



Universitat Oberta
de Catalunya

iAirZoning

Climatización Inteligente por Zonas
controlada mediante Apple HomeKit

José Antonio Jiménez Campos

Grado de Ingeniería Informática
Sistemas Empotrados

Jordi Bécares Ferrés

Pere Tuset Peiró

Junio de 2020



Esta obra está sujeta a una licencia de Reconocimiento - NoComercial - SinObraDerivada 3.0 España de Creative Commons. (https://creativecommons.org/licenses/by-nc-nd/3.0/es/deed.es_ES)

FICHA DEL TRABAJO FINAL

Título del trabajo:	<i>iAirZoning</i>
Nombre del autor:	<i>José Antonio Jiménez Campos</i>
Nombre del consultor/a:	<i>Jordi Bécares Ferrés</i>
Nombre del PRA:	<i>Pere Tuset Peiró</i>
Fecha de entrega:	<i>06/2020</i>
Titulación:	<i>Grado de Ingeniería Informática</i>
Área del Trabajo Final:	<i>Sistemas Empotrados</i>
Idioma del trabajo:	<i>Castellano</i>
Palabras clave:	<i>Domótica, Climatización, HomeKit</i>
Resumen:	
<p>Los sistema de climatización por conductos suponen un alto coste energético, puesto que estos actúan sobre toda una planta de la vivienda o, en algunos casos, en la totalidad de esta. Sin embargo, lo normal es utilizar únicamente una parte de la vivienda, y lo ideal sería que el sistema de climatización pudiera actuar solamente en esa parte.</p> <p>Este trabajo cubre esa necesidad, usando compuertas motorizadas para regular el flujo de aire que llega a cada estancia (zonificación), y controlando la máquina central de climatización. Además, toda la gestión se realiza utilizando el sistema domótico HomeKit desarrollado por Apple Inc., lo que permite poder interactuar con el sistema a través de cualquier dispositivo cliente HomeKit (iPhone, iPad, HomePod, Mac, AppleWatch...), y hacer uso de las integraciones que ofrece esta plataforma.</p> <p>El sistema desarrollado es capaz de mantener, de forma autónoma, y con un alto grado de confort para el usuario, la temperatura establecida en cada zona o estancia, informado al entorno HomeKit de los cambios que se van produciendo (temperatura, modo de trabajo, etc.).</p> <p>La instalación se ha realizado en la planta alta de una vivienda, con 4 estancias, y los resultados han sido muy satisfactorios, puesto que el nuevo sistema ha conseguido reducir el coste energético de forma considerable. Además, la integración con HomeKit le da un valor añadido gracias a la posibilidad de interactuar con el resto de accesorios de esta plataforma.</p>	

Abstract:

HAVC systems have a high energy cost, because they work over an entire floor of a house or, in some cases, over all house. However, it is normal to use only a part of the house, and ideally, HAVC could only work over that zone.

This work covers that need, using motorized gates to regulate air flow that reaches to each room (zoning), and controlling main HAVC system. In addition, management is done using HomeKit home automation system developed by Apple Inc., which allows to interact with the system using any HomeKit client device (iPhone, iPad, HomePod, Mac, AppleWatch ...), working with the integrations supported by this.

The developed system is capable of maintaining, independently, and with a high comfort for the final user, target temperature in each zone or room, reporting to HomeKit environment about the changes that are taking place (temperature, working mode, etc.).

Installation has been done on the top floor of a house, with 4 rooms, and results have been very satisfactory, because the new system has be able to reduce energy cost considerably. In addition, integration with HomeKit system gives an extra value, thanks to the possibility of interacting with other accessories that support this framework.

Índice

1. Introducción	1
1.1. Contexto y justificación del trabajo	1
1.2. Descripción del trabajo	1
1.3. Objetivos del TFG	2
1.4. Enfoque y método seguido	2
1.5. Planificación del trabajo	6
1.6. Recursos empleados	9
1.7. Productos obtenidos	10
1.8. Breve descripción de los otros capítulos de la memoria	10
2. Antecedentes	12
2.1. Estado del arte	12
2.2. Estudio de mercado	13
3. Descripción funcional	15
3.1. iAirZoning	16
3.2. Control centralizado	19
4. Descripción detallada	25
4.1. Esquema eléctrico	26
4.2. Esquema general de hardware	27
4.3. Diseño de la Arquitectura de Software	28
4.4. Partes principales del desarrollo	30
5. Comparación de funcionamiento antes y después	35
6. Monitorización del consumo eléctrico	37
6.1. Instalación	37
6.2. Consumo eléctrico antes y después	39
7. Viabilidad técnica	40
7.1. Valoración de riesgos	41
8. Valoración económica	44
8.1. Materiales utilizados	44
8.2. Industrialización	46
9. Conclusiones	48
9.1. Descripción	48
9.2. Autoevaluación	48
9.3. Líneas de trabajo futuro	49
10. Glosario	50
11. Bibliografía	51

Lista de figuras

Tabla 1: Lista de tareas	6
Imagen 1: Diagrama de Gantt	9
Tabla 2: Equipamiento Informático utilizado	9
Tabla 3: Software utilizado.....	9
Imagen 2: Apple Xcode	10
Imagen 3: Termostato HomeKit	12
Imagen 4: iPad usado como control domótico HomeKit.....	14
Imagen 5: Mando y receptor de la unidad de A/C	15
Imagen 6: Diagrama de Bloques Funcional	17
Imagen 7: Diagrama de Bloques del Control Centralizado	18
Imagen 8: Prototipo inicial del proyecto.....	19
Imagen 9: Distribución de componentes sobre el plano de la vivienda	20
Imagen 10: Prototipo final del proyecto.....	21
Imagen 11: Prototipo instalado	21
Imagen 12: Detalle del prototipo instalado	22
Imagen 13: Acometidas del cableado hacia el prototipo	22
Imagen 14: Instalación de uno de los sensores DHT22.....	23
Imagen 15: Vista del sensor una vez instalado	24
Imagen 16: Rejilla de retorno (Izquierda). Compuerta motorizada (derecha)	24
Imagen 17: Esquema Eléctrico.....	26
Imagen 18: Esquema de conexiones de los diferentes componentes	27
Imagen 19: Diagrama de bloques del Software	28
Imagen 20: Diagrama de flujo	29
Imagen 21: Jerarquía HomeKit.....	31
Imagen 22: Pasos para añadir el accesorio a HomeKit	31
Imagen 23: iAirZoning en HomeKit.....	32
Imagen 24: Diagrama de zonas del termostato.....	32
Tabla 4: Comandos IrDA.....	34
Imagen 25: Comparativa de temperaturas	35
Imagen 26: Sistema de monitor de consumo	37
Imagen 27: Sistema de monitor de consumo instalado.....	38
Imagen 28: Comparativa de consumo.....	39
Imagen 29: Placa de desarrollo ESPDuino	42
Tabla 5: Evaluación de Riesgos	42
Tabla 6: Leyenda de la Evaluación de Riesgos.....	43
Tabla 7: Valoración económica.....	44
Tabla 8: Material necesario para la realización del proyecto.....	44
Tabla 9: Costes de industrialización	46
Tabla 10: Dedicación en horas	48

1. Introducción

El siguiente trabajo se enmarca en el ámbito de los sistemas empotrados, y más concretamente en las áreas de la climatización, el control energético y la domótica.

1.1. Contexto y justificación del trabajo

La motivación principal por la que surge este trabajo es la posibilidad de integrar un sistema convencional de aire acondicionado por conductos en el entorno domótico HomeKit¹ de Apple. Además, aprovechando las capacidades de los sistemas empotrados, aparece una motivación adicional enfocada en el ahorro energético: instalar y controlar un sistema de climatización por zonas mediante compuertas motorizadas.

Actualmente, este tipo de climatizaciones por conductos están sujetas a calentar o enfriar todas las estancias que disponen de sus difusores, teniendo como temperatura de consigna, la misma para cada una de ellas, la cual es comprobada midiendo la temperatura del aire de retorno. Este sistema es, por definición, ineficiente, puesto que obliga siempre a actuar sobre todas las estancias, calentando o enfriando un volumen de aire mucho mayor del deseado en la mayoría de los casos.

Partiendo de la instalación en una vivienda de un sistema de climatización por conductos, se desea desarrollar un sistema que sea capaz de regular la temperatura de cada habitación de forma independiente, integrado además, bajo la domótica del sistema HomeKit.

1.2. Descripción del trabajo

Se ha desarrollado un sistema que permite controlar la climatización por zonas de manera individual, con diferentes temperaturas de consigna, integrado bajo el control domótico de la plataforma HomeKit de Apple.

Este sistema monitoriza de forma constante la temperatura en cada habitación haciendo uso de los correspondientes sensores, y controla tanto las compuertas para permitir o bloquear el paso de aire proveniente de la máquina de climatización (máquina de A/C), como a la propia máquina.

El control de la máquina de A/C se realiza mediante el envío de códigos infrarrojos IrDA, mientras que el control de las compuertas se hace utilizando una serie de relés para determinar la polaridad de la alimentación de cada uno de los motores DC que hacen la función de actuadores, abriendo o cerrando las lamas de las compuerta.

¹ <https://www.apple.com/es/ios/home/> (Mayo de 2020)

El usuario es capaz de manejar el sistema, que se presenta en forma de 4 termostatos totalmente configurables dentro de la interfaz domótica HomeKit, y seleccionar el modo de trabajo (apagado, frío, calor, o automático) y la temperatura de consigna para cada una de las habitaciones. Todo ello, disponible en cualquier terminal iPhone, iPad, HomePod, Mac, AppleWatch que disponga de acceso al entorno HomeKit donde está instalado el sistema. Además, al estar dentro de este ecosistema domótico, puede hacer uso de otras características de valor añadido, como son la activación o desactivación por geolocalización, en función de la hora y el día, control de forma remota (fuera de la red de área local), etc.

1.3. Objetivos del TFG

El objetivo principal del proyecto es el ahorro energético que supondrá tanto la zonificación, como su integración domótica. Al usar HAP (*HomeKit Accessory Protocol*), se integra con el resto de componentes domóticos ya instalados en la vivienda, accediendo así a todas las opciones que nos ofrece la plataforma Apple HomeKit para su control. Con todo esto, se pueden enumerar los siguientes objetivos clave:

- Ahorro energético mediante zonificación de la climatización.
- Integración domótica en el ecosistema Apple HomeKit.
- Monitorización y presentación de la temperatura y humedad de cada estancia a través de las interfaces de Apple HomeKit.

1.4. Enfoque y método seguido

Para acometer el desarrollo del proyecto, se han tenido en consideración diferentes estrategias, principalmente a nivel de *hardware*, ya que este es el que condiciona el desarrollo del *software* del producto final, así como su coste.

Para la parte de zonificación, ya existen algunas soluciones comerciales, entre las que destacan AirZone² y MasterZone³. Sin embargo, no existe ninguna solución comercial que integre el sistema en HomeKit, y realizar una adaptación del *software* de un sistema comercial es imposible, ya que estos son totalmente cerrados.

Estas circunstancias dejan una única línea a seguir: desarrollar en este proyecto tanto el *hardware* como el *software* a utilizar. Así pues, en el marco del trabajo a realizar, hay que tomar decisiones sobre varios aspectos del *hardware*:

² <http://www.airzone.es> (Mayo de 2020)

³ <http://www.masterzone.es> (Mayo de 2020)

A. Microcontrolador

Es necesario uno que disponga de suficientes GPIO, para controlar los actuadores y sensores necesarios, y conexión de red, así como de un SDK estable y robusto, con posibilidad de integración en HomeKit. Con estos factores, hay dos candidatos posibles:

- **ESP32⁴ de Espressif:** Se trata de un microcontrolador muy potente (2 núcleos), con conexión WiFi, multitud de GPIO disponibles para su uso, un SDK muy robusto, e integración con HomeKit, tanto en su licencia comercial como no comercial. Este sería el candidato ideal si se fuese a usar la librería HomeKit bajo su licencia comercial, pero no es el caso, y la librería para uso no comercial está aún en una fase de desarrollo temprana que la hace inadecuada para el propósito de este proyecto.
- **ESP8266⁵ de Espressif:** Este es similar al anterior, pero con menos potencia y menos GPIO disponibles. También cuenta con un SDK muy estable y, aunque no dispone de integración en HomeKit bajo licencia comercial, sí que tiene una librería HomeKit no comercial estable y robusta. Sin embargo, el número de GPIO libres es insuficiente, pero se puede utilizar una interfaz de expansión MCP23017⁶ para tener hasta 16 GPIO más, usando únicamente 2 GPIO del microcontrolador.

