz, contaminación, nivel de ruido, presencia humana, geolocalización del dispositivo, etc.). Se propone implementar también un mecanismo de autoabastecimiento (célula fotovoltaica, batería y convertidor buck-boost). Para minimizar el consumo energético, se prevee la realización de las mediciones cada cierto tiempo. Ésto es configurable. Inmediatamente entraría en modo de hibernación o stand-by, hasta la siguiente captura de datos o el acaecimiento de una interrupción producida por alguna subrutina.

Para implementar las susodichas funcionalidades, se ha optado por utilizar el microcontrolador **ESP32** (como esclavo), y el micro-ordenador **Raspberry Pi 3** (como maestro). También son necesarios múltiples dispositivos y componentes electrónicos (resistencias, amplificadores operacionales, diodos, termistores, sensores específicos, ...). El ESP32 puede ser programado utilizando la plataforma de desarrollo Arduino.

## 1.1 PROPÓSITO

En este documento se especifica el planteamiento inicial de las características técnicas del producto, así como su funcionamiento. Además, proporciona información sobre los dispositivos y tecnologías utilizados. Éste es un documento técnico dirigido a un público con conocimiento tecnológico mínimo, relativo al campo de la electrónica y la programación. Su redacción pretende exponer de manera rigurosa y objetiva las capacidades, propiedades y limitaciones del sistema. Con ello, se espera concretar las amplias posibilidades del producto.

## 1.2 ALCANCE

El término de alumbrado público inteligente hace referencia a alumbrado capaz de detectar diferentes niveles de magnitud y adaptar la intensidad de la luz de manera proporcional. Es decir, para diferentes condiciones la actuación de la farola será diferente. La intensidad baja cuando no se detecta movimiento y, por el contrario, sube cuando se detecta. Esto supone un cambio drástico de paradigma; contrasta con el tradicional alumbrado estático de intensidad variable por tiempos fijos (programado con reloj). Las posibilidades se extienden más allá de la luminosidad, permitiéndonos interactuar con los usuarios del medio público de manera acorde a la situación. Otro ejemplo sería el de cambiar la temperatura de los haces de luz en función del nivel de ruido del entorno. Así pues, es un sistema directamente ligado al desarrollo y la evolución de las Smart Cities [1] [2] [3].

Aclarado esto, los objetivos del producto son la optimización y reducción del consumo energético, el control de la contaminación en sus múltiples expresiones (sonora, lumínica y CO2), mejorar la experiencia de usuario del viandante y la creación de una interfaz homogénea y céntrica (aplicación Android) para consultas varias: búsqueda de parking, trazado de caminos, alertas personalizadas en función del nivel de ruido u otras magnitudes de interés, mapas de calor y un largo etcétera.

También estará equipado con un sistema de posicionamiento global, bien sea un receptor GPS físico (hardware) o un par de coordenadas geográficas fijadas durante la instalación (software). Además el dispositivo ha de ser hermético; se puede estropear si alguno de sus componentes internos entra en contacto con el agua, provocando un cortocircuito.

Cabe recalcar que solo se han mencionado algunas de las funcionalidades pensadas desde un punto de vista empático con el perfil de usuario del viandante. No obstante, se han definido otros perfiles como el del técnico, el administrador y (opcionalmente) el investigador o científico. Las funcionalidades a implementar deberán ser específicas (por lo general) al tipo de usuario en cuestión. Véanse debajo los perfiles de usuario estudiados y los porcentajes aproximados que se sopesan.

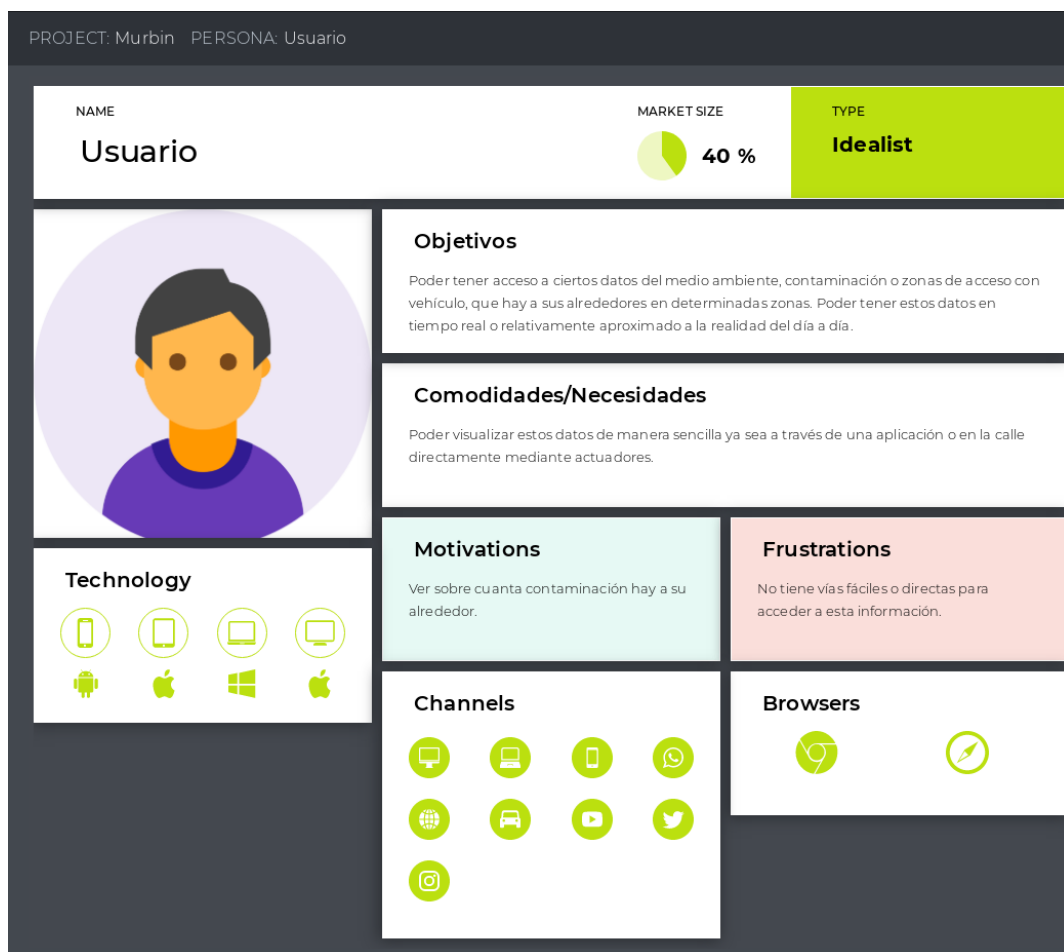


Ilustración 1. Perfil de usuario (usuario).

