



**Escuela Superior  
de Ingeniería y Tecnología**  
Universidad de La Laguna

# Trabajo de Fin de Grado

---

Sistema de apertura remoto y sin llaves

*Remote and keyless aperture system*

Daniel Ojeda Martin

---

D. **Cándido Caballero Gil**, con N.I.F. 42.201.070-A profesor ayudante doctor adscrito al Departamento de Ingeniería Informática y Sistemas de la Universidad de La Laguna, como tutor.

D. **Jose Iván Santos González**, con N.I.F. 78.637.989-T trabajador por cuenta ajena adscrito al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como cotutor.

**C E R T I F I C A N**

Que la presente memoria titulada:

*"Sistema de apertura remoto y sin llaves"*

ha sido realizada bajo su dirección por D. **Daniel Ojeda Martin**, con N.I.F. 54.108.782-V.

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en La Laguna a 9 de junio de 2020

## Agradecimientos

En primer lugar agradecer el apoyo y empeño de mis padres para completar con éxito esta etapa universitaria.

Hacer extensivo este agradecimiento a todos mis compañeros que han compartido este camino de aprendizaje conmigo y haciendo posible este proyecto, sin olvidar a mis amigos cuyo apoyo ha sido más que académico y que no han permitido que me rinda.

Y por último a mi tutor Cándido, director de este proyecto, que junto al cotutor Iván Santos, no han escatimado en dirigirme y guiarme para hacer posible este trabajo final.

## Licencia



© Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional.

## **Resumen**

El terreno de la domótica está en constante evolución y transformación, dando nuevas herramientas y facilidades para tareas del hogar donde tienen cada vez más cabida elementos del internet de las cosas, que permiten la realización de tareas automatizadas a distancia y que por lo general se pueden centralizar en el teléfono móvil u ordenador personal. Esto permite además ceder ciertos privilegios a familiares, empleados del hogar o amigos y controlar su actividad monitorizando los tiempos de uso de los mismos administrados por el propietario. En este sentido también se aprovechan sectores como el turismo para brindar facilidades en el alojamiento a particulares o turistas, donde el principal elemento a controlar es la entrada y salida del alojamiento.

Es por ello que este proyecto se focaliza en el desarrollo de un prototipo de bajo coste capaz de conectarse a porteros, cerraduras electrónicas, puertas de garaje, etc.. para accionar la apertura a través de la red y controlar quiénes y cuándo pueden abrirla.

Para lograrlo se realizará un exhaustivo estudio de los diferentes elementos de la domótica empleados en el mercado para los sistemas de apertura sin llaves que facilitará el entendimiento de los mismos, descubrimiento de las bases de datos existentes que almacenan todo tipo de información relativa al funcionamiento de cada uno de los protocolos dependiendo de la estructura y una indagación sobre el chip ESP8266 con capacidad suficiente para cumplir las necesidades IoT que se abordarán a un coste muy competente, además de las herramientas web para la gestión de datos y usuarios como es Firebase para futuras integraciones con asistentes virtuales como Alexa.

### **Palabras clave**

Domótica, hogar, internet de las cosas, distancia, turismo, alojamiento, apertura, control, cerradura electrónica, bajo coste, asistente virtual.

## **Abstract**

The field of domotic systems is constantly evolving and transforming, giving new tools and facilities for household tasks where elements of the Internet of Things have more impact, that allow automated tasks to be performed remotely and where we can generally centralize on the mobile phone or personal computer. This also allows us to assign certain privileges to family members, household employees or friends and control their activity by monitoring their use times managed by the owner. In this sense, sectors such as tourism are also used to provide accommodation facilities to individuals or tourists, where the main element to control is the entry and exit of lodgings.

That is why this project focuses on the development of a low-cost prototype capable of connecting to doormen, electronic locks, garage doors, etc... to activate the opening through the network and control when and who can open it.

To achieve this, an exhaustive study of the different elements of home automation used in the market for keyless opening systems will be carried out, which will facilitate their understanding, discovery of the existing databases that store all kinds of information related to the operation of each one of the protocols depending on the structure and an inquiry about the ESP8266 chip with sufficient capacity to meet the IoT needs that will be addressed at a very competent cost, in addition to web tools for the management of data and users such as Firebase for future integrations with virtual assistants like Alexa.

## **Keywords**

Domotic, household, internet of things, remotely, turism, lodging, aperture, control, electronic locks, low-cost, virtual assistant.

# Índice

<b>Capítulo 1. Introducción</b>	<b>10</b>
1.1 Antecedentes	10
1.2 Objetivos	11
1.3 Estado del arte	11
1.3.1 Sistemas de apertura sin llaves	11
1.3.1 Smart Locks	13
<b>Capítulo 2. Hardware</b>	<b>15</b>
2.1 Arduino	15
2.1.1 Arduino Mega	15
2.1.2 WEMOS D1 mini Pro	17
2.2 Periféricos y Sensores	18
2.2.1 Periféricos	18
2.2.2 Sensores	19
2.3 Diseño	22
2.3.1 Instalación	24
2.3.2 Comunicación	25
<b>Capítulo 3. Software</b>	<b>26</b>
3.1 IDE Arduino	26
3.1.1 Librerías	26
3.1.2 Desarrollo en Arduino	29
3.1.3 Programación del microcontrolador	33
3.1.4 Pruebas del prototipo	33
3.2 Aplicación web	34
3.2.1 Flujo de acceso a la aplicación	34
3.2.2 Desarrollo	35
3.2.3 Modelo de datos	36
3.2.3.1 Autenticación	37
3.2.3.2 Base de datos	38
3.3 Seguridad	41
<b>Capítulo 4. Presupuesto</b>	<b>42</b>
4.1 Presupuesto del prototipo	42
4.2 Presupuesto de la plataforma	42
<b>Capítulo 5. Conclusiones y líneas futuras</b>	<b>43</b>
5.1 Conclusiones	43
5.2 Líneas futuras	44
<b>Bibliografía</b>	<b>45</b>

## Índice de figuras

Figura 1.1. Xiaomi Sherlock S2	12
Figura 1.2. Garageio	13
Figura 2.1. Arduino Mega	16
Figura 2.2. WEMOS D1 mini Pro	17
Figura 2.3. ENC28J60	19
Figura 2.4. Relé KY-019	19
Figura 2.5 Esquema eléctrico del relé	20
Figura 2.6. Infrarrojo KY-022	19
Figura 2.7. XY-MK-5V y FS1000A	20
Figura 2.8. Esquema con relé conectado	22
Figura 2.9. Imagen del prototipo	23
Figura 2.10. Esquema de carcasa para el prototipo	23
Figura 2.11. Diseño del sistema	24
Figura 2.12. Esquema de comunicación	25
Figura 3.1. Gestor de URLs ADicionales de Tarjetas	27
Figura 3.2. Soporte para ESP8266 en el IDE Arduino.	27
Figura 3.3. Gestor para la instalación de WiFiManager	29
Figura 3.4. Punto de acceso	31
Figura 3.5. Diagrama de flujo en Arduino	32
Figura 3.6. Imagen del prototipo cargado	33
Figura 3.7. Prueba de testeo en puerto serie	34
Figura 3.8. Flujo de acceso a la aplicación	35
Figura 3.9. Prototipo App Web	36
Figura 3.10. Modelo Login	37
Figura 3.11. Modelo para añadir usuario	39
Figura 3.12. Usuario con cerradura fuera de fecha	40
Figura 3.13. Estructura de datos en Firebase	40

## Índice de tablas

Tabla 1.1. Comparación Smart Locks	14
Tabla 2.1. Características de Arduino Mega	16
Tabla 2.2. Características WEMOS D1 mini Pro	17

# Capítulo 1. Introducción

## 1.1 Antecedentes

Con el reciente avance de la domótica y la automatización de tareas, el empleo de sistemas empotrados que tengan el menor impacto visual, económico, eléctrico y de complejidad es cada vez más reclamado. Los microprocesadores y microcontroladores juegan un papel muy importante ya que se usan como placas de desarrollo de hardware para construir dispositivos digitales y dispositivos interactivos que puedan detectar y controlar objetos del mundo real.

En el terreno de la domótica existe una tendencia por automatizarlo todo a través de una aplicación de móvil o página web donde poder centralizar todos aquellos elementos electrónicos y controlarlos a través del mismo dispositivo. En este caso que estudiamos, la apertura de una puerta deberá ser por medio de una llave única o en este caso una clave única. Las puertas de garaje que emplean señales de radiofrecuencia tienen cada uno su propia clave para que solo aquel que tenga el mando pueda abrirla. Pero también puede darse el caso de tener una casa de alquiler vacacional en la que solo puedan acceder a ella durante el tiempo que desean estacionarse y que una vez acabado el plazo dejen de poder entrar en la casa y pase a tener acceso a ella la siguiente persona que ha alquilado la vivienda, apartamento, hotel... O que simplemente desees que otra persona pueda acceder a abrir esa puerta donde no puedes ubicarte físicamente y le permites el acceso a través de internet y una sencilla aplicación o integración con otras como Google Assistant, Airbnb...