Es por lo anterior, que el controlador utilizado en el proyecto es el ESP8266.

B. Sensores de temperatura

Para monitorizar la temperatura de cada estancia, es necesario hacer uso de algún tipo de sensor, así como de un sistema para trasladar esa información al microcontrolador. Aquí se puede optar por instalar sensores de temperatura en microcontroladores adicionales (uno por habitación) y que envíen los datos correspondientes al principal, bien por RF o por WiFi. Otra opción sería conectar directamente los diferentes sensores al microcontrolador principal, eliminando tanto complejidad en el desarrollo, como *hardware* necesario.

La segunda opción es la más indicada si no queremos depender de *hardware* adicional, pero hay que tener en cuenta que el número de GPIO disponibles es limitado, así que los sensores deberán usar el protocolo 1-Wire. Los DHT11 y DHT22⁷ de Adafruit Industries son buenas

⁴ <https://www.espressif.com/en/products/socs/esp32/overview> (Marzo de 2020)

⁵ <https://www.espressif.com/en/products/socs/esp8266/overview> (Marzo de 2020)

⁶ <https://www.adafruit.com/product/732> (Marzo de 2020)

⁷ <https://learn.adafruit.com/dht> (Marzo de 2020)

opciones que utilizan ese protocolo, y además permiten obtener también lecturas de humedad. Sin embargo, queda el tema de la longitud máxima de cable a la que son funcionales: Según las especificaciones de estos, se pueden usar hasta con 20 metros de longitud de cable, siempre que se alimenten a 5VDC con suficiente corriente. Para la instalación del presente proyecto, la longitud máxima de cable es de 5 metros; a pesar de que inicialmente estarán alimentados por una tensión de 3,3VDC, los DHT22 parecen la mejor opción.

C. Compuertas motorizadas

En el mercado no hay gran variedad de este tipo de compuertas, teniendo disponibles compuertas con actuadores de corriente continua de baja tensión (12V o 24V) y de corriente alterna de 230V.

Los de corriente alterna funcionan con un solenoide que hace que la compuerta permanezca abierta mientras este reciba corriente, ya que son del tipo "normalmente cerrada", teniendo la problemática de la disipación de calor y el desgaste que causa esta en el solenoide durante el tiempo que mantiene la compuerta abierta.

Por otra parte, los de corriente continua no presentan este problema, pero se necesitan 2 GPIO para manejar cada compuerta de este tipo, mientras que las de corriente alterna sólo necesitan uno, aunque el desarrollo del control de este tipo de compuertas entraña mayor complejidad que el de las de corriente alterna.

La elección final ha sido el uso de compuertas con actuadores de corriente continua, puesto que, por seguridad, todas las compuertas deberán permanecer abiertas siempre que no se esté utilizando el sistema. Y gracias a la interfaz de expansión MCP23017, la carencia de GPIO suficientes del microcontrolador no supone ningún problema.

D. Control de la máquina de A/C

El microcontrolador tiene que ser capaz de enviar órdenes a la máquina de climatización, y para ello existen 3 opciones:

- Usar el protocolo de comunicaciones que utiliza el mando remoto cableado: Se trata de un protocolo no documentado que utiliza 2 hilos, por los que también se suministra la alimentación del propio mando. Debido a su alto grado de complejidad y dificultad, queda totalmente descartado.
- Usar un módulo comercial WiFi para controlar la máquina a través de Internet: Este sistema comercial es un módulo adicional que se le instala a la máquina de A/C y permite, mediante

una conexión de red WiFi, interactuar con la máquina a través de una aplicación móvil. El protocolo usado está documentado por terceros, pero la dependencia de una conexión a Internet y un servidor externo (MELCloud⁸) hace que no sea posible utilizar el sistema en caso de que existan problemas de conexión con el ISP.

- Usar el receptor de infrarrojos: Instalando un emisor IrDA al microcontrolador, este tendría la capacidad de enviar órdenes a la máquina de A/C de forma directa. La utilización de este sistema implicaría la captura de los códigos necesarios del mando remoto IrDA para su posterior integración en el dispositivo desarrollado. De las 3 opciones, esta es elegida por ser independiente de factores externos y su dificultad es aceptable.

En resumen, se va a hacer uso de un microcontrolador ESP8266, con una interfaz de expansión MCP23017, sensores de temperatura y humedad DHT22, compuertas motorizadas de corriente continua, y un emisor IrDA para el control de la máquina de A/C.

⁸ <https://www.melcloud.com> (Mayo de 2020)

1.5. Planificación del trabajo

A continuación, se presenta una tabla con cada una de las tareas realizadas, junto con su estimación de horas inicial, el número de horas reales invertidas, y el porcentaje de desarrollo alcanzado. El desarrollo ha sido llevado a cabo principalmente en las fases 1 y 2, dejando la fase 3 para instalación, pruebas y mejoras. Por último, la elaboración de la memoria se ha realizado a partir de la fase 2.

Tarea	Horas		
	Est.	Real	%
Fase 0	45	45	100 %
01 Propuesta	20	20	100 %
02 PEC1 - Plan de Trabajo	25	25	100 %
Fase 1	90	86	100 %
03 Preparación del SDK, librerías y entorno de desarrollo	10	8	100 %
04 Conexión WiFi y su Watchdog	10	9	100 %
05 Creación y manejo de los accesorios HomeKit del tipo Termostato	15	16	100 %
06 Lectura de datos de los sensores	10	8	100 %
07 Creación y manejo de los accesorios HomeKit del tipo Sensor de Humedad	10	7	100 %
08 Control de la interfaz de expansión MCP23017	15	17	100 %
09 Control de los 8 relés para manejar los motores de las compuertas	10	10	100 %
10 PEC2 - Seguimiento	10	11	100 %
Fase 2	70	67	100 %
11 Instalación y prueba del sistema de medición de consumo	5	3	100 %
12 Captura de los códigos IR necesarios del mando original del A/C	10	12	100 %
13 Envío de los diferentes códigos IR para controlar la máquina de A/C	15	16	100 %
14 Control de las compuertas en base a las temperaturas actual y objetivo	15	17	100 %
15 Control de la máquina de A/C en base al estado de los termostatos	15	12	100 %
16 PEC3 - Seguimiento	10	7	100 %
Fase 3	65	76	100 %
17 Instalación física del proyecto	35	46	100 %
18 Pruebas finales y posibles correcciones	20	23	100 %
19 Código Final	10	7	100 %

Tarea	Horas		
	Est.	Real	%
Memoria	60	58	100 %
20 Fase 2 - Previo de la Memoria	15	15	100 %
21 Fase 3 - Continuación de la Memoria	15	15	100 %
22 Elaboración Final de la Memoria	30	28	100 %

Tabla 1: Lista de Tareas

Este es el cronograma correspondiente tanto a la planificación como al desarrollo del proyecto:

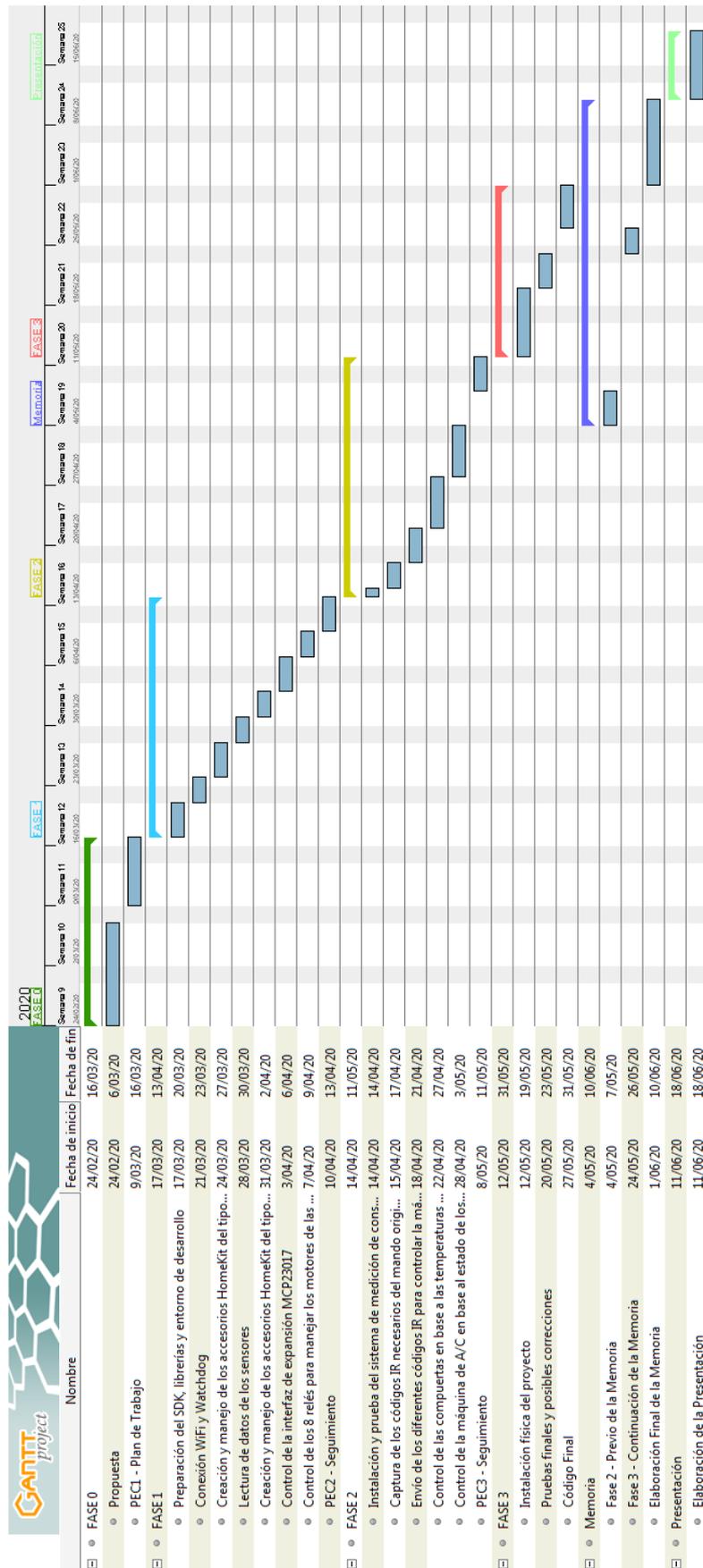


Imagen 1: Diagrama de Gantt

1.6. Recursos empleados

Para el desarrollo y montaje del proyecto, se han utilizado las siguientes herramientas y equipos:

Equipamiento Informático utilizado	
Apple iMac Retina 5K, 27 pulgadas, 2019	<ul style="list-style-type: none">- Procesador: Intel Core i9 de 8 núcleos (16 hilos).- RAM: 32GB de memoria DDR4 2667MHz.- Almacenamiento: SSD de 2TB.- Gráfica: Radeon Pro Vega 48 de 8GB.
Apple iPad Pro, 11 pulgadas	<ul style="list-style-type: none">- Modelo: WiFi + Celular.- Almacenamiento: 256GB.
Apple iPad Air 2	<ul style="list-style-type: none">- Modelo: WiFi + Celular.- Almacenamiento: 64GB.
Apple iPhone XS	<ul style="list-style-type: none">- Almacenamiento: 64GB.
Apple Watch Series 4	<ul style="list-style-type: none">- Modelo: WiFi + Celular.
Apple TV HD	<ul style="list-style-type: none">- Almacenamiento: 32GB.