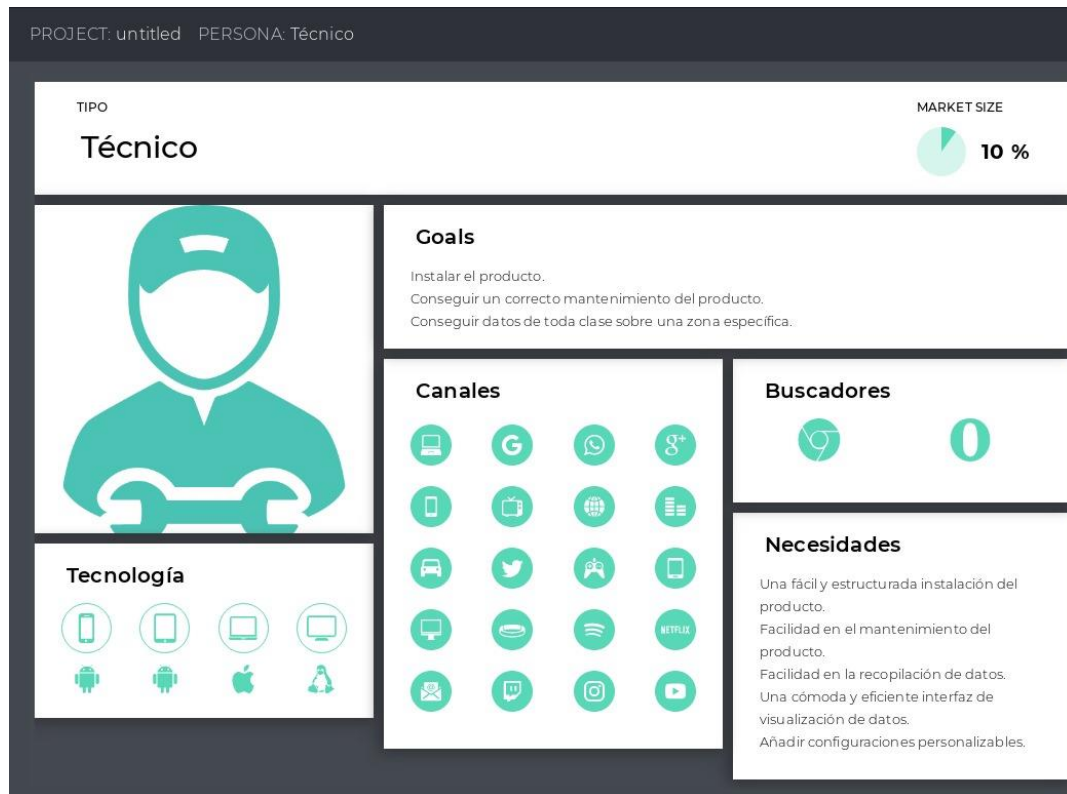


Ilustración 2. Perfil de usuario (técnico).

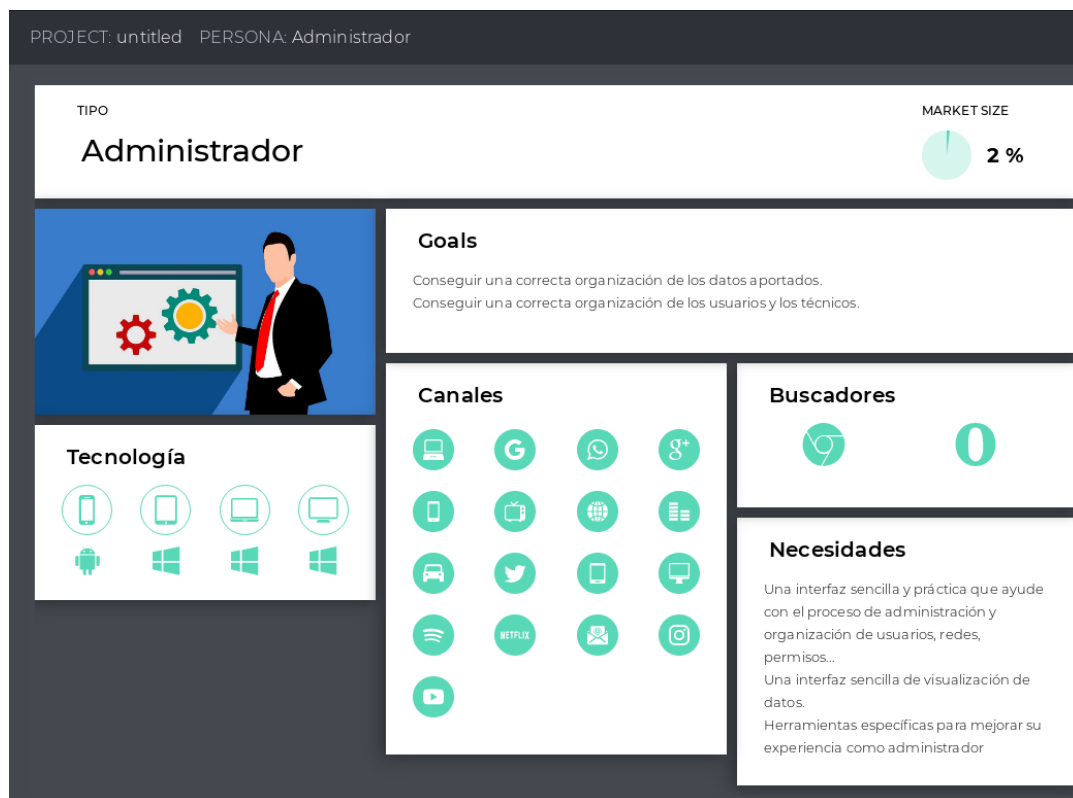


Ilustración 3. Perfil de usuario (administrador).