## **1.2 Objetivos**

Con este trabajo se pretende lograr un acceso remoto a sistemas de apertura eléctricos como puertas de garaje, porteros, vallas eléctricas, pasarelas, etc. Para ello diseñaremos un sistema empotrado capaz de abrir o cerrar una puerta o cerradura a través de internet, generando el menor impacto visual y económico posible, implementando un sistema de seguridad por usuario y clave para que solo pueda acceder aquel que disponga de ella, junto a una plataforma web que nos permita gestionar usuarios y dispositivos, así como los permisos que apliquemos a cada usuario y los tiempos permitidos para acceder a la cerradura.

## **1.3 Estado del arte**

En este apartado se analizará el estado de la tecnología actual y los diferentes productos comerciales basados en la idea de este proyecto.

### **1.3.1 Sistemas de apertura sin llaves**

Estudiaremos algunos sistemas de apertura sin llaves lanzados al mercado con diferentes funcionalidades y que sirven como referencia en el proyecto desarrollado.

- **Xiaomi Sherlock S2**

Una cerradura inteligente es un mecanismo que sirve para abrir y cerrar una puerta mediante código o pin numérico o vía bluetooth desde el móvil. Es una alternativa que sustituye a la llave en una cerradura o candado típicos. Se les llama inteligentes porque se pueden accionar, y configurar, desde un teléfono móvil por bluetooth o un mando a distancia. Dentro de este campo de cerraduras encontramos la Sherlock S2 de Xiaomi [Figura 1.1], una cerradura capaz de encajar la llave tradicional,

con la llave incrustada en el mecanismo, para poderla girar y así abrir o cerrar la cerradura desde la aplicación móvil que dispone. El propietario puede configurar la cerradura con diferentes permisos y distribuirlas de forma remota a miembros de la familia, inquilinos, etc.. Y permite ajustar su tiempo de forma flexible.



Figura 1.1. Xiaomi Sherlock S2

**Ventajas:** Su aplicación móvil y las múltiples opciones que dispone para añadir otros usuarios con tiempos y permisos diferentes.

**Desventajas:** Es un sistema que solo acepta puertas con llave.

- Garageio

Garageio es un dispositivo conectado a una app para Smartphone que permite la apertura y control automáticos de las puertas del garaje en el hogar. El dispositivo, una caja negra [Figura 1.2], se instala en las puertas del garaje , se conecta a la red wifi de la vivienda y se descarga la app Garageio o se accede a la interfaz web,

desde donde el usuario puede comprobar la actividad de la puerta del garaje o compartir el acceso con otros usuarios.



Figura 1.2. Garageio

**Ventajas:** El usuario recibe alertas si se ha marchado de la vivienda y se ha dejado la puerta del garaje abierta, historial de actividad, integración con Alexa.

**Desventajas:** Es un producto centrado en las puertas de garaje y descarta otros mecanismos de apertura eléctricos como son los porteros automáticos y no admite tiempos de uso flexibles para otros usuarios.

### 1.3.1 Smart Locks

Una cerradura inteligente (del inglés Smart Lock) es una cerradura electrónica que se activa o desactiva mediante entradas realizadas por un dispositivo autorizado. Estas entradas se realizan por medio de un protocolo de transmisión inalámbrico y una clave criptográfica. A diferencia de las cerraduras inalámbricas comunes, la cerradura inteligente también controla todos los accesos y puede establecer acciones automáticas, como por ejemplo enviar notificaciones sobre problemas a otros dispositivos.

A continuación realizaremos una comparación de las diferentes cerraduras electrónicas del mercado:

Smart Lock	Precio	Internet	App	Airbnb sync	Teclado
Remote Lock (Lockstate)	187€	si	no	si	si
Kwikset Kevo plus	210€	si	si	no	a parte
August	310€	si	si	si	no
Lockitron	179€	si	si	no	no
Schlage Connect	190€	si	si	no	si
Wink hub	69€	si	si	si	no
Yale Doorman	500€	si	si	no	si
Nuki	325€	si	si	si	si

Tabla 1.1. Comparación Smart Locks

# Capítulo 2. Hardware

## 2.1 Arduino

Arduino es una plataforma hardware de código abierto y licencia libre, basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios.

El hardware consiste en una placa con un microcontrolador Atmel AVR y puertos de entrada/salida. Los microcontroladores más usados en las plataformas Arduino son el Atmega168, Atmega328, Atmega1280, ATmega8 por su sencillez.

Arduino nos proporciona un software consistente en un entorno de desarrollo que implementa el lenguaje de programación de arduino y el bootloader ejecutado en la placa.

Ofrece las siguientes ventajas: barato, multiplataforma, entorno de programación sencillo, software libre y extensible mediante librerías en C++ y dar el salto a AVR-C, hardware libre y extensible

### 2.1.1 Arduino Mega

Arduino Mega [Figura 2.1] es la versión más grande de arduino que utiliza el procesador robusto de 8 bits AVR ATMEGA2560 con una memoria amplia que corre a 16 Mhz. Tiene 54 pines de entrada/salida, de los cuales 14 pueden ser utilizados como salidas de PWM (modulación por ancho de pulso), 16 entradas analógicas y 4 UARTs (puertos serial).

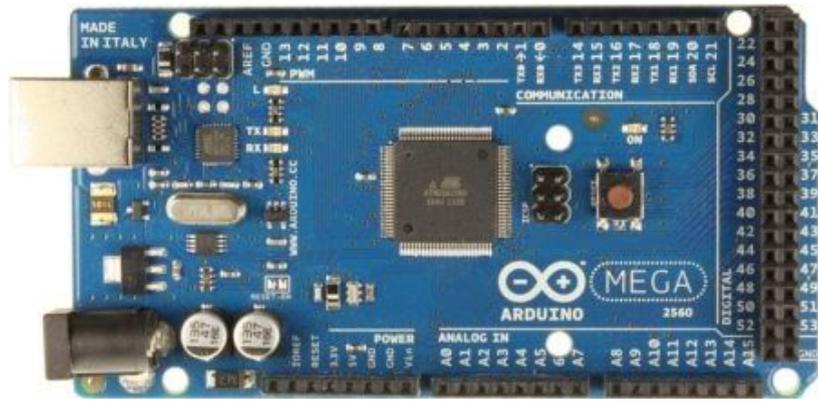


Figura 2.1 Arduino Mega

Las especificaciones de la placa Arduino Mega posee las siguientes características:

Microcontrolador	ATmega2560
Voltaje Operativo	5V
Tensión de Entrada	7-12V
Voltaje de Entrada (límites)	6-20V
Pines digitales de Entrada/Salida	54 (14 PWM)
Pines análogos de entrada	16
Corriente DC por cada Pin Entrada/Salida	40 mA
Corriente DC entregada en el Pin 3.3V	50 mA
Memoria Flash	256 KB (8KB bootloader)
SRAM	8KB
EEPROM	4KB
Clock Speed	16 MHz

Tabla 2.1 Características de Arduino Mega.

## 2.1.2 WEMOS D1 mini Pro

WEMOS D1 mini Pro [Figura 2.2] es una pequeña placa de desarrollo WIFI, basada en el ESP8266, microcontrolador inalámbrico WiFi (en 802.11 a/b/c). Este módulo incluye un conector microUSB y un interface, que permite programar el módulo directamente desde el IDE de Arduino, sin hardware adicional.

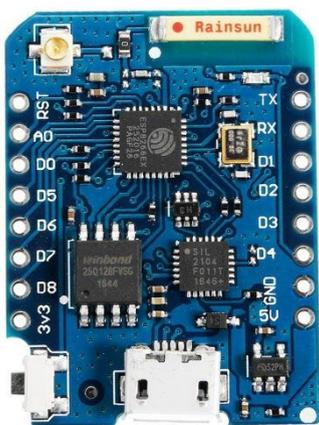


Figura 2.2. WEMOS D1 mini Pro

Las especificaciones de la placa WEMOS D1 mini Pro posee las siguientes características:

Microcontroller	ESP-8266EX
Operang Voltage	3.3V
Digital I/O Pins	11
Analog Input Pins	1 (Max input: 3.2V)
Clock Speed	80MHz/160MHz
Flash	4M bytes
Length	34.2mm
Width	25.6mm
Weight	3g

Tabla 2.2 Características WEMOS D1 mini Pro

Elegimos como microcontrolador la WEMOS D1 mini Pro ya que requerimos pocos pines, lo que reduce el tamaño, y además ésta viene con wifi incluido en la placa permitiendo así tenerla conectada a la red sin hardware adicional.

## **2.2 Periféricos y Sensores**

Para el funcionamiento óptimo de nuestro dispositivo necesitaremos periféricos que permitan la apertura a distancia. Para ello haremos posible que se active esta señal mediante la red y para una mayor robustez mediante sensores.

### **2.2.1 Periféricos**

- ENC28J60

El ENC28J60 [Figura 2.3] es un controlador de Ethernet y se controla a través de bus SPI, operando a 3.3, pero es tolerante a señales de 5V. soporta velocidades de 10Mbps/s y los modos Dúplex (Full-Duplex) y Semi-dúplex (Half-Duplex) con detección y corrección automática de la polaridad. Incorpora filtrado de paquetes para limitar el número de paquetes entrantes, un módulo DMA interno para facilitar el flujo de datos y hardware específico para el cálculo de las sumas de control (IP checksums). Sin embargo, el ENC28J60 carece de una pila de TCP/IP por hardware como sí que incluye el W5100. Por tanto, su uso es más complejo y requiere una mayor carga del procesador.