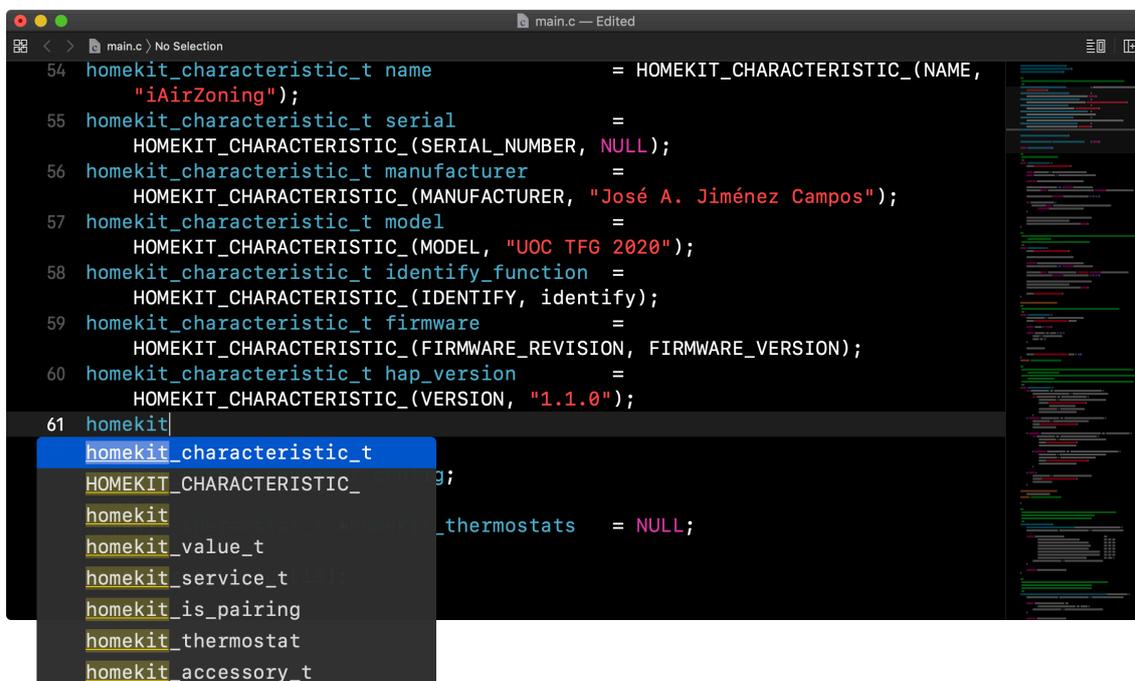
Tabla 2: Equipamiento Informático utilizado

Software específico utilizado	
Apple Xcode	<ul style="list-style-type: none">- Entorno de desarrollo, para programar en C.- Adaptado al SDK utilizado.
Aplicación Casa de Apple y Asistente de Voz Siri	<ul style="list-style-type: none">- Interfaz de usuario para los accesorios HomeKit.
EasyEDA	<ul style="list-style-type: none">- Herramienta para diseñar circuitos electrónicos.
Fritzing	<ul style="list-style-type: none">- Aplicación para crear modelos con componentes electrónicos.
Draw.io	<ul style="list-style-type: none">- <i>Software</i> de creación de diagramas de bloques.
GanttProject	<ul style="list-style-type: none">- <i>Software</i> de creación de diagramas de Gantt.
Apple Pages	<ul style="list-style-type: none">- Procesador de texto.

Tabla 3: *Software* utilizado

Además, también ha sido necesaria la utilización de otro tipo de herramientas "más tradicionales", tales como soldador, taladro, radial, destornilladores, alicates, martillo, etc.

Seguidamente, se puede ver un detalle del editor de código Apple Xcode⁹ utilizado en el desarrollo del proyecto. Este entorno de desarrollo permite utilizar diferentes lenguajes de programación, tales como Swift¹⁰, Objective-C o C; este último ha sido el usado para la programación del código de este trabajo.



```
54 homekit_characteristic_t name = HOMEKIT_CHARACTERISTIC_(NAME,
    "iAirZoning");
55 homekit_characteristic_t serial =
    HOMEKIT_CHARACTERISTIC_(SERIAL_NUMBER, NULL);
56 homekit_characteristic_t manufacturer =
    HOMEKIT_CHARACTERISTIC_(MANUFACTURER, "José A. Jiménez Campos");
57 homekit_characteristic_t model =
    HOMEKIT_CHARACTERISTIC_(MODEL, "UOC TFG 2020");
58 homekit_characteristic_t identify_function =
    HOMEKIT_CHARACTERISTIC_(IDENTIFY, identify);
59 homekit_characteristic_t firmware =
    HOMEKIT_CHARACTERISTIC_(FIRMWARE_REVISION, FIRMWARE_VERSION);
60 homekit_characteristic_t hap_version =
    HOMEKIT_CHARACTERISTIC_(VERSION, "1.1.0");
61 homekit_
```

- homekit_characteristic_t
- HOMEKIT_CHARACTERISTIC_
- homekit_thermostats = NULL;
- homekit_value_t
- homekit_service_t
- homekit_is_pairing
- homekit_thermostat
- homekit_accessory_t

Imagen 2: Apple Xcode

1.7. Productos obtenidos

En la finalización del desarrollo del proyecto, se han obtenido dos productos a destacar:

- Prototipo *hardware* del sistema de control.
- *Firmware* para el sistema de control, compatible con Apple HomeKit.

1.8. Breve descripción de los otros capítulos de la memoria

A continuación, se presenta una breve descripción de los capítulos restantes:

2. Antecedentes

Aproximación a las tecnologías más relevantes utilizadas, y estudio de mercado.

3. Descripción funcional

Explicación, a grandes rasgos, del funcionamiento del sistema desarrollado.

⁹ <https://developer.apple.com/xcode/> (Junio de 2020)

¹⁰ <https://www.apple.com/es/swift/> (Junio de 2020)

4. Descripción detallada

Análisis técnico y detallado del sistema.

5. Comparación de funcionamiento antes y después

Evolución del cambio de temperatura antes y después de usar el sistema desarrollado.

6. Monitorización del consumo eléctrico

Comparación del consumo eléctrico del A/C usando el sistema desarrollado y sin la utilización de este.

7. Viabilidad técnica

Estudio de las tecnologías usadas desde el punto de vista de su viabilidad, así como sus puntos fuertes y débiles.

8. Valoración económica

Análisis económico del desarrollo del proyecto y su posible industrialización.

9. Conclusiones

Reflexiones y líneas futuras acerca del desarrollo del proyecto.

10. Glosario

Definiciones de términos específicos empleados.

11. Bibliografía

Documentación utilizada.

2. Antecedentes

Este proyecto responde a la necesidad de reducir el gasto energético producido por una máquina de climatización por conductos al actuar sobre todas las habitaciones de una misma planta. La instalación hace que el sistema actúe de forma indiscriminada sobre todas las habitaciones de forma simultánea, sin importar si están todas en uso, o si la temperatura a alcanzar es distinta en cada una de las habitaciones.

2.1. Estado del arte

Para la realización del proyecto, se han utilizado diferentes tecnologías. A continuación se destacan aquellas que han sido de mayor relevancia:

- **HomeKit**

Sistema de control domótico desarrollado por Apple Inc. Su principal uso es a través de HAP y, entre sus principales características, podemos destacar la seguridad en las comunicaciones con cifrado de datos y métodos de autenticación. Además, posee una interfaz de usuario muy cuidada, integrada perfectamente en los sistemas de Apple, tal y como se muestra en la siguiente imagen del control de un termostato.

Este protocolo, en su versión no comercial¹¹, es el utilizado para la comunicación entre el sistema iAirZoning y los usuarios.

Gracias al uso de HAP, las conexiones entre el accesorio termostato y los diferentes dispositivos de control de los usuarios (iPhone, iPad...) cuentan con autenticación y cifrado de datos.

Además, otra característica importante es el hecho de que el accesorio no necesita conexión a Internet, sino tan solo conexión a la red de área local (LAN). El acceso remoto (fuera de la LAN) se realiza utilizando un dispositivo puente como central de accesorios¹².



Imagen 3: Termostato HomeKit

En definitiva, gracias a HAP se tiene un dispositivo con conexiones seguras, y con una interfaz de usuario estándar y accesible.

¹¹ <https://developer.apple.com/homekit/specification/> (Abril de 2020)

¹² <https://support.apple.com/es-es/HT207057> (Abril de 2020)

- **Integración de los sensores**

Para la toma de temperatura de las diferentes estancias, se ha hecho uso de sensores DHT22 que usan el protocolo 1-Wire. Estos sensores tienen la capacidad de medir, tanto la temperatura, como la humedad relativa, a intervalos pequeños de tiempo, de hasta 2 segundos, aunque su precisión varía cuanto más pequeño sea este. Para el desarrollo de este proyecto, se ha utilizado un intervalo de 30 segundos, que garantiza que la lectura de los datos del sensor tenga bastante precisión.

- **Framework esp-open-rtos¹³**

Se trata de un *framework* desarrollado por la comunidad, con integración WiFi, basado en FreeRTOS¹⁴ y Espressif IOT RTOS SDK¹⁵. Su uso está enfocado tanto para productos comerciales, como para proyectos de código abierto. El hecho de que este SDK se utilice en desarrollos comerciales, hace tener una idea de su robustez y estabilidad, permitiendo desarrollar *firmwares* de calidad profesional. Entre sus principales características, están:

- *Framework* de calidad profesional para proyectos RTOS con conectividad WiFi, para microcontroladores ESP8266.
- Código fuente abierto para todas las capas por encima de la capa de enlace de red.
- Fácil integración con proyectos escalables.
- Diferentes opciones de compilación.

2.2. Estudio de mercado

A día de hoy, existen diferentes marcas comerciales que ofrecen la zonificación de los sistemas de climatización por conductos. Una de estas marcas es AirZone¹⁶, la cual se podría considerar como la marca líder del sector. De hecho, las compuertas motorizadas usadas en el presente proyecto son de esta marca.

AirZone ofrece una amplia gama de productos para poder crear diferentes zonas en instalaciones de aire acondicionado y calefacción por conductos. A continuación, se detallan las principales diferencias con este proyecto:

¹³ <https://github.com/SuperHouse/esp-open-rtos> (Marzo de 2020)

¹⁴ <https://www.freertos.org> (Marzo de 2020)

¹⁵ https://github.com/espressif/ESP8266_RTOS_SDK (Marzo de 2020)

¹⁶ <http://www.airzone.es> (Junio de 2020)

Puesto que es necesario disponer de un sensor de temperatura en cada zona o estancia, AirZone hace uso de termostatos individuales, que cuentan con una pantalla y controles físicos, para que el usuario pueda interactuar con ellos. Cada zona necesita la instalación de uno de estos termostatos, lo que hace que el coste se incremente. En este proyecto, tan sólo es necesario la instalación de un simple sensor de temperatura, puesto que la interacción del usuario se realiza a través de los dispositivos cliente Apple HomeKit.

AirZone utiliza *hardware* propio para la interfaz de control de la máquina de A/C, dejando el control original de esta totalmente inservible. Sin embargo, con iAirZoning, se preserva el control original de la máquina, pudiendo ser usado en caso de emergencia. Además, al usar el control propio del A/C, se reducen costes ya que no hay necesidad de instalar *hardware* adicional.

Por último, el punto fuerte en el que este proyecto es muy superior a los productos de AirZone: la integración con Apple HomeKit. De hecho, no existe ningún producto de este tipo en el mercado que se integre en el ecosistema domótico de Apple. Así, es posible instalar diferentes interfaces de usuario, ya sea mediante iPods o iPads (Imagen 4), con las que poder interactuar, ya no sólo con iAirZoning, sino también con el resto de dispositivos HomeKit. A pesar de que el precio de estos productos puede ser elevado, sigue siendo más económico que la instalación de las interfaces que ofrece AirZone, y hay que tener en cuenta que estas últimas sólo sirven para controlar sus productos.

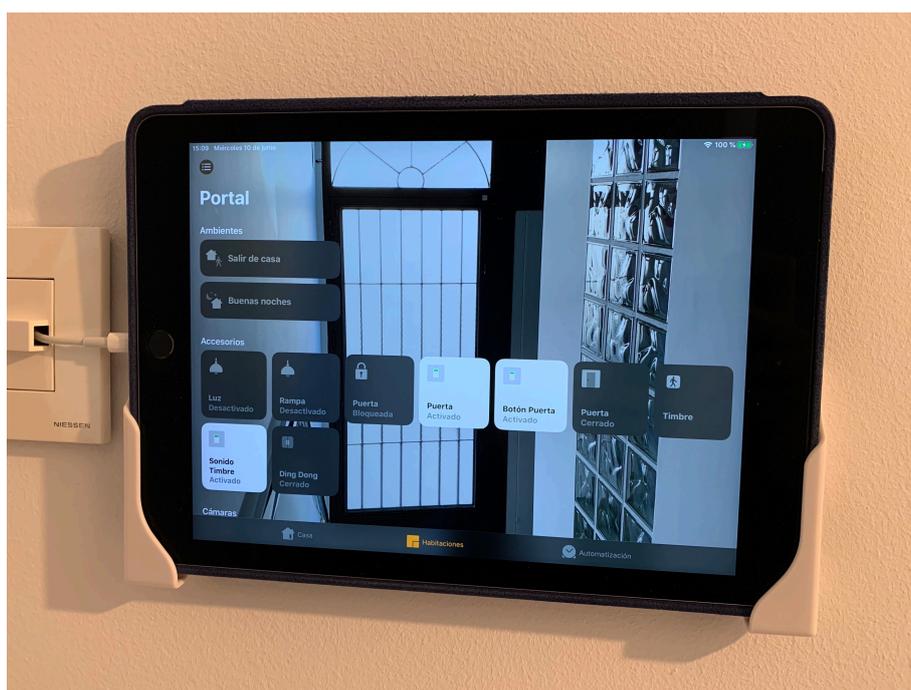


Imagen 4: iPad usado como control domótico HomeKit

3. Descripción funcional

A partir de una instalación de aire acondicionado (A/C) centralizado por conductos de una vivienda, se instala un sistema de compuertas motorizadas para controlar el flujo de aire en cada habitación en función de la temperatura registrada por su correspondiente sensor y la temperatura de consigna. Este sistema se maneja a través del entorno domótico HomeKit de Apple, haciendo uso de HAP vía WiFi. Todo el *hardware* está controlado íntegramente por un microcontrolador ESP8266 de Espressif.