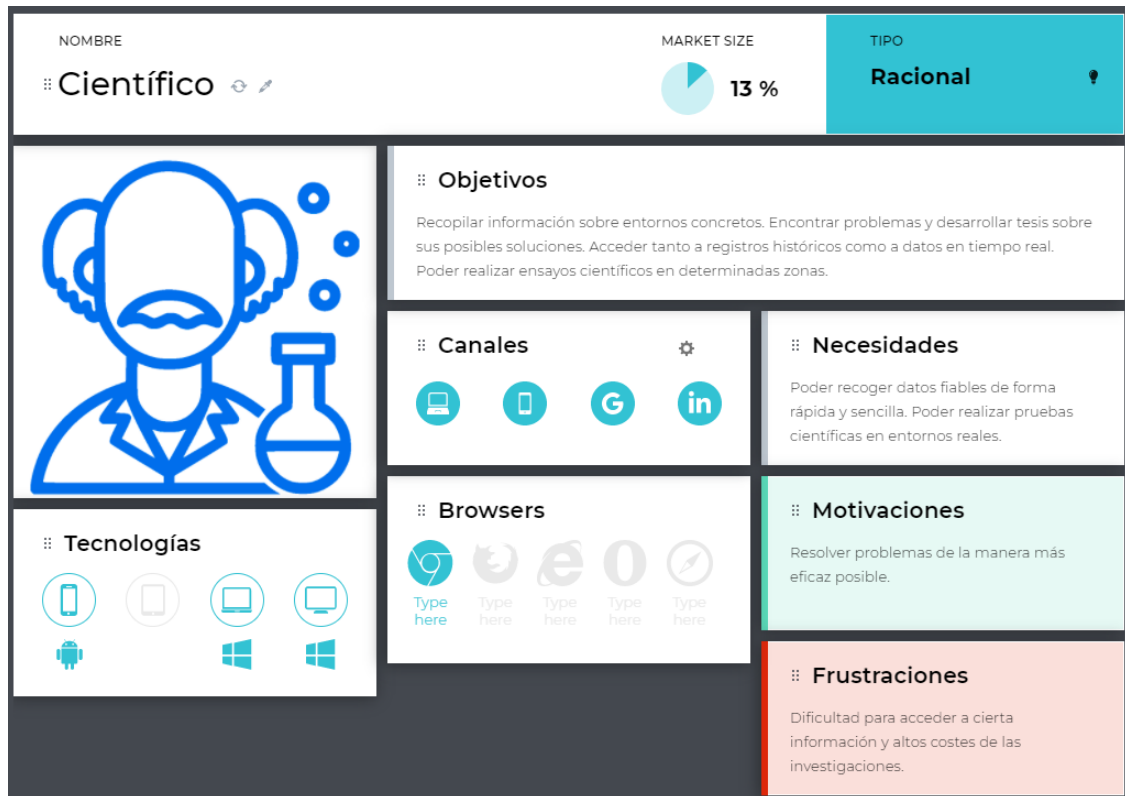


Ilustración 4. Perfil de usuario (científico).

### 1.3 EQUIPO DE DESARROLLO E INTERESADOS EN EL PROYECTO

A continuación se listan todos los integrantes del equipo de desarrollo del proyecto, así como algunas personalidades de interés por su carácter asesorativo y/o de formación.

#### ***Miembros del equipo***

Abidán Brito Clavijo

- Estudiante de 2º del grado en Tecnologías Interactivas.
- Manager del equipo.
- Desarrollo electrónico.
- Desarrollo software embebido.

Alejandro Losa García

- Estudiante de 2º del grado en Tecnologías Interactivas.
- Diseño visual y de interacción.
- Maquetación interfaz gráfica.

Francisco Javier Paños Madrona

- Estudiante de 2º del grado en Tecnologías Interactivas.
- Desarrollo de aplicación Android.

Yeray Candel Sampedro

- Estudiante de 2º del grado en Tecnologías Interactivas.
- Desarrollo electrónico.
- Desarrollo software embebido.

Raúl De La Fe Robles

- Estudiante de 2º del grado en Tecnologías Interactivas.
- Generalista.

***Otros interesados en el proyecto***

José Marín-Roig Ramón

- Profesor Titular en Universitat Politècnica de València.
- Product Owner.
- Asesor.

Jesús Tomás Gironés

- Profesor Titular en Universitat Politècnica de València.
- Formación teórica Android.
- Asesor.

Stella María Heras Barberá

- Profesora Asociada en Universitat Politècnica de València.
- Formación práctica Android.
- Asesora.

Enrique Colomar Pous

- Profesor Titular en Universitat Politècnica de València.
- Formación práctica de laboratorio (desarrollo electrónico).

Tomás Carlos Sogorb Devesa

- Profesor Titular en Universitat Politècnica de València.
- Formación teórica microprocesadores y acondicionadores de señal.

#### **1.4 DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS**

- IDE: entorno de desarrollo integrado (Integrated Development Environment).
- TCP: Transmission Control Protocol.
- IP: Internet Protocol address.
- UDP: User Datagram Protocol.
- HTTP: Hypertext Transfer Protocol.
- MQTT: Message Queuing Telemetry Transport.
- UART: Universal Asynchronous Receiver-Transmitter.
- GPS: Global Positioning System.