Figura 2.3. ENC28J60

Como la WEMOS D1 mini Pro ya incluye Wifi en su placa, descartamos este componente.

- Relé KY-019

El módulo de relé 5V [Figura 2.6] permite controlar una carga con corriente continua DC o corriente alterna AC desde Arduino. Desde un punto de vista de control, actúa como un interruptor controlable. Un relé está compuesto por una bobina que al circular una pequeña corriente (3.3V, 5V o 12V) genera un campo magnético y este a su vez hace mover una placa metálica abriendo o cerrando un circuito eléctrico independiente que es por el que circula un voltaje superior (220V).



Figura 2.4. Relé KY-019

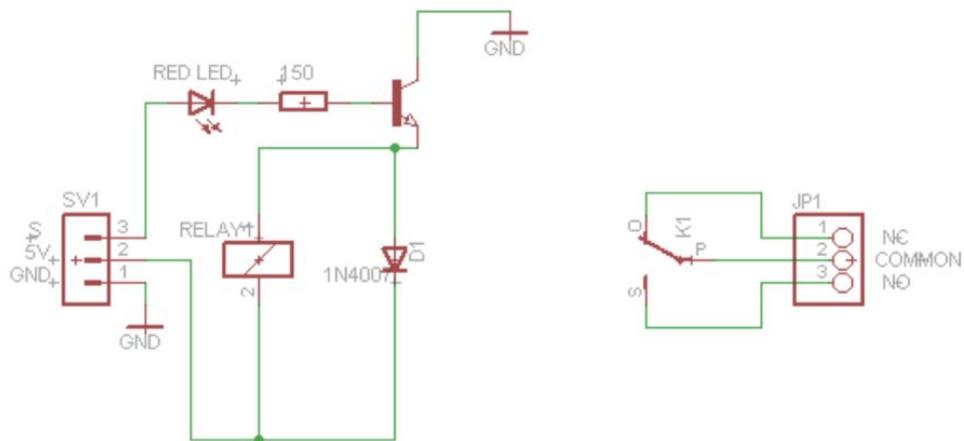


Figura 2.5. Esquema eléctrico del relé

- +: 5V (Alimentación)
- : 0V (Masa)
- S: Señal de control
- NC: Contacto cerrado
- NO: Contacto abierto
- COMMON: Contacto común

Con este módulo seremos capaces de conectarlo al circuito que acciona el mecanismo de apertura, como puertas de garaje, porteros automáticos o cerraduras inteligentes.

## 2.2.2 Sensores

- Infrarrojo KY-022

El módulo KY-022 [Figura 2.4] es un receptor de señales infrarrojas que trabajan en 38kHz. Utiliza la librería IRremote para recibir y procesar las señales infrarrojas en nuestro programa sobre el Arduino.



Figura 2.6. Infrarrojo KY-022

Con este componente podremos accionar el mecanismo del arduino de manera analógica sin tener que acceder a internet mediante un emisor de infrarrojos.

- Radiofrecuencia RF 433MHZ

Los módulos de radio frecuencia RF 433MHz son transmisores/receptores inalámbricos y vienen separados el receptor XY-MK-5V y el emisor FS1000A [Figura 2.5]. La frecuencia de operación es de 433MHz y el alcance depende del voltaje con el que alimentemos el módulo y la antena que usemos. A 5V y con la antena del módulo, el alcance difícilmente excederá de los 2 metros. Alimentando a 12V y con una antena de cobre de 16.5cm el rango en exteriores puede alcanzar 300 metros.

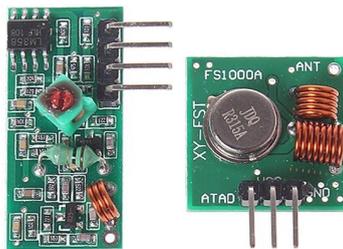


Figura 2.7. XY-MK-5V y FS1000A

Utilizando este módulo, al igual que el receptor de infrarrojos, podremos accionar el mecanismo de apertura sin internet y con uso de un emisor de radiofrecuencia.

## 2.3 Diseño

Finalmente con respecto al diseño hardware, como hemos dicho anteriormente, haremos uso de la placa WEMOS D1 mini y un Relé como elementos básicos y esenciales para el funcionamiento correcto del prototipo.

Como podemos ver en la [Figura 2.8], el relé requiere de dos pines para la masa (-) y alimentación (+) que se conectan respectivamente a los pines de la placa GND y 5V. Luego el pin de control (s) lo conectaremos al pin D5 dejando por completa las conexiones necesarias para su funcionamiento.

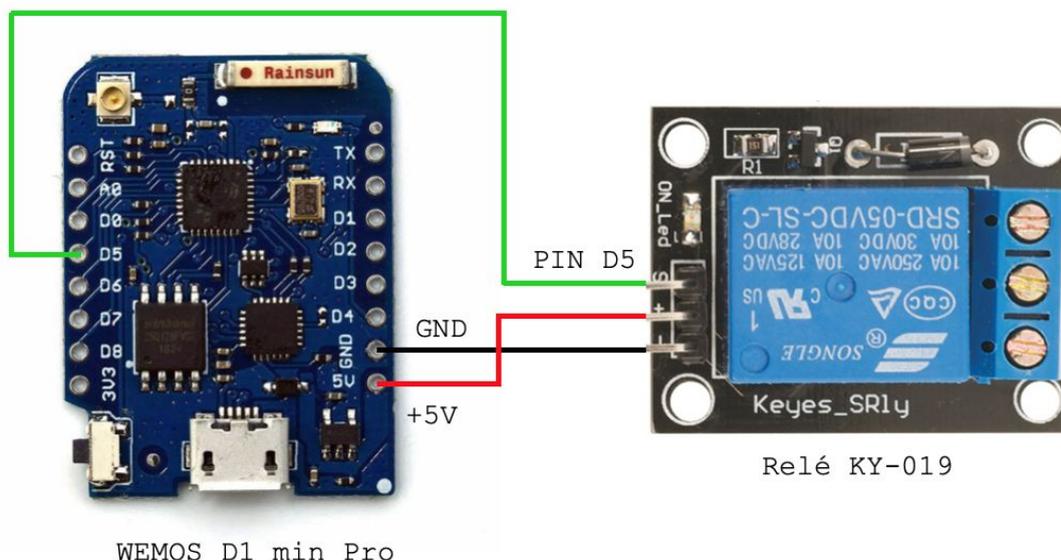


Figura 2.8. Esquema con relé conectado.

Para la implantación de los distintos componentes, se ha utilizado una pequeña protoboard que permite hacer pruebas con el hardware del prototipo [Figura 2.9].

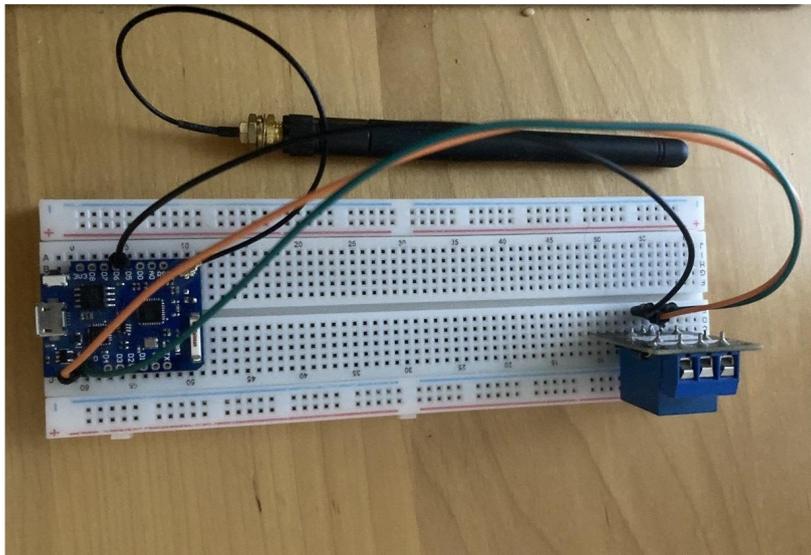


Figura 2.9. Imagen del prototipo

Para la protección de los diferentes componentes diseñamos una pequeña caja [Figura 2.10] donde colocar todos los componentes conectados con acceso al usb para darle corriente, acceso a las entradas del cableado para el relé, la salida de la conexión para la antena y dos leds de control.