Imagen 5: Mando y receptor de la unidad de A/C

Por otra parte, la máquina de A/C se controla mediante el envío de códigos infrarrojos IrDA, tal y como se aprecia en la imagen de la izquierda.

Con este sistema se pretende tener un control de climatización más eficiente y confortable, haciendo uso de una instalación estándar, ya que así el A/C no actúa por igual y de forma indiscriminada en cada uno de los 4 dormitorios a los que llega. Además, permite establecer diferentes temperaturas para cada uno de ellos.

Como valor añadido, gracias al uso de HAP, se puede controlar utilizando directamente la aplicación oficial *Casa*¹⁷ de Apple, incluida de serie en iOS, iPadOS, WatchOS y macOS, quedando totalmente integrado en el ecosistema HomeKit. Además, si se cuenta con una central de accesorios HomeKit, se pueden crear automatizaciones domóticas, como por ejemplo, que el sistema se apague automáticamente cuando no haya nadie en casa.

¹⁷ <https://www.apple.com/es/ios/home/> (Mayo de 2020)

3.1. iAirZoning

a) Requerimientos:

Dada la naturaleza del proyecto, los siguientes requerimientos son principales y necesarios:

- Control de Accesorios HomeKit del tipo Termostato usando HAP.
- Lectura de la temperatura de cada una de las habitaciones.
- Control de las compuertas motorizadas.
- Control por IrDA de la máquina de A/C.
- Control independiente de la temperatura de consigna de cada habitación.

Además, como requerimientos secundarios están:

- Instalación de un sistema de medición de consumo eléctrico para comparar el gasto antes y después de la instalación del proyecto.
- Lectura del grado de humedad de cada una de las habitaciones, así como su correspondiente presentación a través de la interfaz de usuario de HomeKit.

b) Diagrama de bloques funcional:

A continuación se presenta un diagrama de bloques donde se puede ver la manera en la que los usuarios interactúan con el sistema a través de Apple HomeKit haciendo uso de HAP, para interactuar con el Control Centralizado del sistema de A/C.

Destacar el hecho de que la central de accesorios HomeKit se encarga, además del acceso remoto, de la ejecución de las posibles automatizaciones creadas, tales como el encendido y apagado automático a determinadas horas, o en base a la geolocalización de los usuarios.

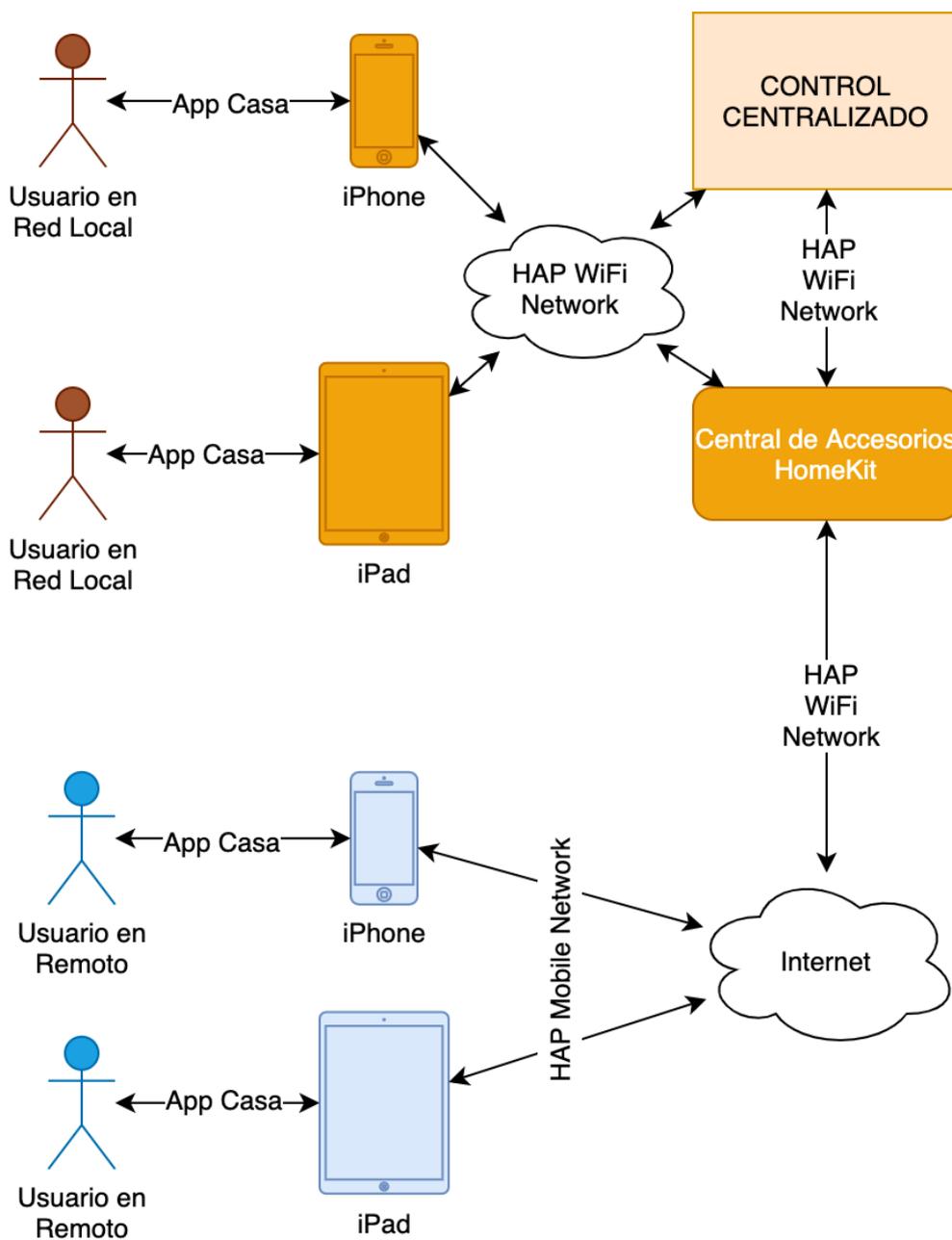


Imagen 6: Diagrama de Bloques Funcional

En el siguiente diagrama de bloques se muestra en detalle el **Control Centralizado**, donde se aprecia la forma en la que el microcontrolador maneja los diferentes periféricos: actuadores, sensores de temperatura, etc.

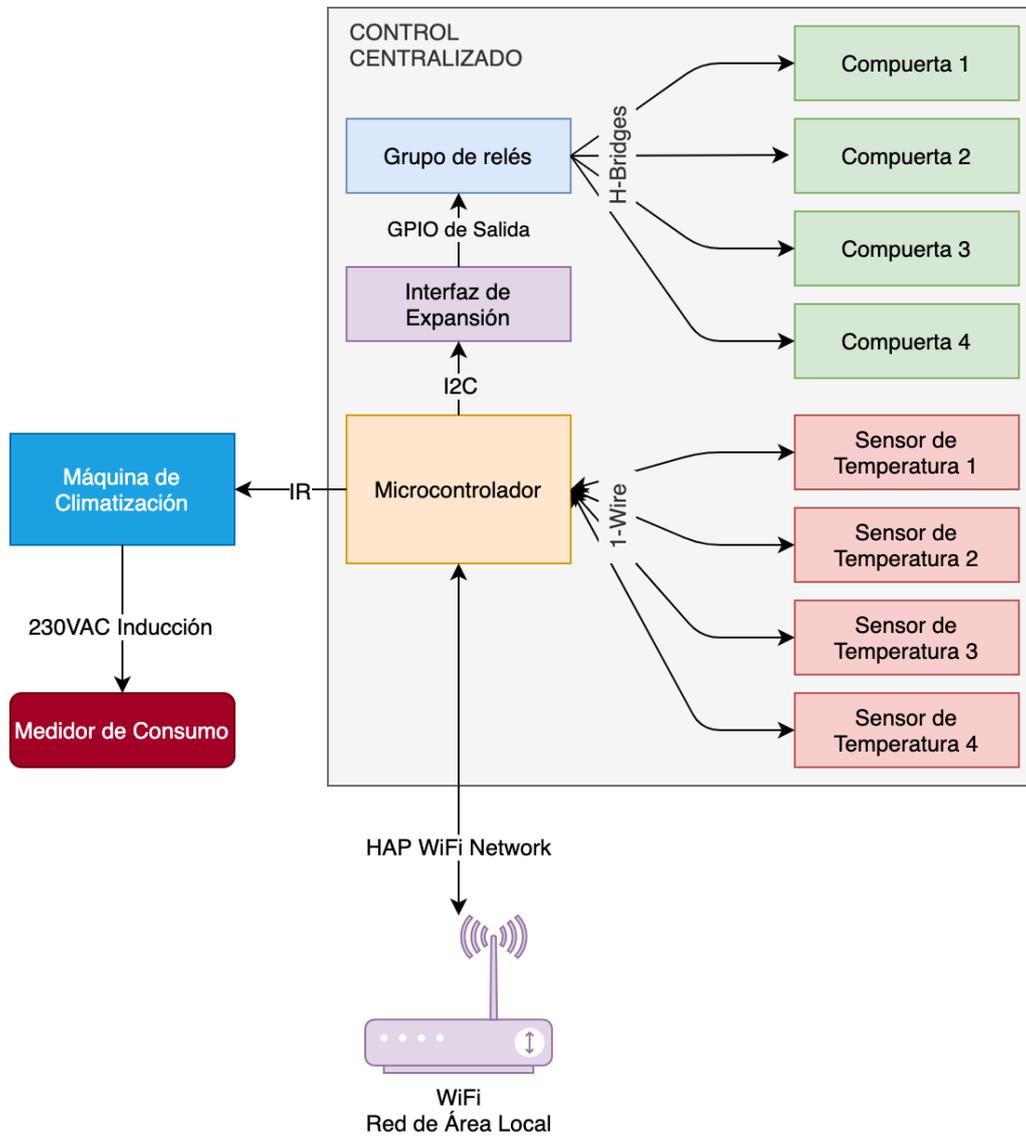


Imagen 7: Diagrama de Bloques del Control Centralizado

3.2. Control centralizado

a) Microcontrolador:

En este proyecto se ha hecho uso de diferentes tecnologías y *hardware* para conseguir construir un prototipo plenamente funcional y operativo para su posterior instalación en la vivienda, con la intención de usarlo de forma rutinaria, y no sólo para pruebas de concepto y demostraciones.