#### **1.5 ESTRUCTURA DEL DOCUMENTO**

El documento se estructura de la siguiente manera:

- /1 La Sección 1 es la introducción e incluye una descripción detallada del proyecto y los documentos referenciados.
- /2 La Sección 2 proporciona una visión general del sistema.
- /3 La Sección 3 incluye el mapa de la red de nodos especificando las diferentes capas
- /4 La Sección 4 presenta las conclusiones del trabajo desarrollado y las posibles líneas de mejora.
- /5 Por último, se presentan las referencias del documento

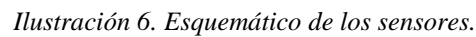
#### **1.6 REPOSITORIO GIT Y LISTADO DOCUMENTAL**

Para la base de código y documentos pertinentes a este proyecto, se ha utilizado un repositorio Git como sistema de control de versiones, alojado en la plataforma GitHub. Para acceder al repositorio, haga click sobre el hipervínculo aquí expuesto:

<https://github.com/abidanBrito/murbin>



Nuestro sistema realiza lecturas analógicas de los diversos sensores a través de los pines **GPIO**. La mayoría de estos pines se encuentran conectados internamente a un **ADC** (convertidor analógico-digital).



La directiva 2011/65/UE es aplicable. El producto también cumple la normativa europea de concentración máxima de sustancias peligrosas. El reciclaje de aparatos eléctricos y electrónicos es una cuestión sensible. Éstos están formados por diferentes materias peligrosas. La negligencia de su correcto tratamiento o reciclaje puede suponer contaminación ambiental. Siguiendo el Real Decreto 110/2015, se insta a extraer los componentes internos de la caja y llevarlos al punto limpio más cercano; depositando la caja y los residuos electrónicos por separado y en sus respectivas localizaciones. El reciclaje de los RAEE depende del papel activo de Documento Técnico de Diseño Pág.9 GTI 1A 19 TEAM 02 Vol. 1 cada uno de nosotros. Debemos concienciarnos y entender la responsabilidad y deuda ecológica que ello conlleva. Todos los sensores implementados cumplen la normativa RoHS: MPU-9250, BMP280, A2235-H.

*Ilustración 7. Diagrama de conexiones del sistema.*

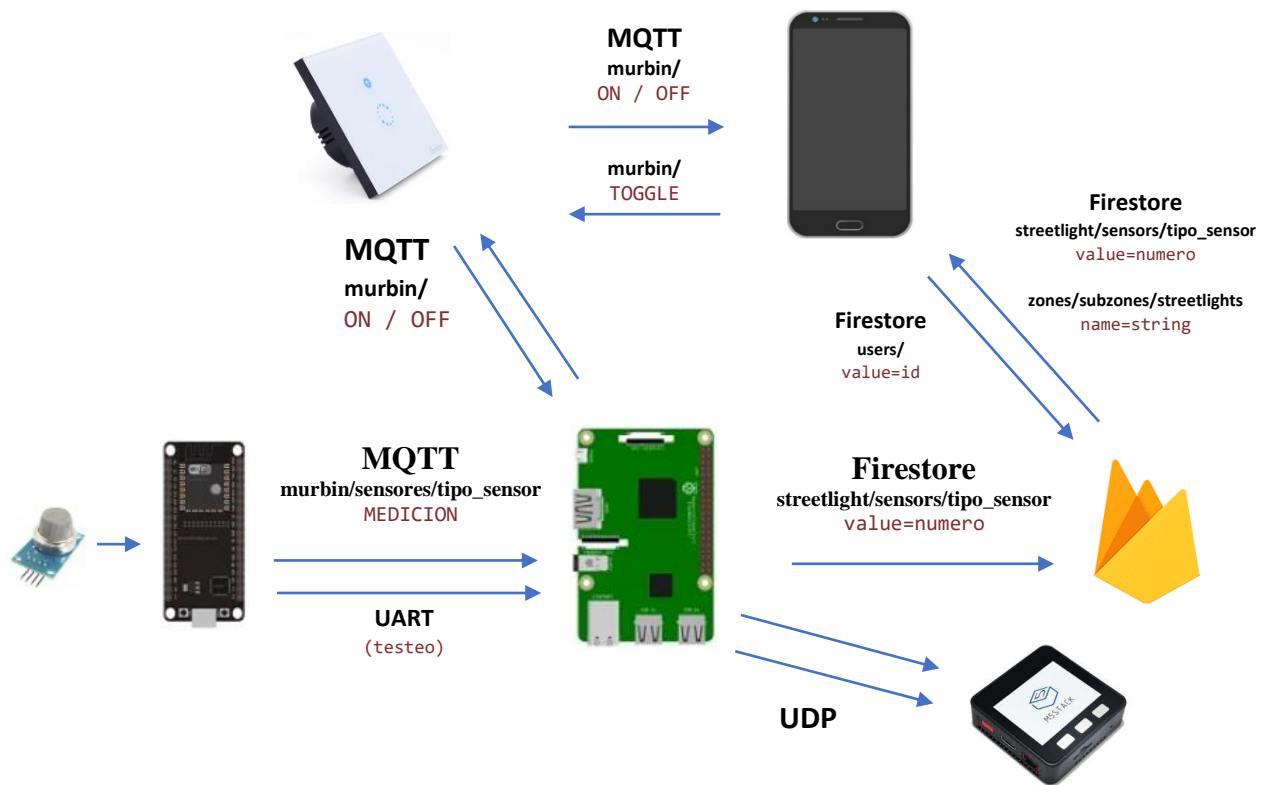


Ilustración 8. Diagrama de Comunicaciones del sistema.

### 3 RED DE NODOS

Se puede desglosar el camino de los datos de manera esquemática en los siguientes 5 niveles:

1. Comunicación hardware (nivel 0).
2. Acceso a la red (nivel 1).
3. Internet (nivel 2).
4. Transporte (nivel 3).
5. Aplicación (nivel 4).