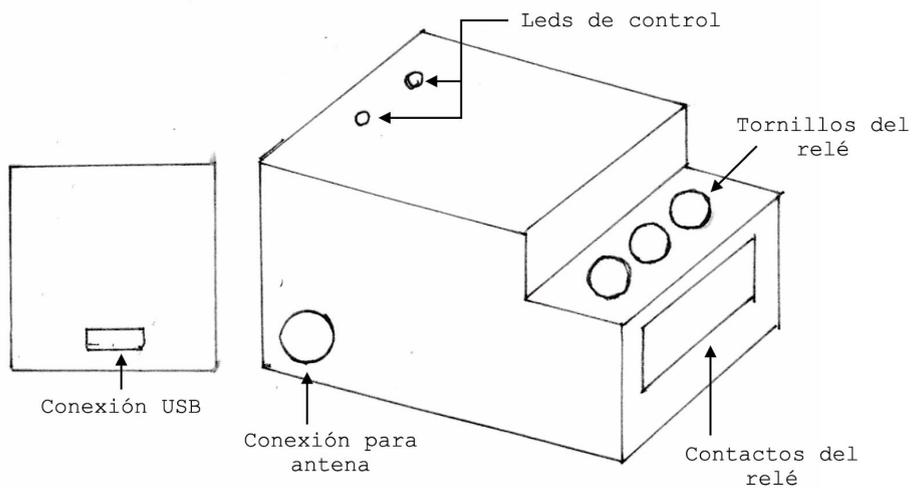


Figura 2.10. Esquema de carcasa para el prototipo

En la siguiente figura [Figura 2.11] podemos ver cómo interactúan los diferentes elementos que requiere el prototipo para que cumpla su función a través de internet y llegar a las diferentes plataformas web y poder gestionar las diferentes funcionalidades de la misma.

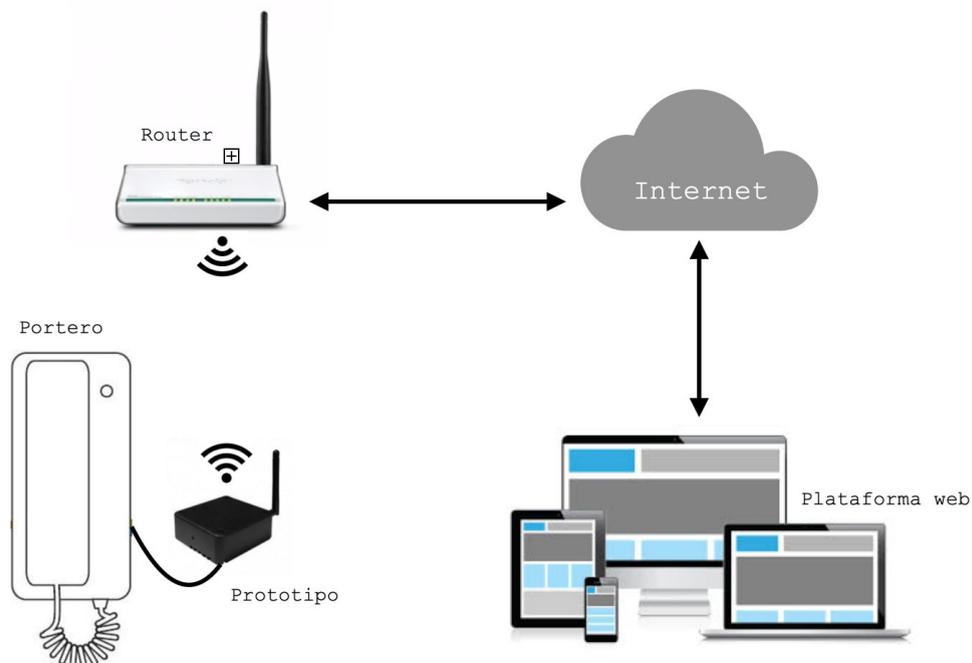


Figura 2.11. Diseño del sistema

### 2.3.1 Instalación

La instalación del producto final consistirá en varios pasos que tendremos que seguir para su correcto funcionamiento:

1. Colocar el dispositivo cerca del mecanismo de apertura electrónico (portero, puerta de garage..)
2. Conectar los cables encargados de accionar la apertura al relé.
3. Conectar a la corriente mediante el conector micro USB.
4. Una vez encendido, conectar el dispositivo a la red mediante un punto de acceso creado por el dispositivo.
5. Entrar en la plataforma web, registrar el producto con su id y comenzar a utilizarlo.

### 2.3.2 Comunicación

En esta figura [Figura 2.12] se muestra un esquema general del sistema con diferenciación del diseño electrónico y diseño de datos y web.

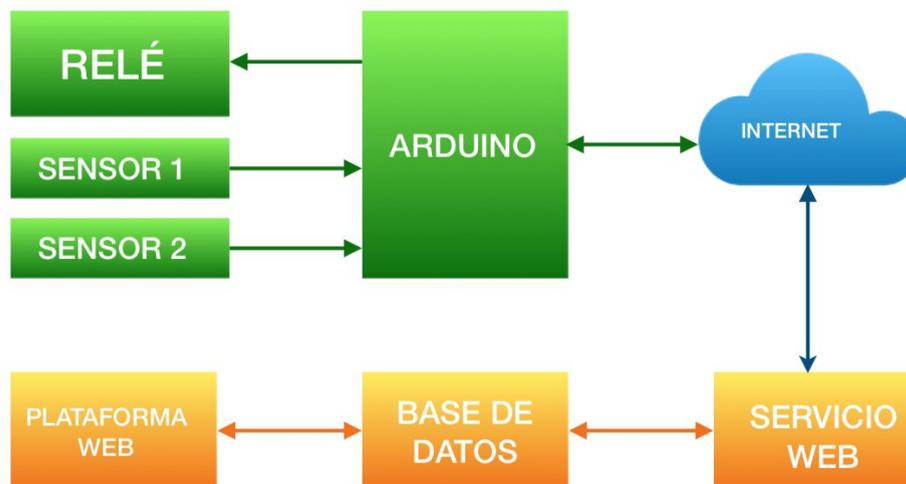


Figura 2.12. Esquema de comunicación

## Capítulo 3. Software

En este apartado incluimos la descripción del software empleado en el desarrollo del presente trabajo de fin de grado. Por un lado requerimos programar la placa arduino para su correcto funcionamiento a través de internet y de los periféricos que lleva conectados, y por otro una plataforma web de gestión de usuarios y cerraduras.

### 3.1 IDE Arduino

Para la programación de Arduino se ha utilizado el Arduino IDE 1.8.5, que es un entorno de desarrollo en el que se realiza la programación de estas placas basado en C++, ofreciendo unas librerías que facilitan la programación de los pines de entrada/salida y de los puertos de comunicación. Nuestro objetivo será el de desarrollar un código capaz de conectar la placa a la red y accionar el mecanismo del relé a través de la misma.

#### 3.1.1 Librerías

##### a) ESP8266

Wemos son una serie de placas de prototipado con chip ESP8266 integrado para conexión WiFi. El ESP8266 es un chip Wi-Fi de bajo coste con pila TCP/IP completa y capacidad de MCU (Micro Controller Unit). Este pequeño módulo permite a los microcontroladores conectarse a una red Wi-Fi y realizar conexiones TCP/IP sencillas utilizando comandos de tipo Hayes. El ESP8266 dispone internamente de un pequeño

procesador que prácticamente es capaz de replicar casi cualquier cosa que los Arduinos puedan hacer.

Para poder programar la placa WEMOS D1 mini a través del IDE Arduino tenemos primero que instalar el driver del conversor USB a serie, que es el CH340G, para programarla mediante USB y las librerías para usar el chip ESP8266.

Podemos instalar el soporte a terceros en nuestro IDE [Figura 3.1] simplemente añadiendo el texto "http://arduino.esp8266.com/stable/package\_esp8266com\_index.json" en preferencias:



Figura 3.1. Gestor de URLs Adicionales de Tarjetas

Y luego desde el gestor de tarjetas [Figura 3.2] dar a instalar el soporte para ESP8266.



Figura 3.2. Soporte para ESP8266 en el IDE Arduino.

Una vez instalados los requerimientos de la placa WEMOS D1 mini para el IDE de Arduino podemos pasar a programarla.

## b) SPI

SPI (Serial Peripheral Interface, Interface para Periféricos Serie) es un protocolo de transferencia de datos serie sincronizados, utilizado por microcontroladores para comunicaciones rápidas con uno o más dispositivos. También puede usarse para comunicar dos microcontroladoras entre sí. En nuestro caso nos hará falta para la comunicación con los diferentes periféricos que utilicemos en el prototipo.

## c) WiFiManager

La librería WiFiManager es una librería de código abierto que permite configurar los parámetros de una red WiFi desde un móvil en un ESP8266, de forma rápida sin cargar un nuevo código, simplemente rellenando un formulario en una página web.

Anteriormente ajustamos los parámetros de la red wifi manualmente en el código para luego cargarlo en la placa. Esto supone un problema ya que si la red wifi cambia el dispositivo no será capaz de conectarse y tendremos que modificar el código para introducir las credenciales nuevamente. Aquí es donde entra la librería WiFiManager para que en caso de que no pueda conectar a la red WiFi que tiene configurada por defecto, crea un access point (punto de acceso) y active un servidor DNS y un servidor web. Entonces, desde cualquier dispositivo con WiFi (ordenador, portátil, teléfono o tablet), aparecerá en las redes WiFi un nuevo punto de acceso al que una vez conectado aparecerá un portal cautivo que nos permite configurar la red WiFi del dispositivo sin tener que cargar de nuevo el programa.