El microcontrolador es el encargado de toda la gestión de la lógica y las comunicaciones, tanto WiFi, como de los componentes internos que conforman el control centralizado. Es decir, este se encarga de leer la temperatura y la humedad de cada habitación, procesar el estado y la temperatura de consigna establecida para cada zona, evaluar la situación, y actuar en consecuencia, manipulando las compuertas motorizadas y enviando las correspondientes órdenes a la máquina de A/C. Además, se encarga de administrar las interfaces HomeKit de usuario mediante HAP a través de la conexión WiFi.

b) Prototipo inicial:

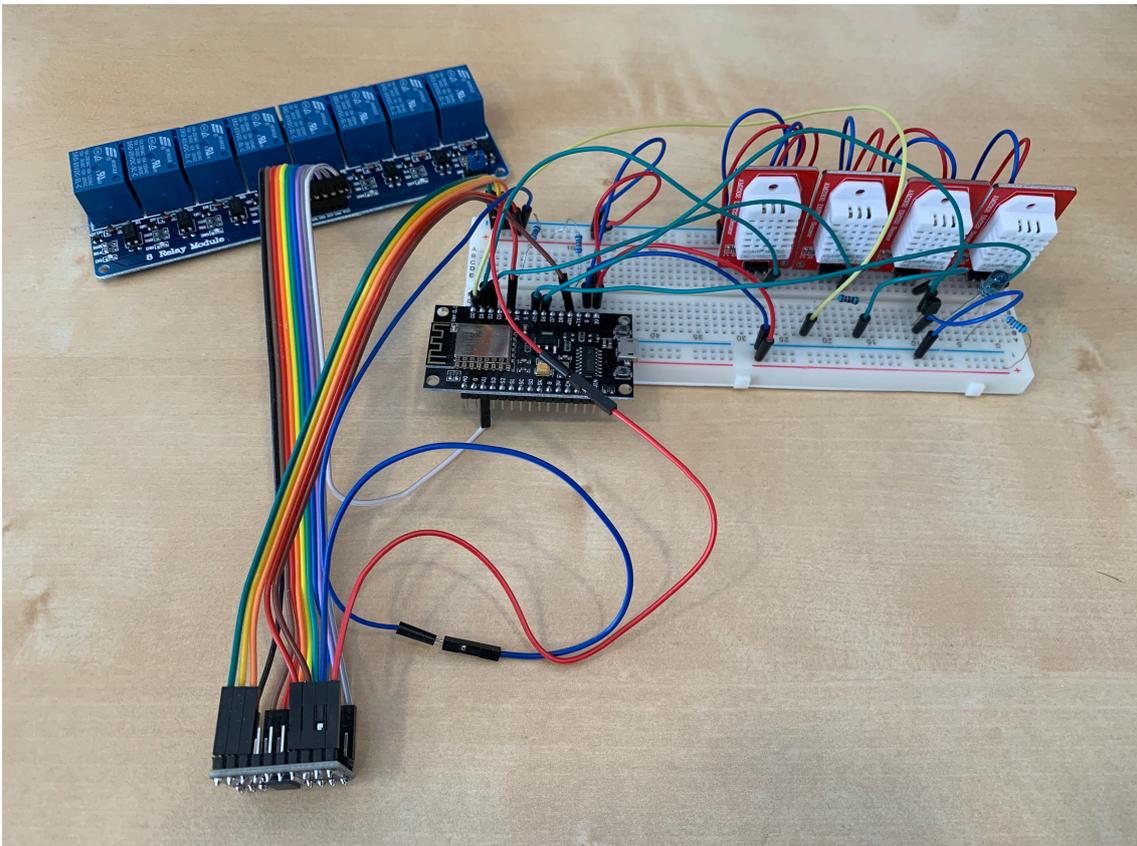


Imagen 8: Prototipo inicial del proyecto

c) Plano de la instalación:

La instalación se ha realizado en un entorno real, con la disposición que figura en el siguiente plano. La máquina de climatización se encuentra alojada en el falso techo del cuarto de baño principal de la planta alta.

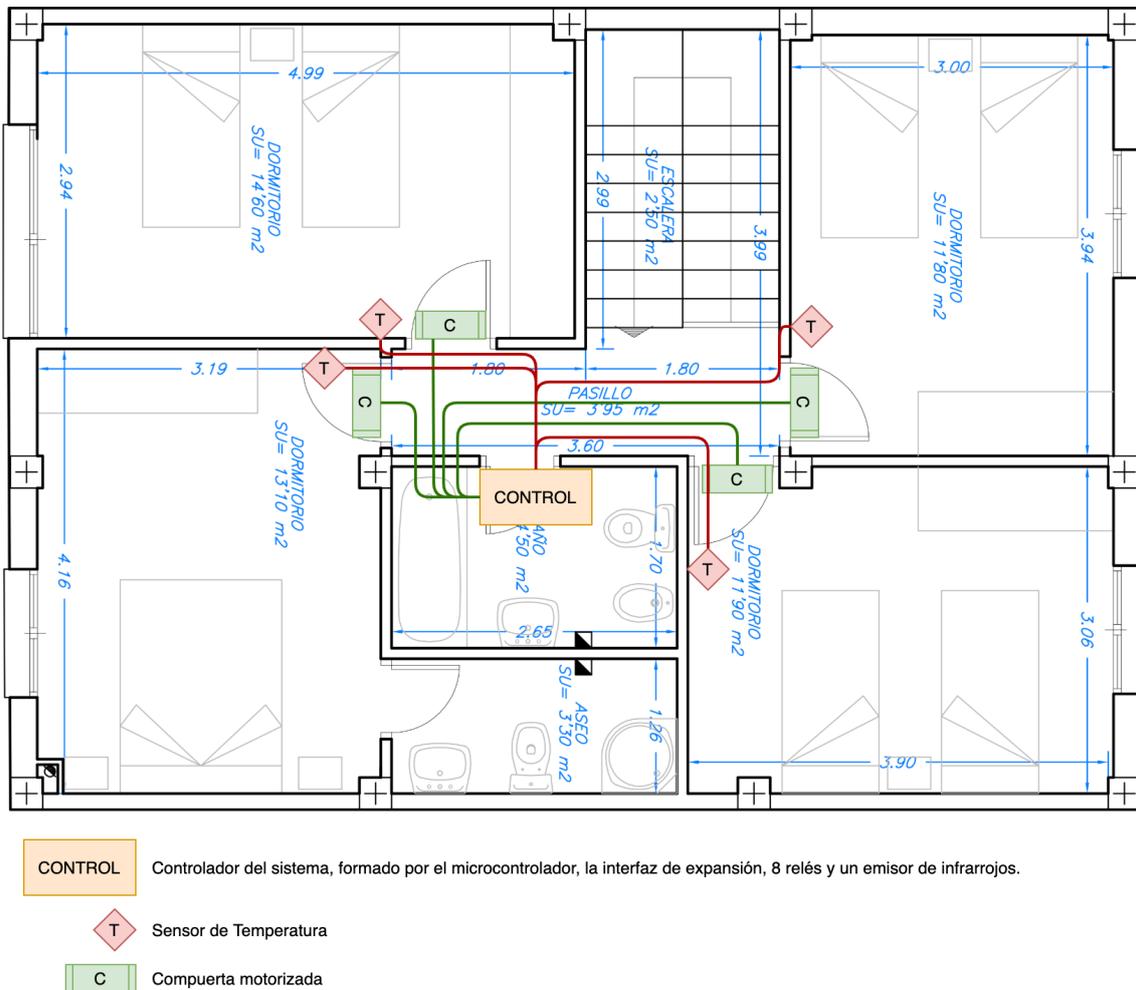


Imagen 9: Distribución de componentes sobre el plano de la vivienda

El cableado de alimentación de las compuertas va por el interior de los conductos, mientras que el cableado de los sensores de temperatura lo hace por el hueco de retorno de aire. La unidad receptora IrDA queda oculta en el falso techo del cuarto de baño, pero fácilmente accesible para poder manejar la máquina de A/C con su mando a distancia original en caso de producirse una emergencia.

d) Instalación del prototipo final:

A continuación se muestra el prototipo final del control centralizado antes de ser instalado en el falso techo del cuarto de baño.

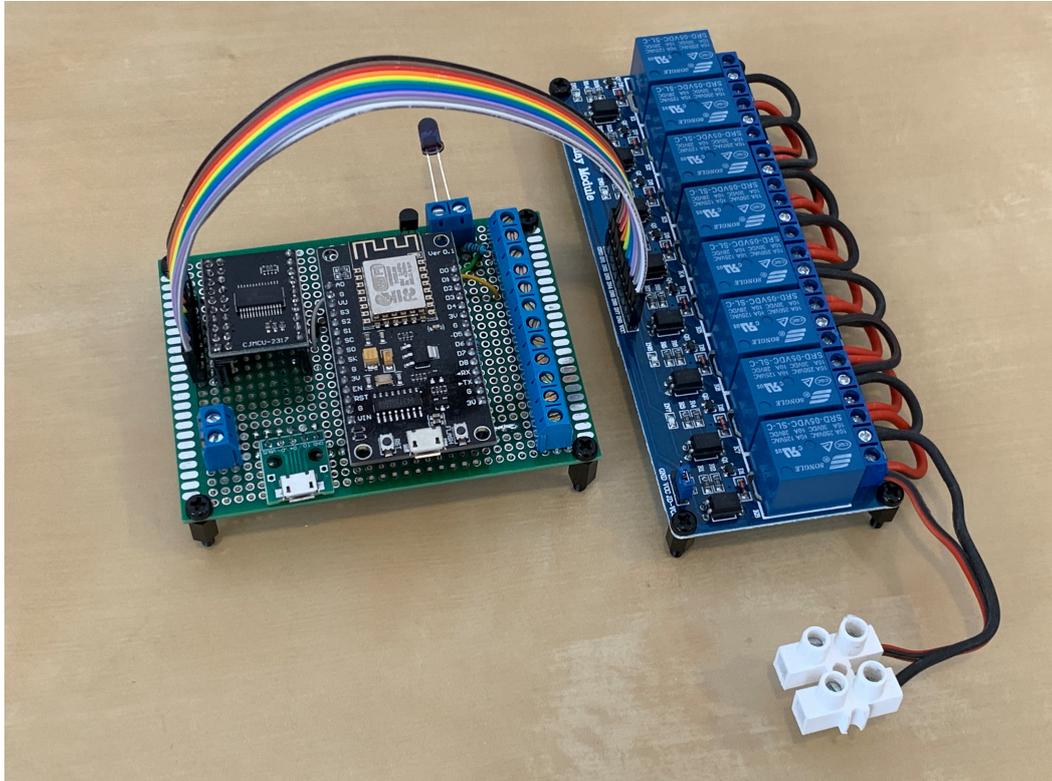


Imagen 10: Prototipo final del proyecto

Aquí se presenta ya instalado, con el LED emisor IrDA dirigido hacia el receptor infrarrojos de la máquina de A/C.



Imagen 11: Prototipo instalado

En la siguiente imagen se muestra una vista más cercana del prototipo, donde pueden verse las conexiones del cableado blanco de los sensores de temperatura, y el cableado rojo y negro de los actuadores de las compuertas (12VDC).

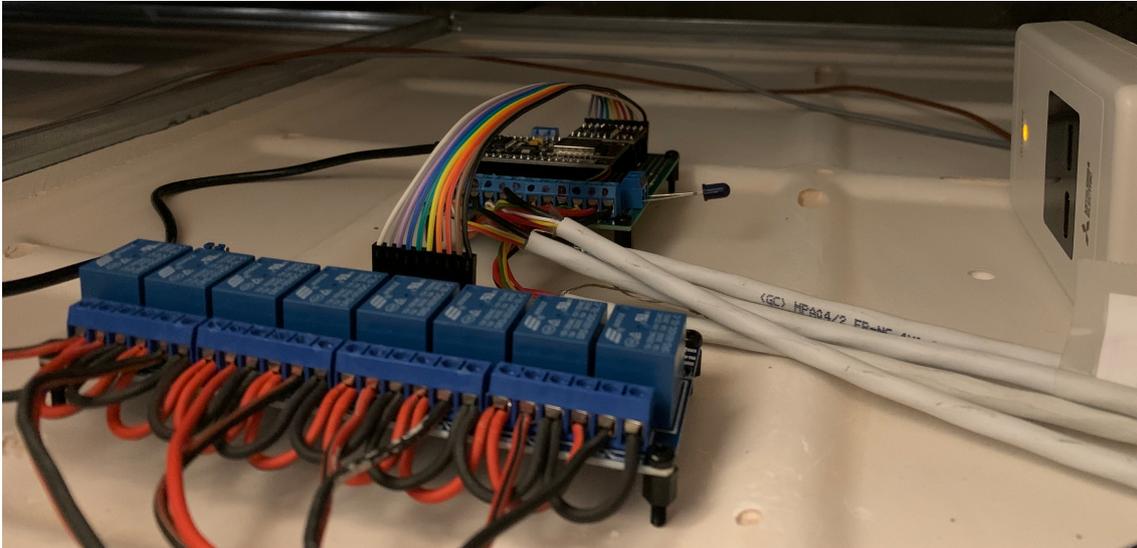


Imagen 12: Detalle del prototipo instalado

A continuación se pueden apreciar las acometidas del cableado de los actuadores de las compuertas (cables paralelos de color rojo y negro que salen del conducto del A/C) y del cableado de los sensores (cables blancos blindados que salen del hueco del falso techo del pasillo, provenientes de las respectivas habitaciones).