En primer lugar nuestro microcontrolador realiza lecturas analógicas de los sensores a través de los pines **GPIO**, y con la Raspberry Pi mediante el protocolo **UART** (con la finalidad de enviar los datos capturados). Éstos, son enviados mediante conexión inalámbrica (Wi-Fi) a un servidor en la nube y almacenados en una **base de datos NoSQL** (Firebase), utilizando el protocolo **IP**. Ya que la conexión a Internet es la función que más consume, y no es necesario enviar datos en grandes cantidades, se podría utilizar el protocolo **TCP** o el **UDP**. Finalmente estos datos acaban en el servidor web mediante el protocolo **HTTP**. Nótese que, en la práctica no sería necesario tener el prototipo conectado a ningún otro dispositivo, pues es autosuficiente y los datos serían enviados directamente al servidor. A continuación, adjuntamos la siguiente ilustración a modo de ayuda visual. También se utiliza el protocolo **MQTT** como pasarela de comunicación entre los distintos dispositivos, para el envío y recepción de datos.

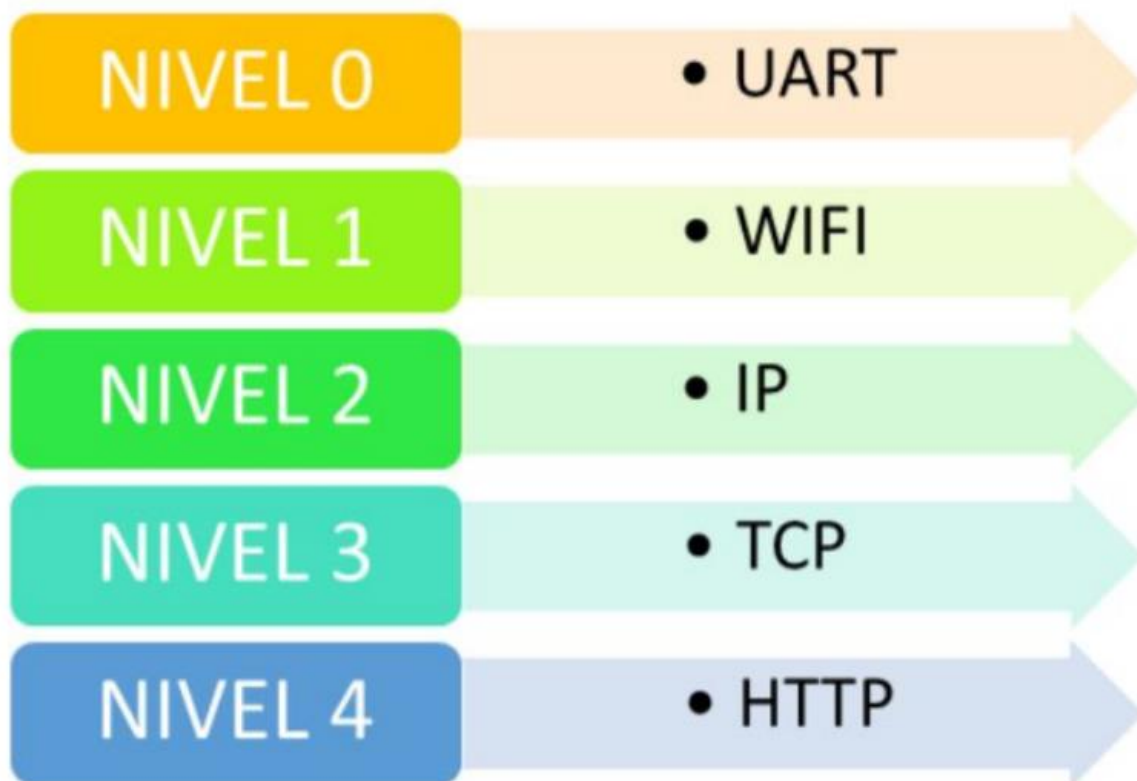


Ilustración 9. Red de nodos.

## 4 FUNCIONALIDADES SENSOR

### 4.1 SENSOR KY-038

El sensor de sonido KY038 para Arduino está diseñado con un pequeño micrófono electreto que detecta las variaciones de ruido del ambiente.

Los micrófonos electret tienen una respuesta en frecuencia bastante buena (50 a 15.000 Hz) y una sensibilidad entre -50 dB y -70 dB, aunque lejana de la de los micrófonos de condensador, que son mucho más sensibles en la zona de los agudos). Además, es una respuesta poco plana.

Entre sus principales ventajas destaca su insensibilidad a la humedad y el calor (aunque la humedad y las partículas causen un cortocircuito en parte del diafragma, siempre se obtiene señal eléctrica a la salida).

El principal inconveniente que presentan los micrófonos electret es el polvo, que deteriora su rendimiento con el uso. Cuando un micrófono electret empieza a producir *zumbidos* (ruidos) inexplicables, es una indicación de que debe ser sustituido, ya que ha terminado su vida activa.

Puedes utilizar este sensor para tus proyectos que requieran la detección de ruido, como ser alarmas o interruptores con detección de voz o sonido (un aplauso, por ejemplo).