Para hacer uso de esta librería tendremos que instalarla a través del menú del IDE Arduino en Programa>Incluir Librerías>Administrar Bibliotecas... [Figura 3.3]

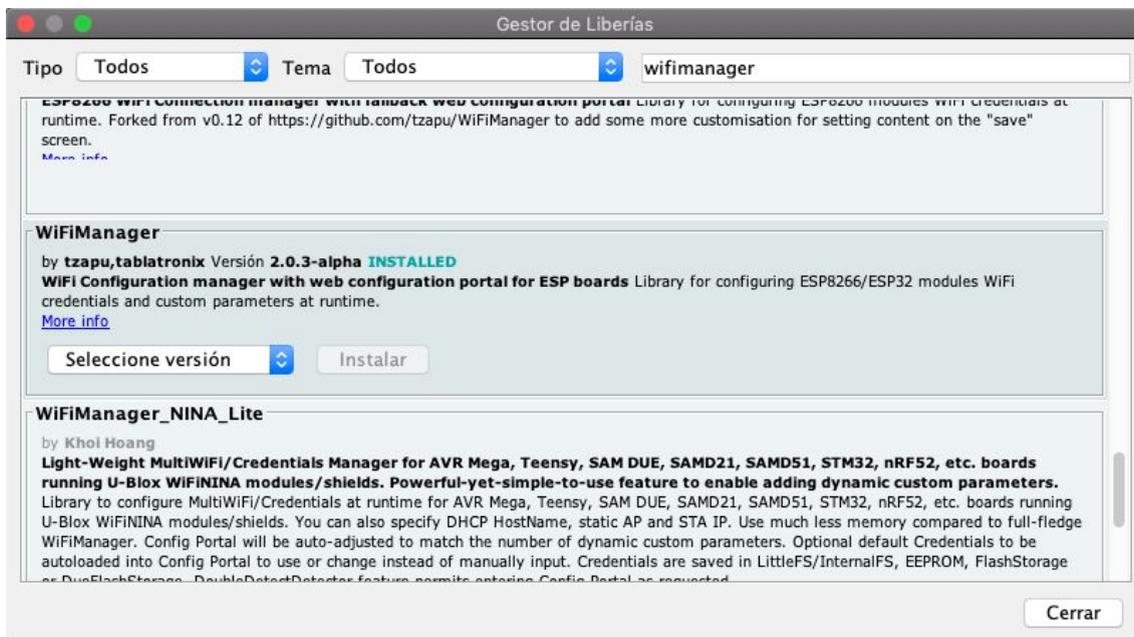


Figura 3.3. Gestor para la instalación de WiFiManager

### 3.1.2 Desarrollo en Arduino

Una vez armado el sistema y con las dependencias de librerías instaladas, se necesita desarrollar el código que va a ser programado en el microcontrolador. En primer lugar se han declarado las variables globales para posteriormente rellenar la función `void setup()`, que inicializa el sistema, y `void loop()`, que es la función principal.

a) `void setup()`:

En la función `setup` configuraremos la conexión a la red WiFi, el servidor que se lanzara sobre la placa y los pines de salida para accionar el relé, que en este caso estableceremos en el pin D5, y el LED de la placa.

```

void setup() {

    Serial.begin(9600);
    pinMode(D5, OUTPUT);
    pinMode(BUILTIN_LED, OUTPUT);
    digitalWrite(BUILTIN_LED, HIGH);

    WiFiManager wifiManager;
    // Descomentar para resetear configuración
    // wifiManager.resetSettings();
    wifiManager.setSTAStaticIPConfig(_ip,_gw,_sn);
    wifiManager.autoConnect("ApertureDevice");

    digitalWrite(BUILTIN_LED, LOW);
    Serial.printf("Connected\n");
    server.on("/", HTTP_POST, rele);
    server.begin();
    printWifiStatus();
}

```

Primeramente inicializa el serial a 9600 baudios para poder monitorizar el estado de la conexión a la red por el monitor serie. Seguidamente inicializa el pin del relé situado en el PIN D5 como OUTPUT para accionar el mecanismo y el pin del LED incluido en la placa para monitorizar y comprobar el estado de la red WiFi, que primeramente lo apagaremos para que una vez conectado a la red se encienda. Luego inicia WiFiManager, estableciendo como parámetros estáticos la dirección IP que va a tener el dispositivo. Si no se ha conectado a una red WiFi, WiFiManager crea un punto de acceso para acceder a la configuración de red del dispositivo y escanear las redes a las que tiene acceso y conectarla [Figura 3.4], permitiendo además establecer la IP del dispositivo.

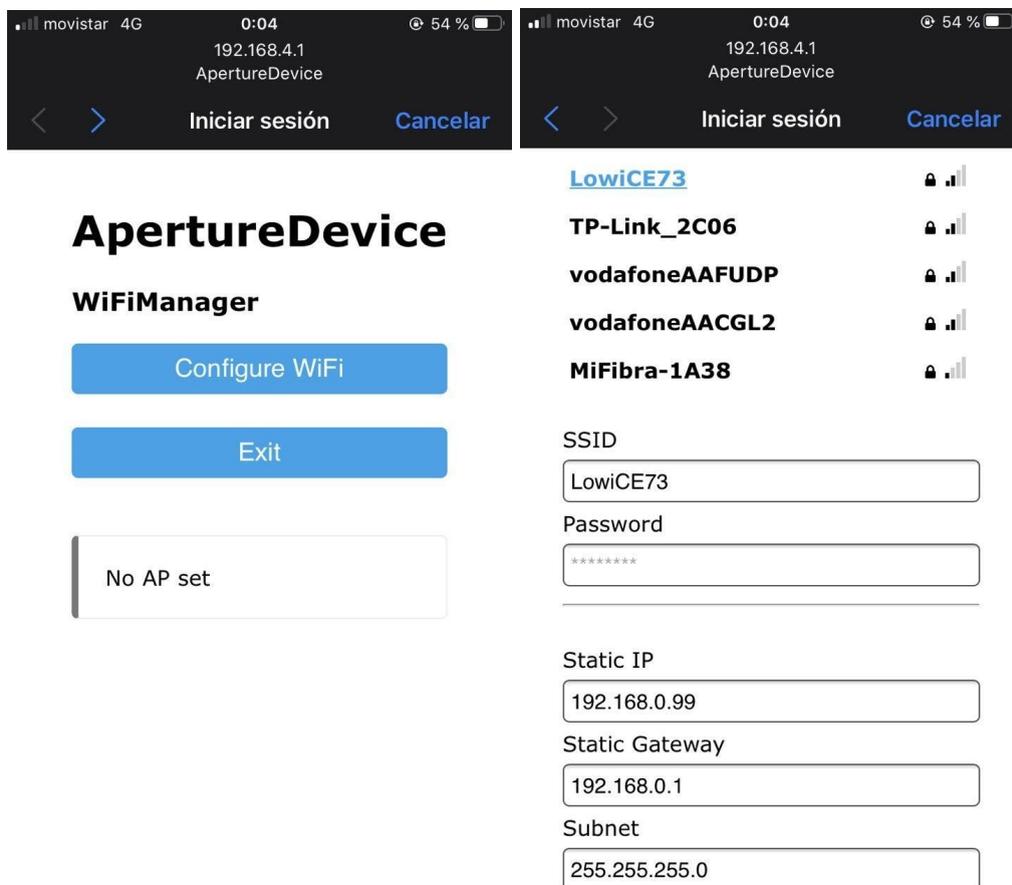


Figura 3.4. Punto de acceso

Una vez conectado guarda la configuración para conectarse automáticamente la próxima vez que se encienda el dispositivo y lanzará el servidor establecido en el puerto 80, habilitando la escucha para peticiones POST que gestiona la función relé:

```
void rele(){
    if (server.arg("rele") == "ON"){
        digitalWrite(D5, HIGH);
        delay(1500);
        digitalWrite(D5, LOW);
    }
    server.sendHeader("Location","/");
    server.send(303);
}
```

Esta pequeña función escucha los argumentos que envían las peticiones POST y dependiendo del valor (que en nuestro caso solo acepta "ON") del parámetro "rele", encenderá el relé durante un breve periodo de tiempo, lo suficiente como para que se active el mecanismo de apertura.

b) void loop():

Para manejar las solicitudes HTTP entrantes, necesita llamar al método *handleClient* en el servidor, dentro de la función principal *void loop()*.

```
void loop() {
  server.handleClient();
}
```

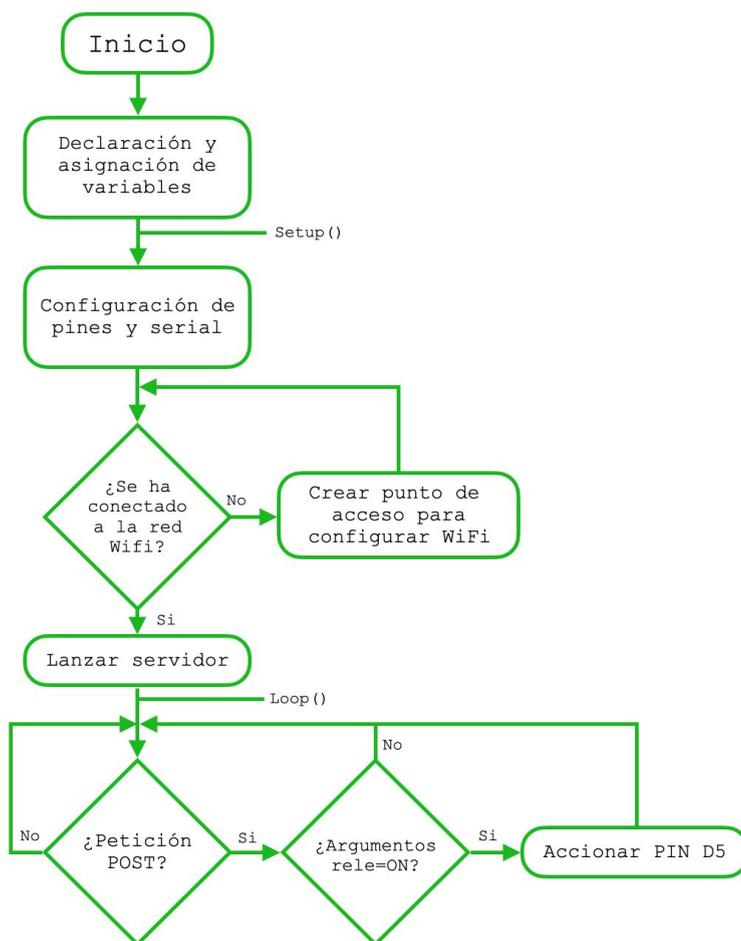


Figura 3.5. Diagrama de flujo en el desarrollo del Arduino

### 3.1.3 Programación del microcontrolador

Una vez desarrollado el firmware, se necesita programar el microcontrolador, que consiste en compilar el código del firmware y grabarlo en la memoria del microcontrolador para que sea ejecutado.