Imagen 13: Acometidas del cableado hacia el prototipo

En detalle, la instalación de uno de los cuatro sensores encargados de la monitorización de la temperatura y humedad.

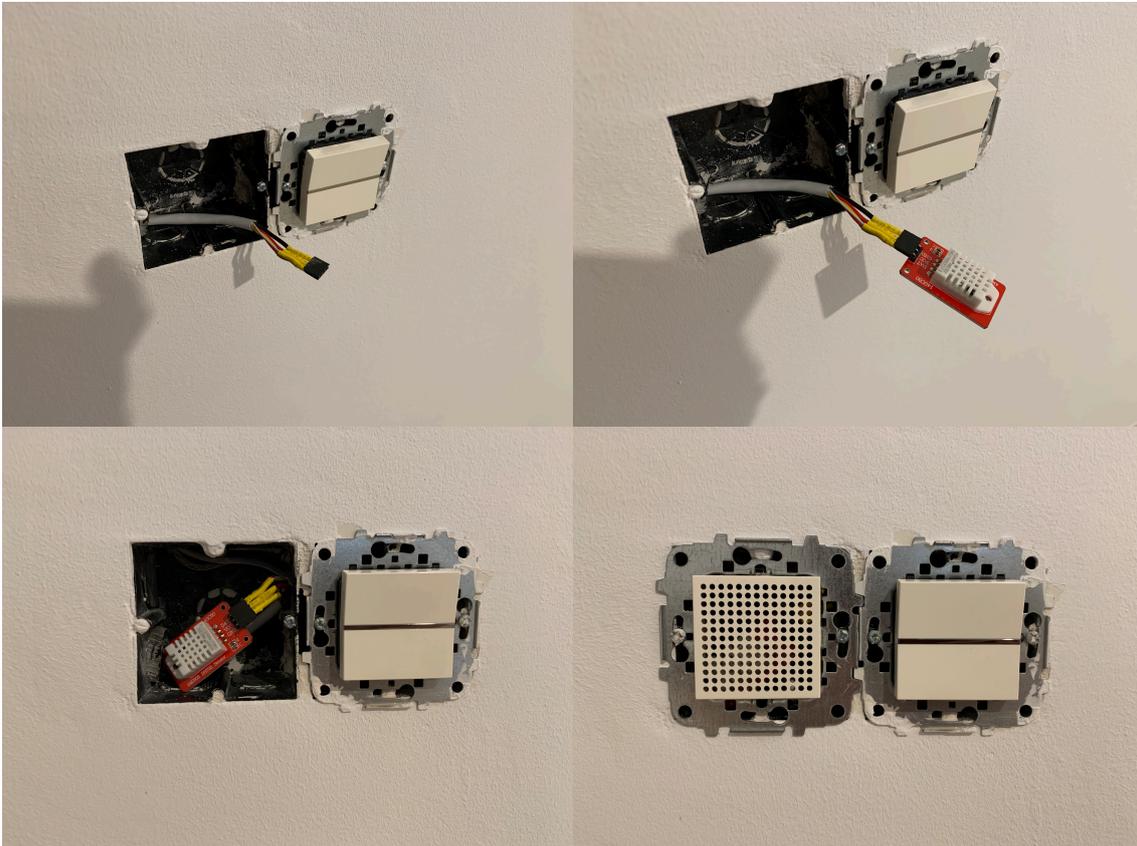


Imagen 14: Instalación de uno de los sensores DHT22

El resultado final de la instalación de los sensores hace que estos queden perfectamente integrados junto a los mecanismos de la instalación eléctrica actual.

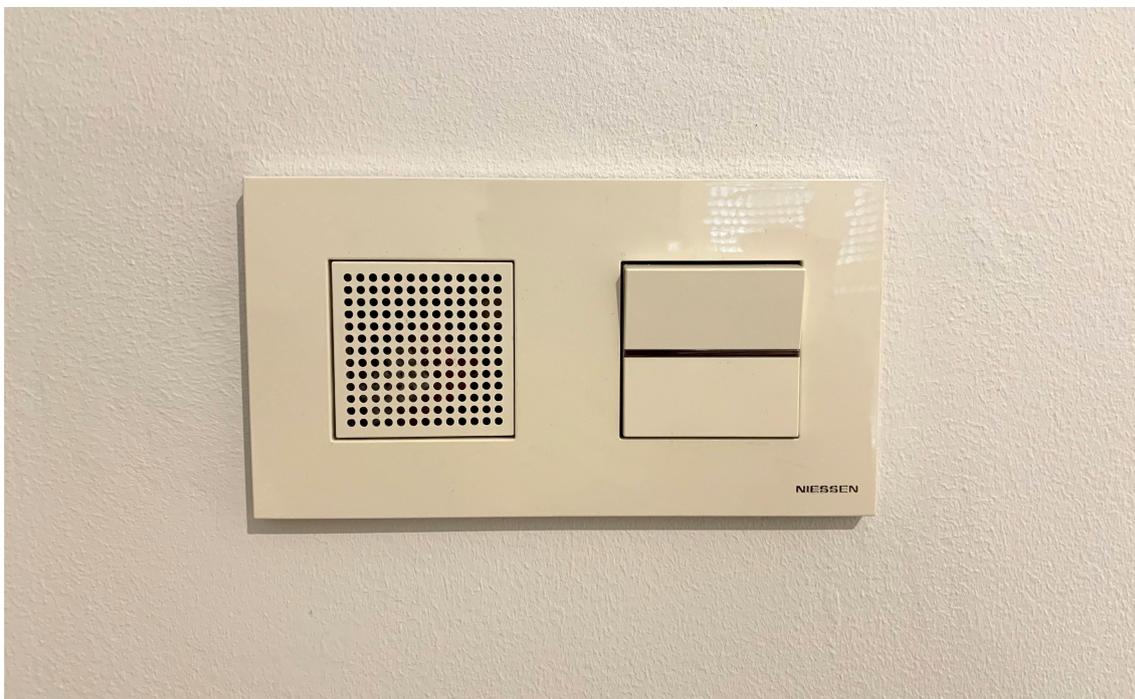


Imagen 15: Vista del sensor una vez instalado

Por último, se muestra una imagen de una de las compuertas motorizadas instaladas en la salida de aire de uno de los conductos (derecha). Se puede apreciar una caja gris en la parte derecha de la compuerta, que alberga el actuador propiamente dicho, compuesto por el motor DC y la caja de engranajes de par reductor.



Imagen 16: Rejilla de retorno (Izquierda). Compuerta motorizada (Derecha)

4. Descripción detallada

El núcleo del sistema que se encarga de gestionar las interfaces, comunicaciones, actuadores, etc. es el microcontrolador ESP8266.

Las principales razones por las que se ha escogido el microcontrolador ESP8266 son que su CPU dispone de la potencia necesaria, así como suficientes GPIO libres para poder gestionar los diferentes sensores y actuadores, conjuntamente con la interfaz de expansión MCP23017.

Entre las características a destacar del ESP8266 están:

- Arquitectura RISC¹⁸ de 32bits.
- Frecuencia de reloj de la CPU de 80MHz.
- 17 GPIO (Hasta 13 pueden ser usados por el usuario, dependiendo de la configuración).
- Interfaz WiFi con pila TCP/IP completa.
- Bajo consumo energético.
- Bajo voltaje de funcionamiento: 3,3VDC.

Como factor de forma, se ha elegido la placa de desarrollo NodeMCU¹⁹, que expone todos los GPIO del microcontrolador ESP8266, e incorpora un regulador de voltaje, así como un conector USB con un convertidor RS232 (CH340) enlazado a la UART0 del sistema.

¹⁸ https://en.wikipedia.org/wiki/Reduced_instruction_set_computer (Junio de 2020)

¹⁹ <https://github.com/nodemcu/nodemcu-devkit-v1.0> (Mayo de 2020)

4.1. Esquema eléctrico

En la siguiente imagen se detallan todos los componentes electrónicos que conforman el **Control Centralizado**, así como su conexionado y alimentación correspondientes. Destacar que la alimentación de 12VDC sólo se aplica a los actuadores de las compuertas motorizadas, pero no interviene realmente en el circuito eléctrico. Este se alimenta externamente con 5VDC, y luego se hace uso de la tensión de 3,3VDC del regulador de voltaje interno del NodeMCU (U2) para alimentar a la interfaz MCP23017 (U1).

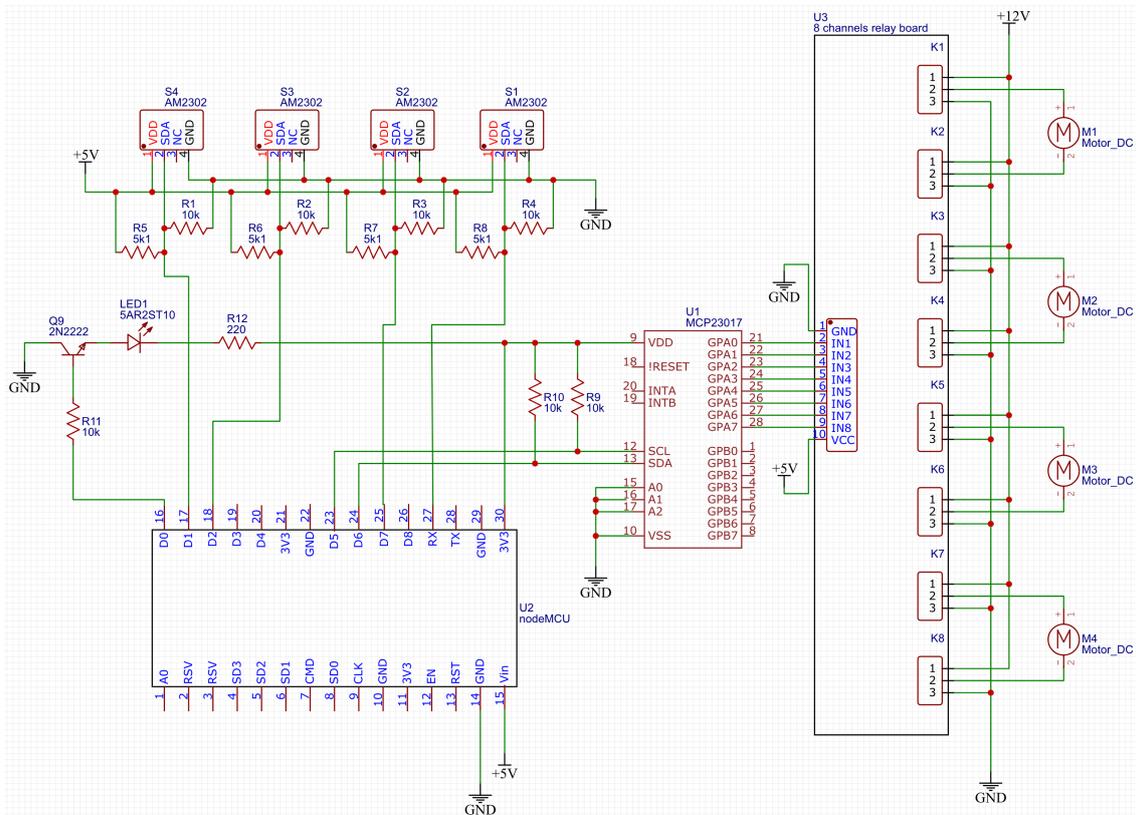


Imagen 17: Esquema Eléctrico

En un primer diseño, no se incluyeron las resistencias R1 a R8, ya que los sensores S1 a S4 estaban alimentados con una tensión de 3,3VDC. Una vez realizada la instalación final, ha sido necesario alimentarlos con un voltaje de 5VDC e incluir las mencionadas resistencias para reducir la tensión (divisor de voltaje) en el pin de datos, ya que se obtenían multitud de lecturas erráticas debido a la longitud de los cables y la falta de alimentación que esta causaba.

4.2. Esquema general de *hardware*

Siguiendo el esquema eléctrico, a continuación se muestra cómo son las conexiones de un prototipo utilizando *hardware* real.