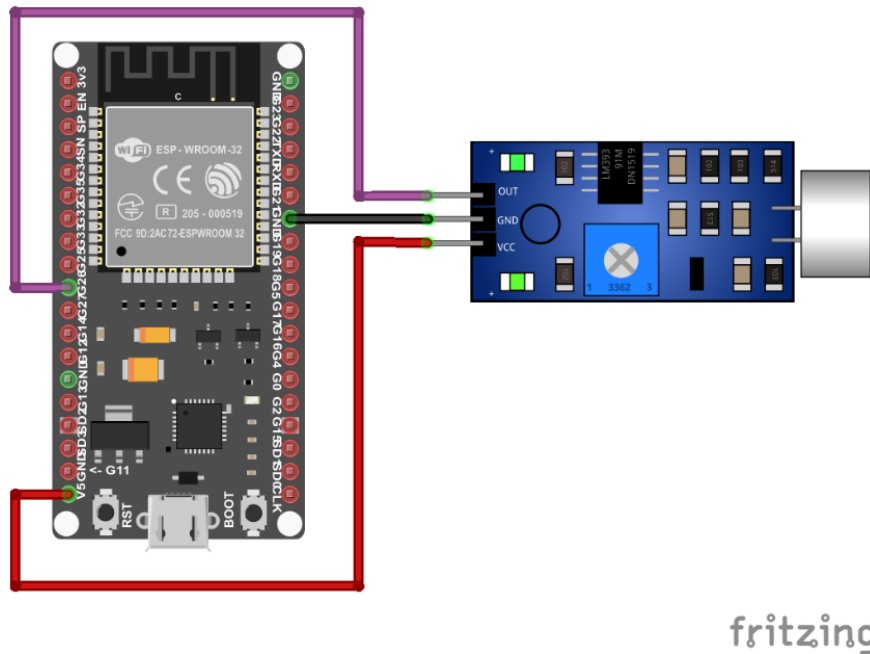
De manera general, estos son algunos rangos de operación típicos que son útiles de tenerlos en cuenta a la hora de implementar un circuito con este sensor.

- Voltaje de trabajo: 5.0V
- Muy bajo consumo de corriente
- Temperatura de trabajo:  $-40 < T < +85$  °C

**Tabla de conexión**

Pines Arduino	Pines Sensor
5V	+
GND	G
A0	A0
2	D0

#### 4.1.1.1 Circuito del sensor de sonido KY038



## 4.2 SENSOR PIR

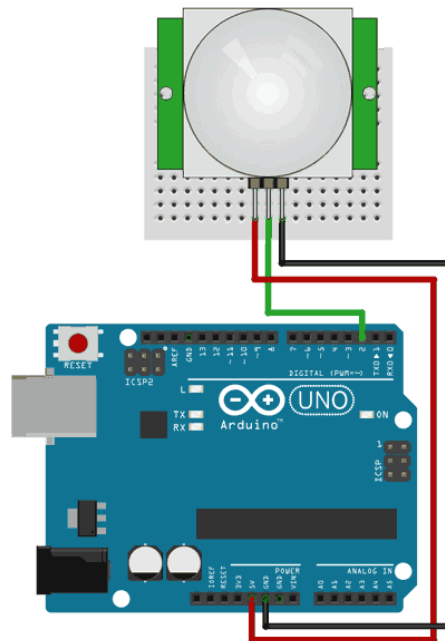
Los detectores PIR (Passive Infrared) o Pasivo Infrarrojo, reaccionan sólo ante determinadas fuentes de energía tales como el calor del cuerpo humano o animales. Básicamente reciben la variación de las radiaciones infrarrojas del medio ambiente que cubre. Es llamado pasivo debido a que no emite radiaciones,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

Su componente principal son los sensores piroeléctricos. Se trata de un componente electrónico diseñado para detectar cambios en la radiación infrarroja recibida. Generalmente dentro de su encapsulado incorporan un transistor de efecto de campo que amplifica la señal eléctrica que genera cuando se produce dicha variación de radiación recibida.

La información infrarroja llega al sensor piroeléctrico a través de una lente de fresnell que divide el área protegida en sectores. Se distribuyen lentes con diferentes características: gran angular, cortina, corredor, anti mascotas, etc.

Los sensores PIR son ampliamente usados en sistemas de seguridad y en detección de intrusos en entornos domésticos y comerciales, ya que se pueden instalar en cualquier punto de entrada potencial (puertas y ventanas) y avisar de la presencia de los intrusos antes de que lleguen al sensor y lo puedan sabotear.

#### 4.2.1.1 Circuito del sensor PIR



### 4.3 SENSOR MQ4

Los sensores de gases MQ son una familia de dispositivos diseñados para detectar la presencia de distintos componentes químicos en el aire. Podemos conectar estos dispositivos a un autómata o procesador como Arduino.

Existe una gran variedad de sensores MQ. Cada modelo está diseñado para detectar una o más sustancias, pensadas para un uso específico, como por ejemplo detección gases inflamables, calidad del aire o detección de alcohol en aire respirado.

Los sensores de gases deben ser calibrados antes de obtener una medida precisa. Aun calibrados estos sensores no disponen de la garantía necesaria para formar parte de un sistema de seguridad.

Los sensores MQ están compuestos por un sensor electroquímico que varía su resistencia al estar en contacto con las sustancias.

Los sensores de gases son dispositivos con alta inercia, es decir, la respuesta necesita tiempos largos para estabilizarse tras un cambio de concentración de los gases medidos. Ello es debido a la necesidad física de que el gas abandone el material sensible, lo cual es un proceso lento.



Todos los modelos MQ disponen de un calentador necesario para elevar la temperatura del sensor, y que sus materiales adquieran la sensibilidad. Mientras el calentador no alcance la temperatura de funcionamiento, la lectura del sensor no será fiable.

El tiempo de calentamiento depende de cada modelo de sensor. En la mayoría de los modelos es suficiente para con unos pocos minutos, pero algunos modelos requieren hasta 12 y 48 horas hasta obtener mediciones estables.

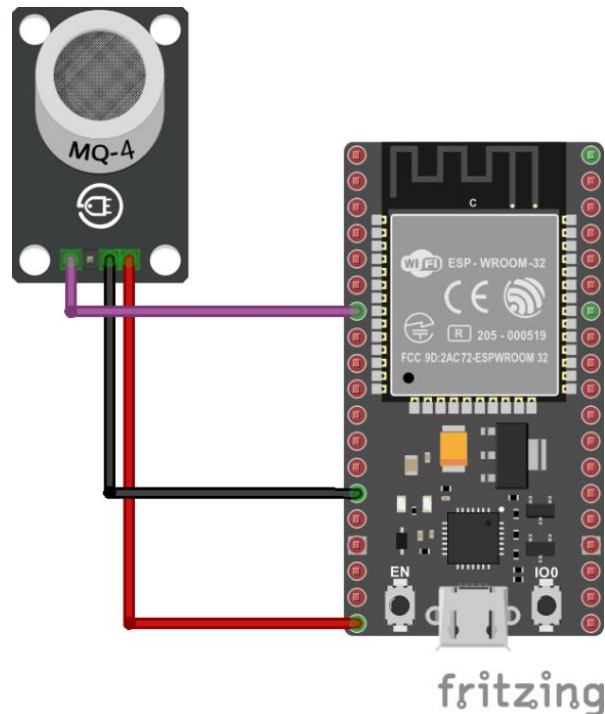
Por otro lado, cada modelo necesita su propia tensión para alimentar el calentador. En muchos modelos esta tensión es de 5V, pero algunos modelos tienen condicionantes especiales para la alimentación.