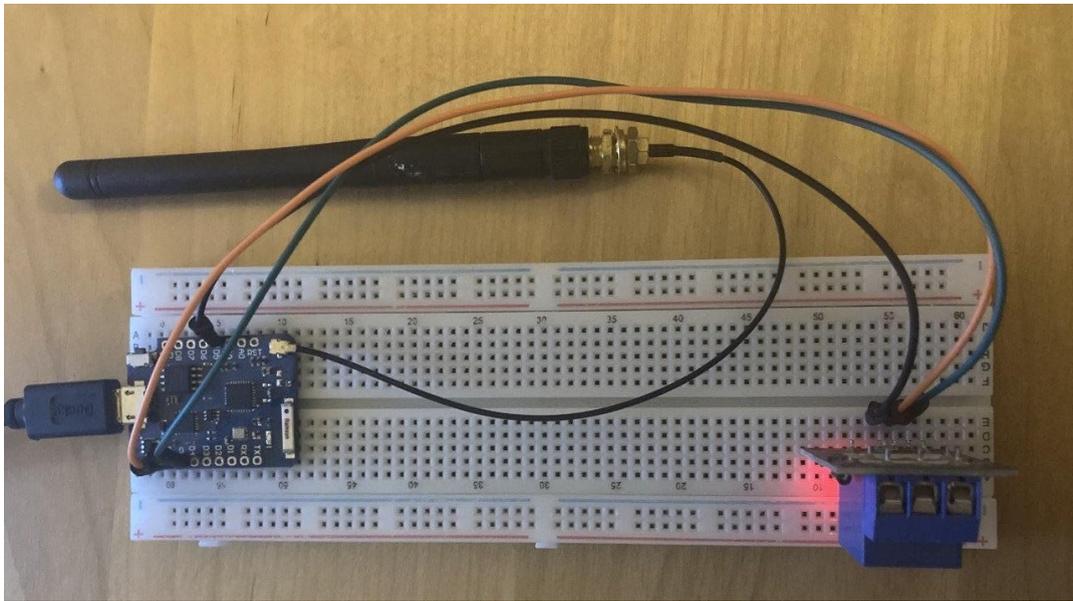
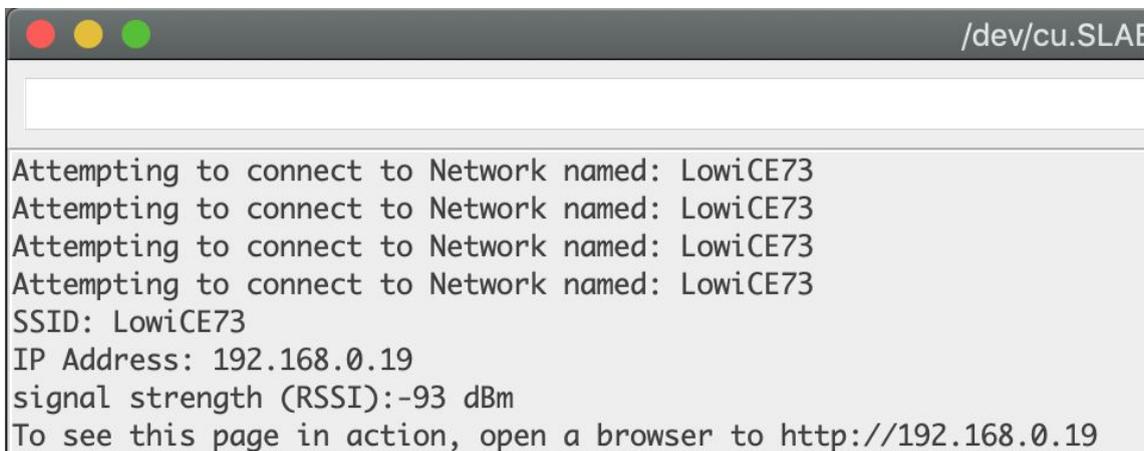


Figura 3.6. Imagen del prototipo cargado

### 3.1.4 Pruebas del prototipo

Una vez programado el microcontrolador, se configura el dispositivo para acceder a la red WiFi. A continuación se abre un puerto serie configurado con 9600 baudios para observar si los datos y la conexión funciona correctamente.

La siguiente imagen [Figura 3.4], muestra la salida que imprime en el puerto serie a modo de testeo.



```

/dev/cu.SLAB
Attempting to connect to Network named: LowiCE73
SSID: LowiCE73
IP Address: 192.168.0.19
signal strength (RSSI):-93 dBm
To see this page in action, open a browser to http://192.168.0.19

```

Figura 3.7. Prueba de testeo en puerto serie

Tras conectarse a la red WiFi podremos hacer un POST a través de nuestra plataforma web, creando un enlace que envíe los datos a través de la dirección IP del dispositivo terminado en `/?rele=ON`.

## 3.2 Aplicación web

Para accionar el mecanismo de apertura del arduino elaboramos una aplicación web que permita gestionar usuarios y cerraduras a través de la red. La aplicación se ha desarrollado en HTML5, CSS3 y JavaScript, siguiendo un modelo vista-controlador. Para la gestión de datos y usuarios haremos uso de la plataforma Firebase, que mediante scripts en Javascript nos permite recibir y emitir datos entre la web y Firebase.

### 3.2.1 Flujo de acceso a la aplicación

A continuación se muestra un esquema [Figura 3.6] de flujo de acceso a la aplicación web:

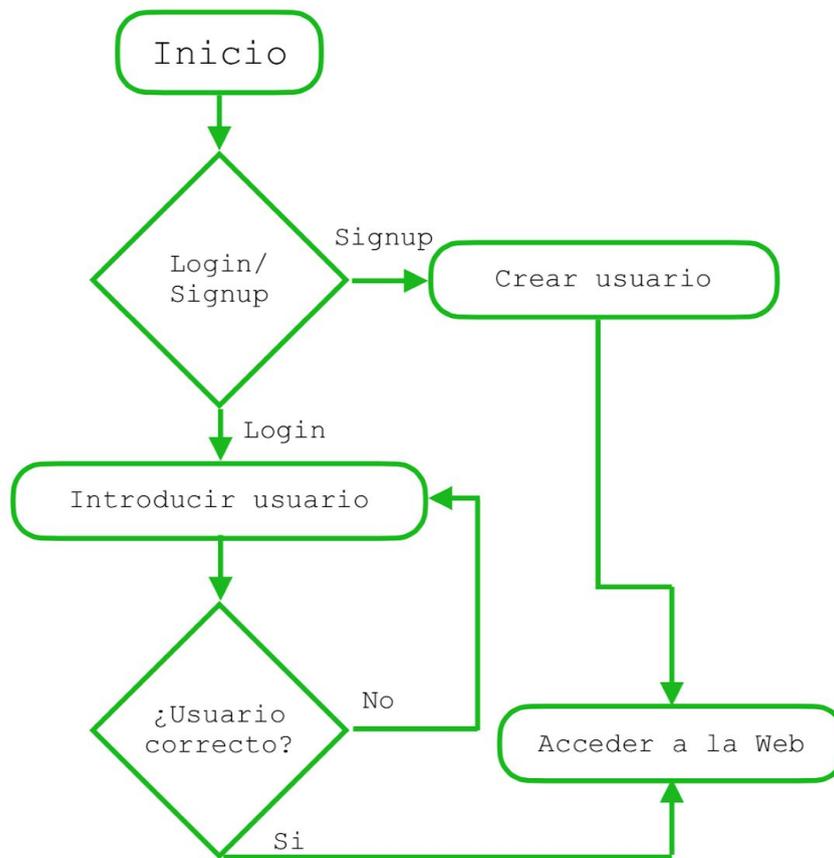


Figura 3.8. Flujo de acceso a la aplicación

### 3.2.2 Desarrollo

La función de la página web es la de identificar al usuario para poder añadir la/las cerraduras así como accionar el mecanismo de apertura [Figura 3.7]. Los administradores de cada cerradura pueden añadir o eliminar usuarios con fechas de entrada y salida, los cuales podrán accionar el mecanismo de dicha cerradura dentro de las fechas establecidas.