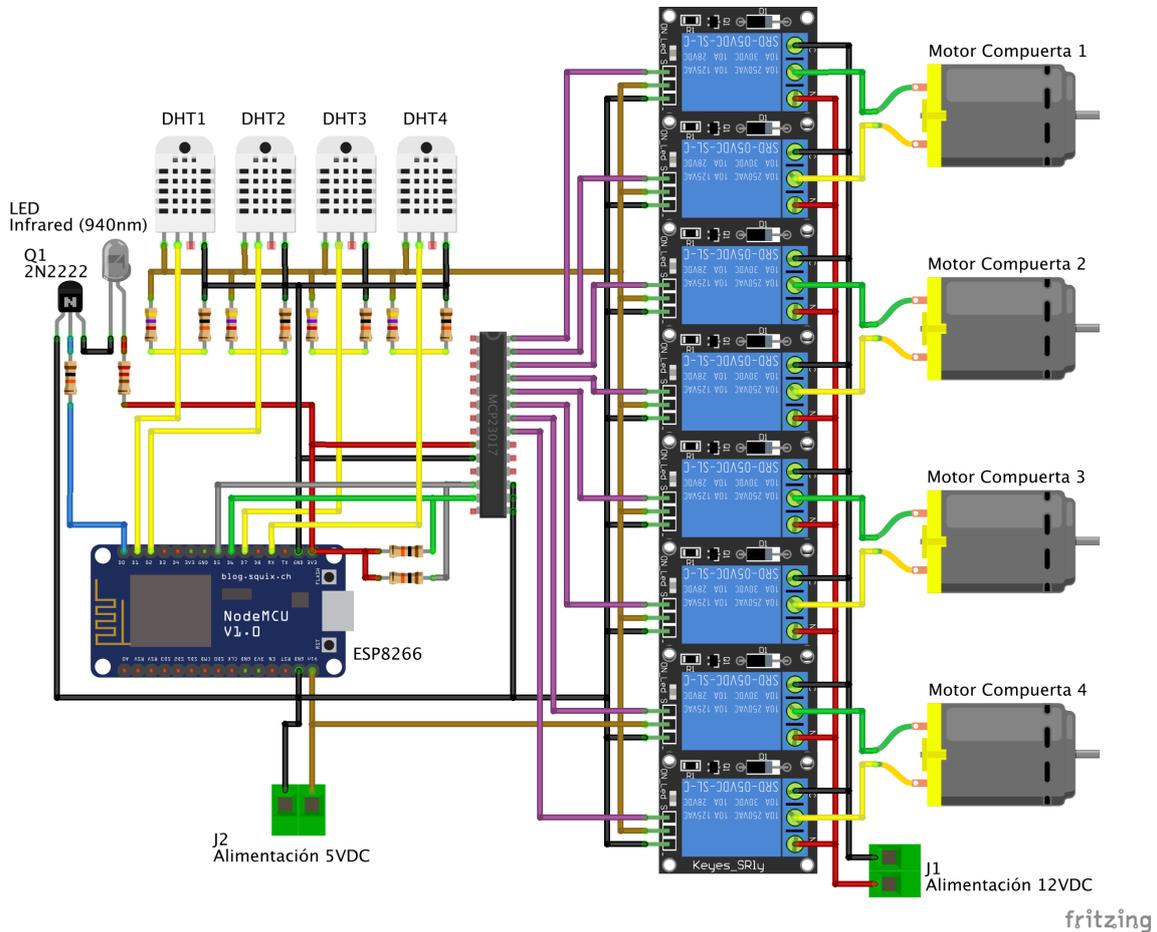


Imagen 18: Esquema de conexiones de los diferentes componentes

4.3. Diseño de la Arquitectura de *Software*

Seguidamente, se presenta el diagrama de bloques del sistema que muestra conceptualmente su funcionamiento:

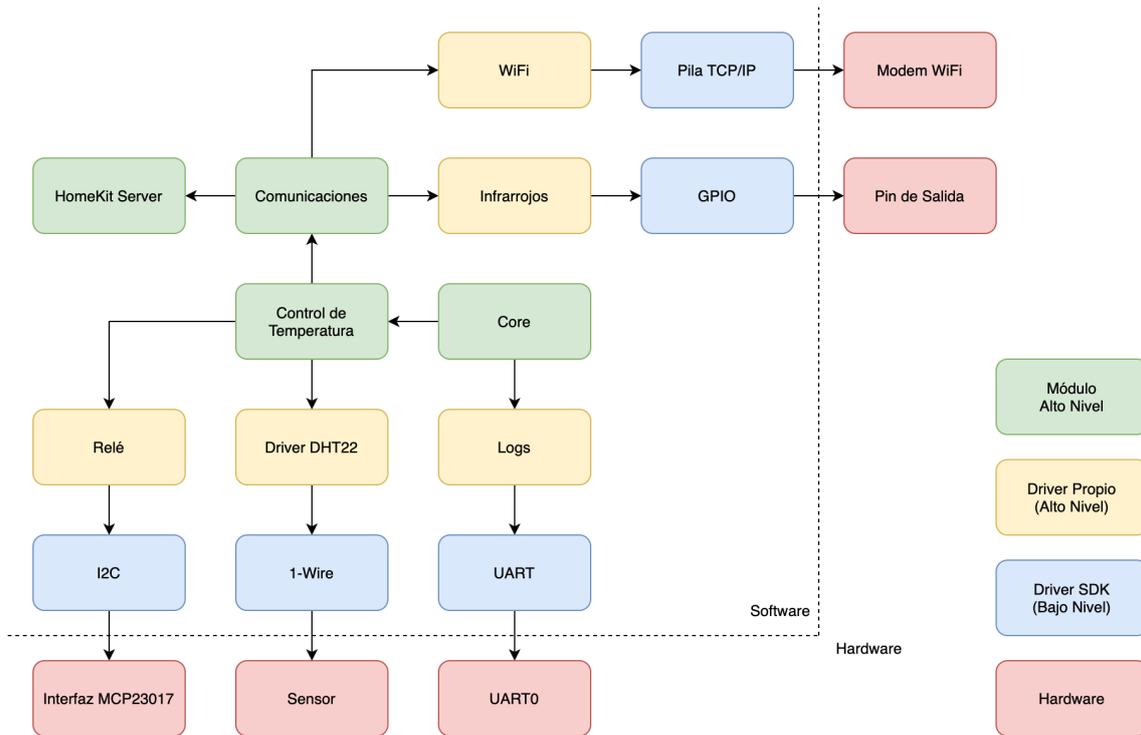


Imagen 19: Diagrama de bloques del Software

El siguiente diagrama de flujo muestra, a grandes rasgos, el funcionamiento del *software* del microcontrolador con los principales casos de uso:

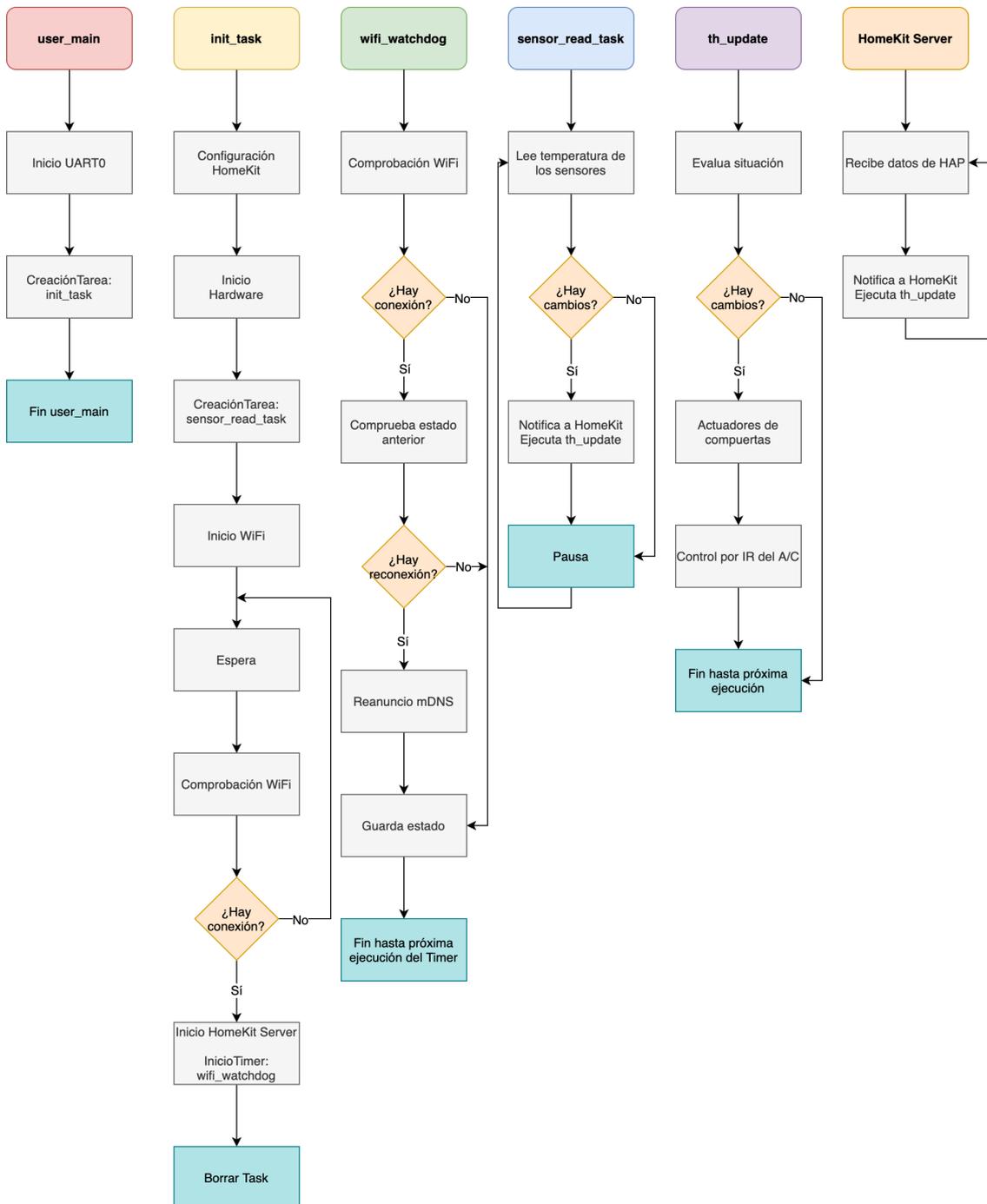


Imagen 20: Diagrama de flujo

4.4. Partes principales del desarrollo

Para una mejor comprensión del funcionamiento del *software* del control centralizado, esta sección se ha estructurado en diferentes partes:

A. *Watchdogs* del sistema

Esta parte no ha sido desarrollada en el ámbito de este proyecto, pero sin embargo hay que destacarla, puesto que en el desarrollo se ha tenido muy en consideración. Existen dos *watchdogs* en el sistema, encargados de reiniciar el dispositivo en caso de fallo. Se trata de uno por *hardware*, y otro por *software*. En esencia, son dos temporizadores o *timers* que necesitan ser rearmados (lo que comúnmente se conoce como "alimentar al perro") o de lo contrario ejecutarán un reinicio del sistema. El *watchdog* implementado por *software* tiene un tiempo máximo de espera de 1 segundo, y su rearme forma parte de una tarea del sistema a la que se le llama cuando se tiene ocasión (*taskYIELD* o *vTaskDelay*). El *watchdog* por *hardware* está implementado a través de un contador interno, y este ejecutará el reinicio cuando hayan pasado 6 segundos sin que el *software* "lo alimente" a través de un señalizador concreto.

De esta manera, si ocurriera cualquier fallo en el *software* y este se quedara bloqueado, el sistema se reiniciaría automáticamente. Esto ofrece un buen mecanismo de seguridad, ya que el *software* establece un punto de apagado total al iniciarse. Es decir, cuando el dispositivo inicia, asume que las compuertas están cerradas y que la máquina de A/C está encendida, independientemente de su estado real, con lo que este ordena abrir todas las compuertas motorizadas y apagar la máquina de A/C.

B. HomeKit

Para una correcta implementación en el entorno domótico de Apple, ha sido necesaria la correcta definición de los accesorios, servicios y características HomeKit. Existe una jerarquía que hay que mantener, y que se estructura de la siguiente manera:

- Dispositivo.
 - Accesorio.
 - Servicio.
 - Característica.

Así, el dispositivo podrá tener uno o más accesorios, cada accesorio podrá tener uno o más servicios, y cada servicio podrá tener una o más características.