El consumo de los sensores MQ puede ser elevado debido al calor necesario para funcionar el calentador, que puede llegar hasta 800 mW en algunos modelos. Esto es superior a la potencia que puede suministrar el regulador de Arduino, por lo que será necesario proporcionar una fuente de alimentación externa.

A continuación, una tabla de resumen con los distintos modelos de sensores disponibles, los gases a los que son sensibles, y algunos datos sobre el calentador. Nosotros utilizaremos el MQ-135.

MQ-309A	Monóxido de carbono, gases inflamables	Alternado 0.2 y 0.9V
MQ-131	Ozono	6V
MQ-135	Benceno, alcohol, humo, calidad del aire	5V
MQ-136	Ácido sulfhídrico	5V
MQ-137	Amoniaco	5V

#### 4.3.1.1 Circuito del sensor de gas MQ-4



### 4.4 SENSOR LUMINOSIDAD

Un fotorresistor, o LDR (light-dependent resistor) es un dispositivo cuya resistencia varía en función de la luz recibida. Podemos usar esta variación para medir, a través de las entradas analógicas, una estimación del nivel de la luz.

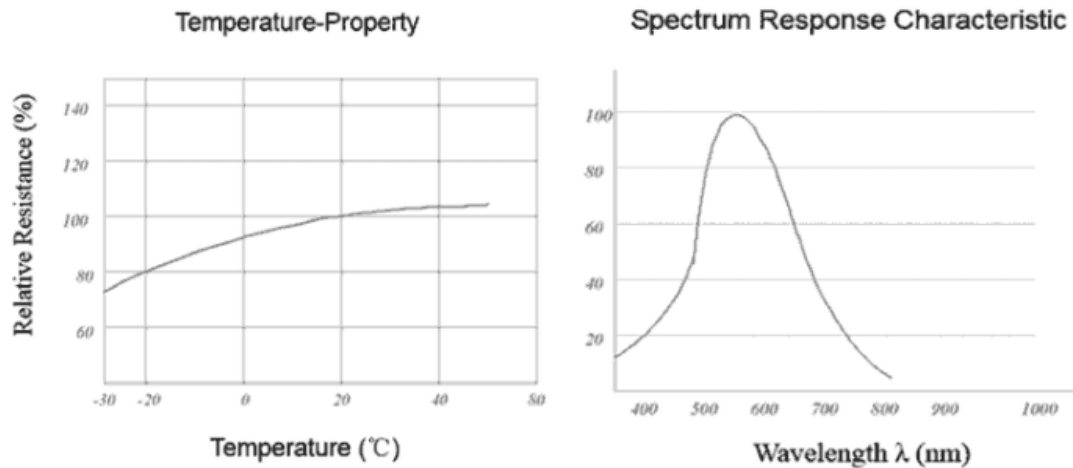
Un fotorresistor está formado por un semiconductor, típicamente sulfuro de cadmio CdS. Al incidir la luz sobre él algunos de los fotones son absorbidos, provocando que electrones pasen a la banda de conducción y, por tanto, disminuyendo la resistencia del componente.

Por tanto, un fotorresistor disminuye su resistencia a medida que aumenta la luz sobre él. Los valores típicos son de 1 Mohm en total oscuridad, a 50-100 Ohm bajo luz brillante.

Por otro lado, la variación de la resistencia es relativamente lenta, de 20 a 100 ms en función del modelo. Esta lentitud hace que no sea posible registrar variaciones rápidas, como las producidas en fuentes de luz artificiales alimentadas por corriente alterna. Este comportamiento puede ser beneficioso, ya que dota al sensor de una gran estabilidad.

Finalmente, los fotorresistores no resultan adecuados para proporcionar una medición de la iluminancia, es decir, para servir como luxómetro. Esto es debido a su baja precisión, su

fuerte dependencia con la temperatura y, especialmente, a que su distribución espectral no resulta adecuada para la medición de iluminancia.



Matemáticamente, la relación entre la iluminancia y la resistencia de una LDR sigue una función potencial.

$$\frac{I}{I_0} = \left( \frac{R}{R_0} \right)^{-\gamma}$$

Siendo  $R_0$  la resistencia a una intensidad  $I_0$ , ambas conocidas.

La constante  $\gamma$  es la pendiente de la gráfica logarítmica, o la pérdida de resistencia por década. Su valor típicamente 0.5 a 0.8.