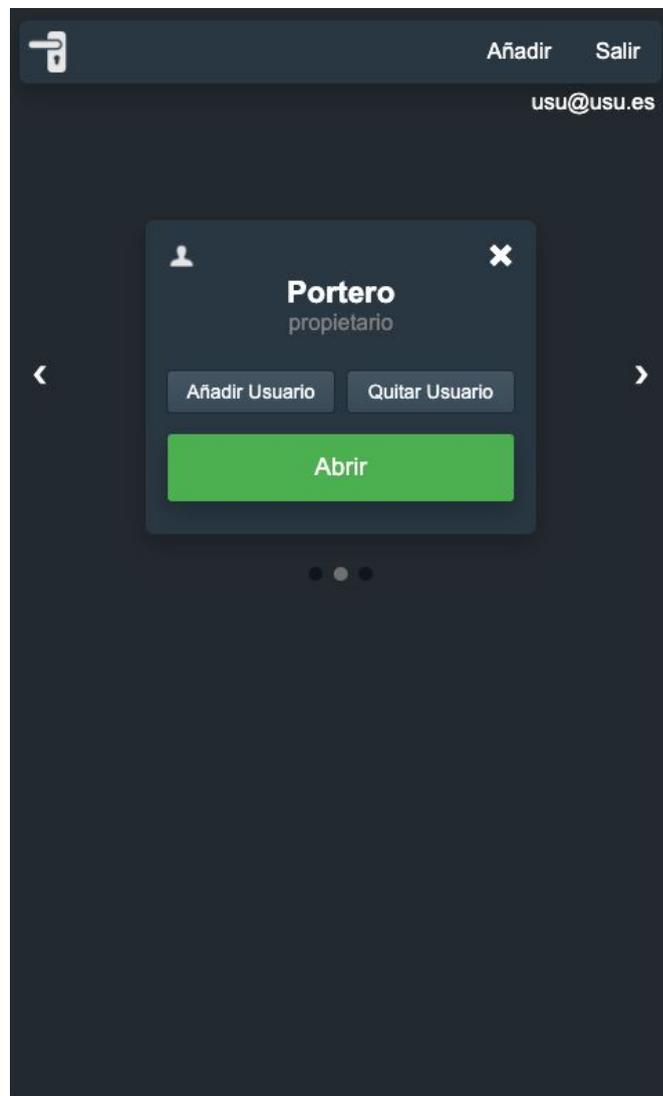


Figura 3.9. Prototipo App Web

### 3.2.3 Modelo de datos

En cuanto a la administración de usuarios y cerraduras empleamos Firebase, un servicio web que proporciona un backend en la nube con una fuente de datos NoSQL en tiempo real y librerías para poder acceder a la base de datos desde aplicaciones Web, IOS o Android. Firebase también facilita la autenticación con el propio sistema o mediante conectores como pueden ser Facebook, Twitter, Mail, etc.

### 3.2.3.1 Autenticación

Para la gestión de cuentas de usuario y la autenticación de las mismas, utilizamos Firebase Authentication, que nos permitirá crear el acceso con dirección de correo electrónico y contraseña.

Antes que nada, crearemos un formulario que permita a los usuarios nuevos registrarse en la app mediante su dirección de correo electrónico y una contraseña [Figura 2.8]. Cuando un usuario complete el formulario, valida la dirección de correo electrónico y la contraseña que proporcionó el usuario para después pasarlos al método `createUserWithEmailAndPassword` (Firebase, 2020).

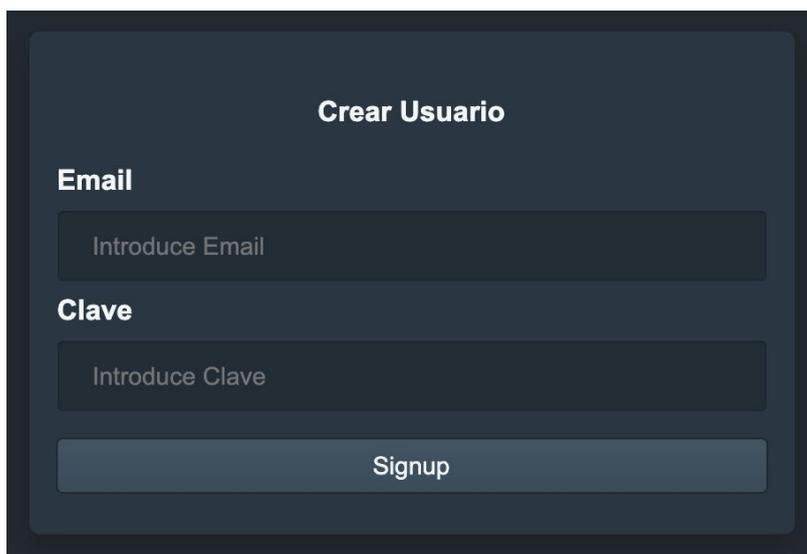
El formulario tiene un fondo oscuro con un título centralizado "Crear Usuario". Debajo del título, hay un campo de entrada etiquetado "Email" con el texto "Introduce Email" dentro. A continuación, hay otro campo de entrada etiquetado "Clave" con el texto "Introduce Clave" dentro. En la parte inferior del formulario, hay un botón rectangular con el texto "Signup".

Figura 3.10. Modelo LogIn

Luego añadimos otro formulario que permita a los usuarios existentes acceder con su dirección de correo electrónico y una contraseña. Cuando un usuario complete el formulario, llama al método `signInWithEmailAndPassword`.

Por último añadimos un método que permita salir a nuestro usuario de la plataforma con el método `signOut`.

Con estos tres métodos tendremos completa las acciones relacionadas con la autenticación de los usuarios a la aplicación web y por lo tanto a la base de datos de la aplicación en Firebase.

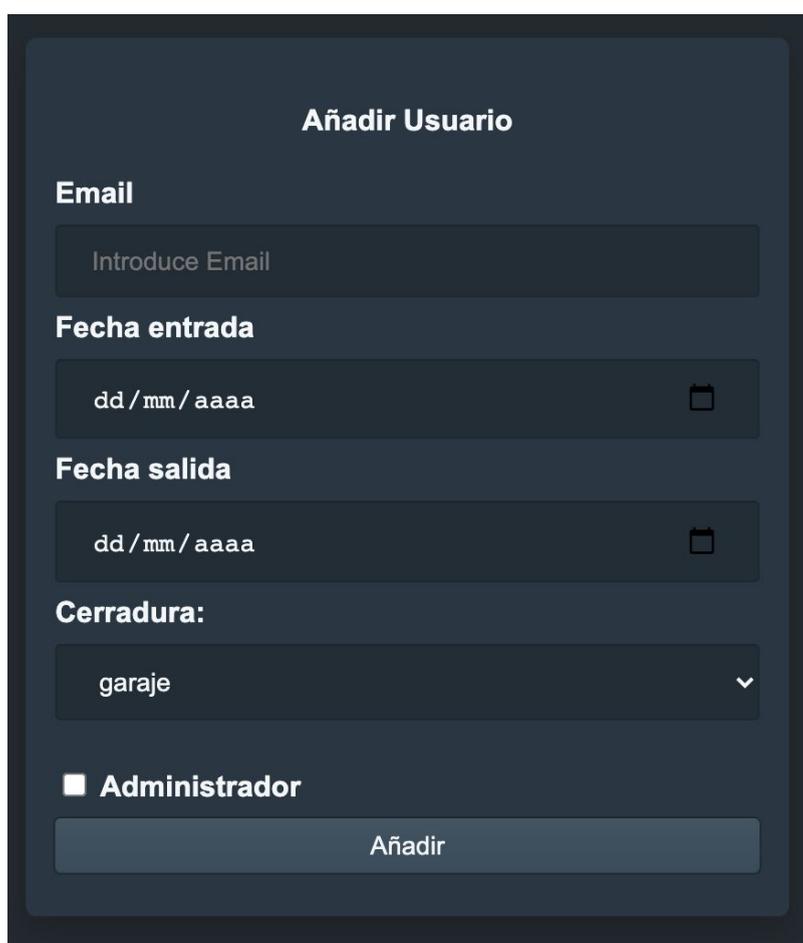
### 3.2.3.2 Base de datos

La estructura de la base de datos se fundamenta en tres tablas:

- Tabla autenticación: almacena el correo y contraseña de cada usuario para la autenticación, además de la fecha de creación del usuario, fecha del último inicio de sesión y el UID (clave que identifica únicamente a cada usuario).
- Tabla usuarios: esta tabla contiene los datos de cada usuario así como las cerraduras vinculadas a ella. El UID de esta tabla coincide con el de la tabla de autenticación y además contiene el email. Por último tiene una referencia a cada una de sus cerraduras que administra y referencias a las cerraduras con acceso limitado con fecha de entrada y salida.
- Tabla cerraduras: contiene cada una de las cerraduras registradas en la aplicación y almacena su nombre e identificador que nos permita conectarnos a ella, además del propietario de la misma.

Cada vez que la plataforma registre un usuario éste quedará almacenado en la tabla autenticación (email y contraseña) generando un identificador único, y en la tabla usuarios guardará con el mismo identificador su email. Una vez identificado en la plataforma podrá añadir una cerradura que se almacenará en la tabla de cerraduras guardando el nombre de la misma, su id y el administrador (usuario que la registra), mientras que en la tabla del usuario se añadirá el nombre de la tabla a un vector que almacena todas las cerraduras a las que tiene acceso y una nueva

entrada que lleve también el nombre de la cerradura donde guardará que es administrador de la cerradura, ya que es el usuario el que está registrando el dispositivo en la plataforma. Cuando se añada un usuario para que acceda a una de las cerraduras que administre (no podrá añadir usuarios a aquellas cerraduras que no es administrador), se modificará la tabla del usuario en cuestión, añadiendo un nuevo elemento al vector de cerraduras y una nueva entrada con el nombre de la cerradura donde se guardarán tres valores: la fecha de entrada, la fecha de salida y el valor false a la variable de administrador.



**Añadir Usuario**

**Email**

Introduce Email

**Fecha entrada**

dd/mm/aaaa

**Fecha salida**

dd/mm/aaaa

**Cerradura:**

garaje

**Administrador**

Añadir

Figura 3.11. Modelo para añadir usuario.

La plataforma también eliminará, una vez identificado el usuario en la web, aquellas cerraduras que no administra y que tienen fecha caducada, mientras que las que todavía no ha llegado la fecha para acceder aparecerán en la plataforma pero sin posibilidad de abrirla.



Figura 3.12. Usuario con cerradura fuera de fecha.

Se incluye además la posibilidad de que el usuario elimine las cerraduras, eliminando sus respectivas entradas en la tabla del usuario e impidiendo así que vuelva a tener acceso a ellas. El administrador podrá eliminar los usuarios de las cerraduras que administre.

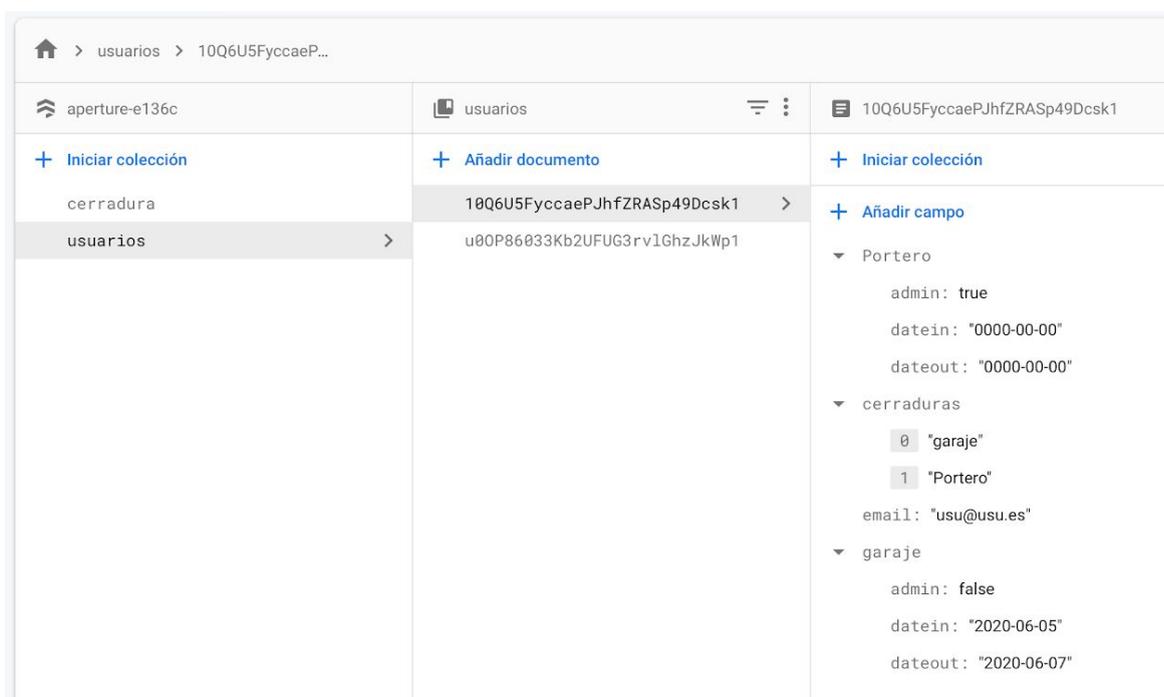


Figura 3.13. Estructura de datos en Firebase.

### **3.3 Seguridad**

En cuanto a la seguridad estudiamos diferentes aspectos de nuestro proyecto para hacerlo seguro y robusto ante ataques. Para proteger la página web podemos habilitar el protocolo HTTPS, un protocolo de comunicación de Internet que protege la integridad y la confidencialidad de los datos de los usuarios entre sus ordenadores y el sitio web.

Firebase proporciona reglas de seguridad para sus datos que proporcionan una protección sólida y completamente personalizable. En este sentido se establecen reglas para que solo usuarios propietarios de los contenidos almacenados en la base de datos puedan acceder a él.

Para el registro de usuarios podemos implementar una verificación del correo con el que se registra para verificar que el usuario tiene acceso a la dirección de email.

## Capítulo 4. Presupuesto

### 4.1 Presupuesto del prototipo

El siguiente presupuesto recoge los costes para producir una unidad del prototipo sin sensores adicionales, aunque opcionalmente se le pueden añadir tantos sensores como requiera el sistema de apertura eléctrico a controlar.

Componente	Cantidad	Precio	Total
Wemos D1 mini Pro	1	3,30	3,30
Relé KY-019	1	1,45	1,45
Carcasa	1	3,50	3,50
Mano de obra	2	5	10
<b>TOTAL</b>			<b>18,25</b>

### 4.2 Presupuesto de la plataforma

El siguiente presupuesto está dividido en un coste de desarrollo y coste anual de hosting y dominio.

Componente	Cantidad	Precio	Total
Desarrollo de la plataforma web	90	10	900
Servicio de dominio + hosting anual	1	300	300

# Capítulo 5. Conclusiones y líneas futuras

## 5.1 Conclusiones

En el desarrollo de este proyecto se han estudiado las tecnologías relacionadas con el objetivo principal definido en el primer capítulo del documento. Estas tecnologías están estrechamente relacionadas con la domótica ya que automatizan tareas del hogar y que debido al avance diario encontramos cada vez nuevos elementos a la vez que se renuevan con nuevas funcionalidades, cada vez más simples, económicos y de fácil uso e instalación.

Gracias a los microcontroladores como Arduino, podemos abstraernos de la microelectrónica y la programación de bajo nivel, que sumado al internet de las cosas, facilitan la creación de elementos conectados a la red de manera rápida y sin un elevado conocimiento de las materias desarrolladas.

Teniendo en cuenta los objetivos del proyecto y viendo los resultados en cuanto a simplicidad, bajo costo y facilidad de uso, podemos determinar que se han cumplido con éxito los objetivos principales con un prototipo funcional como resultado final.

## 5.2 Líneas futuras

Una vez finalizado el proyecto e implementación y pruebas del prototipo, podemos ver el potencial de este proyecto, así como su comercialización y posibles mejoras de diseño para que ocupe el menor espacio posible y sea fácil de instalar.

También podemos destacar que la posibilidad de integrar con otras plataformas como Google Assistant es muy viable ya que hacemos uso de Firebase que pertenece a Google, u otras integraciones a plataformas que puedan verse beneficiadas de este producto como son las compañías de alquiler de viviendas vacacionales como Airbnb.

Como elemento principal a desarrollar en un futuro es la implantación de un DNS que nos permita acceder al dispositivo desde internet y con ello accionar el mecanismo desde cualquier lugar, así como todas las medidas pertinentes para que el dispositivo sea seguro ante ataques tanto a nivel físico como software.

# Bibliografía

1. Arduino Language Reference.  
<http://arduino.cc/en/Reference/HomePage> (consultado el 10/04/2019)
2. WEMOS D1 mini Pro documentation.  
[https://docs.wemos.cc/en/latest/d1/d1\\_mini\\_pro.html](https://docs.wemos.cc/en/latest/d1/d1_mini_pro.html)  
(consultado el 28/04/2019)
3. Firebase. <https://firebase.google.com/docs> (consultado el 08/06/2019)
4. ESP8266 Arduino Core's documentation.  
<https://arduino-esp8266.readthedocs.io/en/2.7.1/>  
(consultado el 25/05/2019)
5. W3Schools <https://www.w3schools.com/>
6. Proteger sitios web con el protocolo HTTPS  
<https://support.google.com/webmasters/answer/6073543?hl=es> (consultado el 12/05/2020)
7. Luis del Valle Hernández. Relé y Arduino MKR1000, controla la luz de tu casa. Programar Fácil.  
<https://programarfácil.com/blog/arduino-blog/rele-y-arduino-mkr1000/> (consultado el 25/05/2019)
8. ESP8266 WiFi Configuration Library  
<https://tzapu.com/esp8266-wifi-connection-manager-library-arduino-ide/> (consultado el 13/05/2020)
9. Luis del Valle Hernández. WiFiManager librería para configurar red WiFi de un ESP8266. Programar Fácil.  
<https://programarfácil.com/esp8266/wifimanager-configura-wifi-esp8266/> (consultado el 13/05/2020)