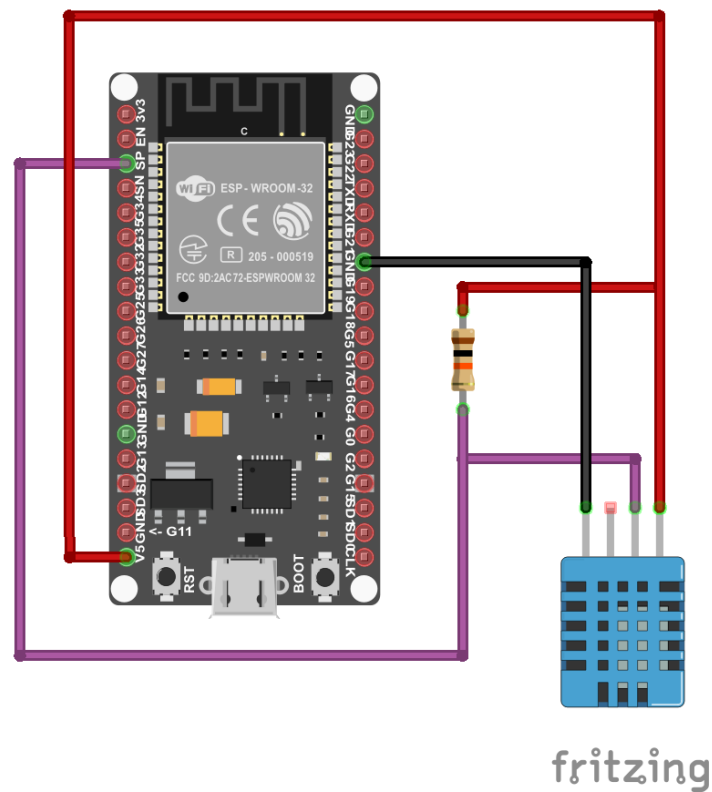
Por este motivo, frecuentemente las gráficas que relacionan ambos valores se representan en escalas logarítmicas para ambos ejes. Bajo esta representación, la relación se muestra como una gráfica lineal.

Las características del DHT11 son en realidad escasas, especialmente en rango de medición y precisión.

- Medición de temperatura entre 0 a 50, con una precisión de 2°C
- Medición de humedad entre 20 a 80%, con precisión del 5%.
- Frecuencia de muestreo de 1 muestras por segundo (1 Hz)

El DHT11 es un sensor muy limitado que podemos usar con fines de formación, pruebas, o en proyectos que realmente no requieran una medición precisa.

#### 4.5.1.1 Circuito del sensor PIR



## **5 ESTUDIO DE ENERGIA Y CONSUMO**

También se han optimizado el consumo energético al máximo, optando por soluciones renovables, limpias y efectivas, así como usando la hibernación del microcontrolador y diseñando una PCB específica para esta primera versión del prototipo, también hemos optado por la programación con tareas para reducir el uso innecesario. Todo ello de acuerdo con la normativa medioambiental vigente.



## 6 CONCLUSIONES Y LÍNEAS DE MEJORA

Tras la implementación de la segunda versión del producto (prototipo del Sprint 3) se extraen las siguientes conclusiones:

- Se ha ampliado la red de sensores interconectada de manera inalámbrica a Internet (tecnología Wi-Fi). A medida que avance el desarrollo la red de sensores seguirá creciendo en magnitud.
- Se ha terminado el wireframe (layouts) de la aplicación, teniendo en cuenta el diseño visual, el diseño de interacción y la experiencia de usuario de la aplicación móvil.
- Se ha creado terminado la base de datos NoSQL en tiempo real.
- Se ha maquetado y configurado una primera versión de la aplicación móvil (Android) con varios roles de usuario autenticados, así como el tipo de usuario invitado.
- Se han implementado nuevos actuadores y se hace uso de relés.
- Se han reemplazados los LEDs individuales por actuadores propios (anillo de LEDs y foco LED). Esto implica concebir las funcionalidades de los distintos perfiles de usuario ya definidos.
- Se han de realizar el calibrado de los sensores para el siguiente Sprint.



## REFERENCIAS

- [1] Zak Derler (2020). [Smart street lighting will help future-proof our cities.](#)
- [2] Esther Fuldauer (2019). [Lighting the road to smart cities and sustainability.](#)
- [3] Chris Teale (2020). [Cities ‘finally waking up’ to the benefits of smart streetlights: survey.](#)
- Wikipedia (2014). [Intelligent street lighting.](#)
- Wikipedia (2020). [Universal asynchronous receiver-transmitter.](#)
- Campbell, Scott (2016). [Basics of UART Communication.](#)
- Loflin, Lewis (2018). [Photodiode Circuit Operation and Uses.](#)
- Universitat Politècnica de València (2017). [Master en Desarrollo de Aplicaciones Android.](#)
- Gironés, T. (2013). *El gran libro de Android*. Barcelona: Marcombo.

## CONTROL DEL DOCUMENTO

**Título:** *Documento Técnico de Diseño*  
**Volumen:** *Vol. 2*  
**Fecha:** *21 Enero 2021*  
**Autor:** *Raúl de la Fe Robles*

**Referencia:** *GTI\_20\_2A\_TEAM\_12*  
**Nombre de fichero:** *GTI\_20\_2A\_TEAM\_12-01*

## FIRMAS DEL DOCUMENTO

Naturaleza del firmante	Nombre	Firma	Fecha	Rol
<i>Autor</i>	<i>Abidán Brito Clavijo</i> <i>Raúl de la Fe Robles</i> <i>Javier Paños Madrona</i> <i>Alejandro Losa García</i> <i>Yeray Candel Sampedro</i>		<i>21/1/2021</i>	<i>Miembros del equipo.</i>
<i>Revisor</i>				

## REGISTRO DE CAMBIOS DEL DOCUMENTO

Fecha	Versión	Autor	Detalles de los cambios
<i>20 Octubre 2020</i>	<i>Vol 1. Borrador 1</i>	<i>Abidán Brito Clavijo.</i>	<i>Primer borrador del sistema (Sprint 1).</i>
<i>19 Noviembre 2020</i>	<i>Vol 2. Borrador 1.</i>	<i>Abidán Brito Clavijo</i>	<i>Segundo borrador del sistema (Sprint 2).</i>
<i>10 Diciembre 2020</i>	<i>Vol 2. Borrador 2.</i>	<i>Abidán Brito Clavijo</i>	<i>Diagrama de conexiones, comunicaciones, corrección de los protocolos.</i>
<i>21 Enero 2021</i>	<i>Vol 2. Borrador 3.</i>	<i>Raúl de la Fe Robles</i>	<i>Actualizar los elementos del documento.</i>

			<i>Aportar información sobre: uso energético y medioambiente.</i>
--	--	--	---