En el caso del desarrollo de este proyecto, esta jerarquía ha sido construida con la siguiente estructura concreta:

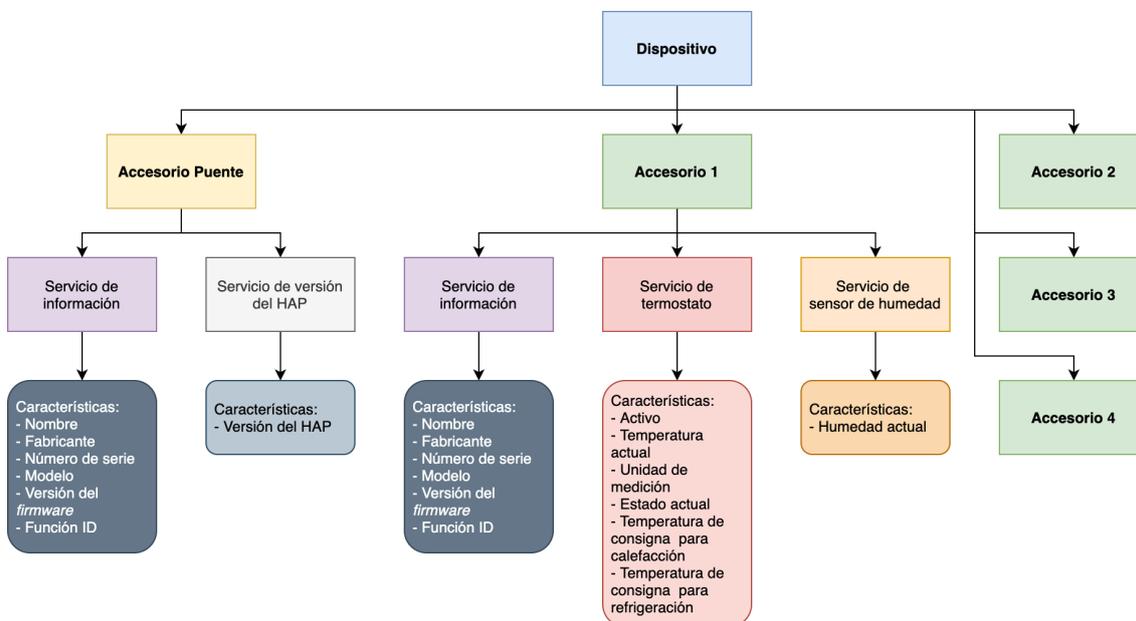


Imagen 21: Jerarquía HomeKit

Los accesórios 2, 3 y 4 tienen la misma estructura y componentes que el accesorio 1. El accesorio puente simplemente sirve para proveer al ecosistema HomeKit de la información básica de funcionamiento. Este accesorio no es visible para el usuario, pero es el encargado del proceso de emparejamiento de HomeKit, que para este caso hay que utilizar el código 021-82-017 en la configuración de la aplicación Casa de Apple, tal y como se ve en la siguiente imagen.

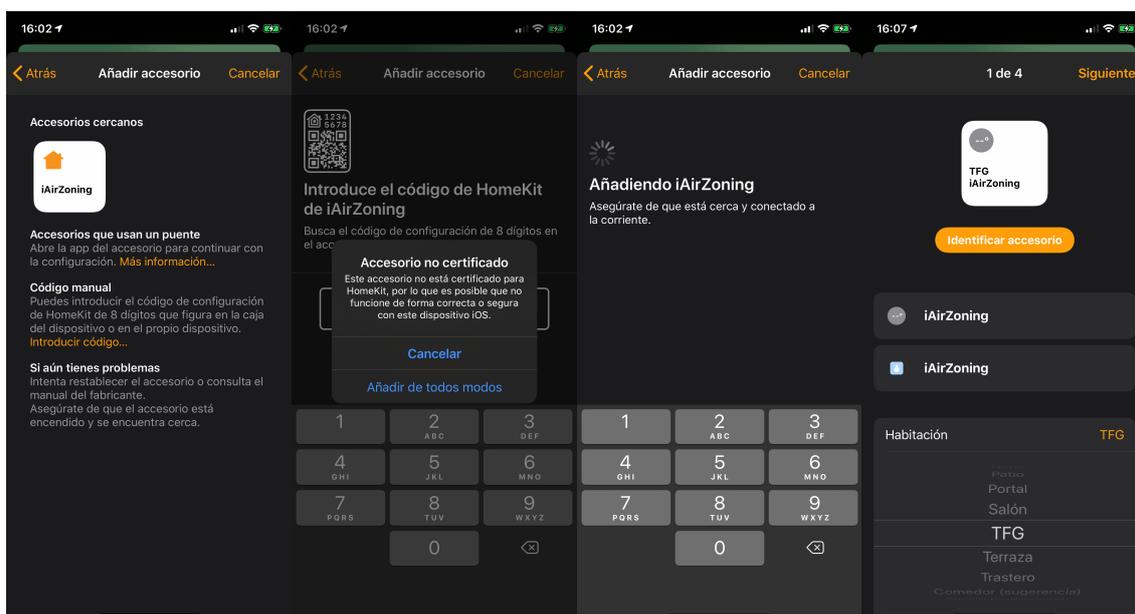


Imagen 22: Pasos para añadir el accesorio a HomeKit

Los accesorios 1 al 4 son los que crean la interfaz de usuario, y tanto el servicio termostato como el servicio de sensor de humedad disponen de una serie de características que el sistema tiene que controlar y mantener actualizadas. En esta imagen se puede ver la apariencia que tiene la interfaz de control de cada uno de estos servicios en la aplicación Casa:

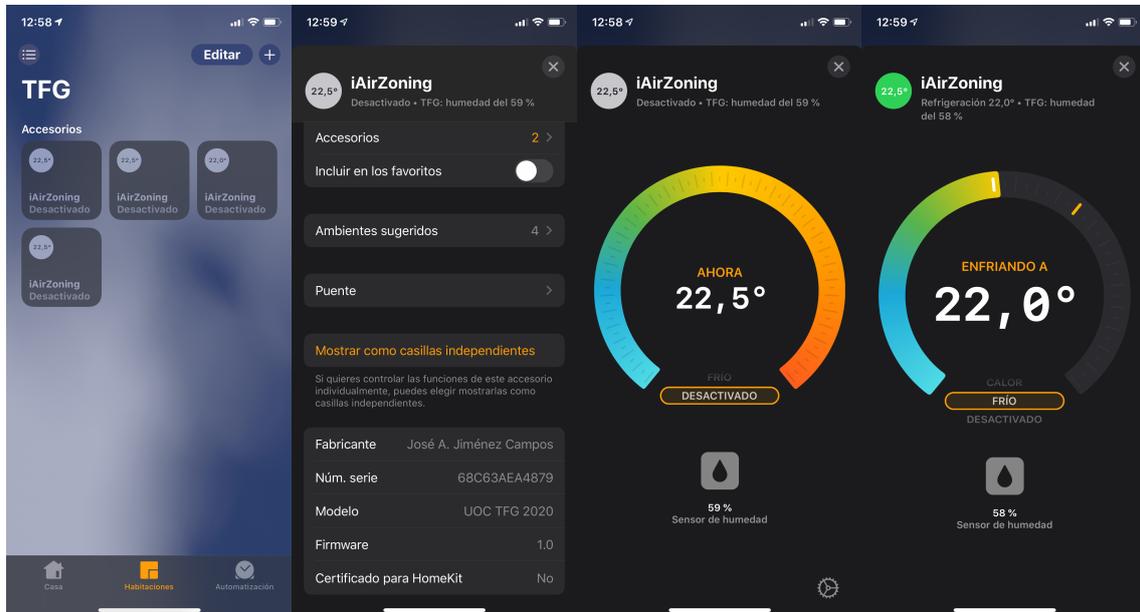


Imagen 23: iAirZoning en HomeKit

C. Lógica de control

Cada vez que el sistema recibe una orden a través de HAP, o hace una lectura de temperatura de los sensores, la lógica de control se pone en marcha para determinar si es necesario realizar algún cambio. Estos vienen determinados por el estado de cada uno de los termostatos, y se calcula en base a la siguiente figura:

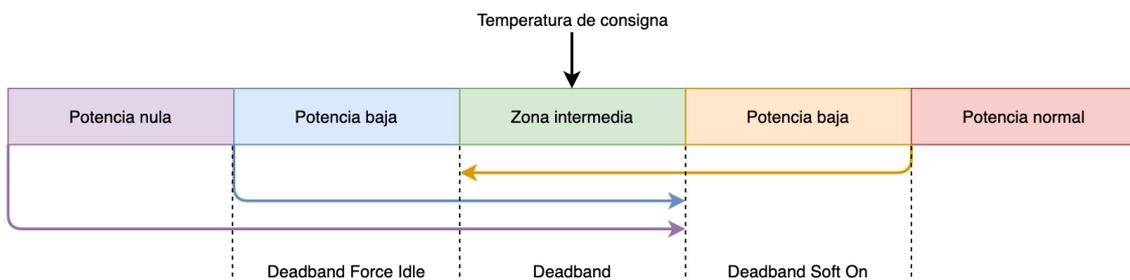


Imagen 24: Diagrama de zonas del termostato

En función de la temperatura de consigna y su comparación con la temperatura actual, se determina la zona de trabajo (potencia) en la que se encuentra el termostato, siempre que esté en funcionamiento, ya que de lo contrario el estado será de apagado.

Una vez que han sido evaluados todos los termostatos y se ha definido cuál debe ser su estado (su zona de trabajo), se aplica la lógica de control general para determinar el estado de cada una de las compuertas, y de la máquina de A/C.

Es importante tener en cuenta los casos imposibles, determinados por la imposibilidad de que la máquina de A/C no puede enfriar y calentar a la vez. Una vez que se enciende un termostato, su modo de funcionamiento (frío o calor) será el que condicione el modo de funcionamiento el resto hasta que todos se apaguen. Así, si un termostato está funcionando en modo frío, al tratar de encender otro termostato en modo calor, el sistema apagará este último automáticamente.

Además, la potencia baja sólo se aplicará para el caso en los que todos los termostatos encendidos se encuentren en esa zona de trabajo, de lo contrario, se usará la potencia normal, o nula, según se requiera.

Para evitar problemas en la sincronización de los datos enviados por HAP y no permitir que el sistema trabaje innecesariamente, se ha establecido un retardo de 5 segundos en la aplicación de la lógica, una vez que el sistema ha dejado de recibir órdenes. De esta manera, se evitan situaciones de riesgo, como el abrir o cerrar compuertas que ya estén en movimiento.

D. Control de las compuertas

Cada vez que es necesario abrir o cerrar compuertas, el sistema envía las ordenes correspondientes al grupo de relés. Estos, en parejas, son usados para controlar los motores DC de las compuertas. El estado actual de cada compuerta es guardado en memoria, y sólo se actúa sobre una de estas si realmente hay que cambiar su estado. Por ejemplo, una compuerta que ya esté abierta no recibirá energía de los relés para volverse a abrir en el caso de que se envíe nuevamente la correspondiente orden de apertura.

E. Control de la máquina de A/C

La unidad principal de refrigeración y calefacción es controlada mediante el envío de códigos IrDA. Estos códigos se mandan cada vez que hay que realizar algún cambio en el modo de operación, en base a la lógica de control. Para el desarrollo de este proyecto, se han definido los siguientes códigos infrarrojos para el control de la máquina de A/C:

Potencia	Modo	Ventilador	Temperatura
Apagado	Apagado	-	-
Normal	Calefacción	2	28 °C
Baja	Calefacción	1	28 °C
Nula	Calefacción	1	17 °C
Normal	Refrigeración	2	19 °C
Baja	Refrigeración	1	19 °C
Nula	Refrigeración	1	30 °C

Tabla 4: Comandos IrDA

La intención de utilizar estos niveles de potencia es aprovechar el funcionamiento *inverter*²⁰ de la máquina de climatización.

F. Watchdog WiFi

Si el sistema pierde la conectividad de red, automáticamente escaneará en busca de redes hasta encontrar la suya, y tratará de volver a conectarse a ella. Esto ya viene implementado en el SDK, sin embargo, una reconexión puede tener consecuencias inesperadas, y hacer que el sistema no sea accesible desde la interfaz de usuario, puesto que HomeKit utiliza el protocolo mDNS²¹ para anunciar sus servicios a la red. Así que, además de los *watchdogs* del sistema, se ha añadido otro que monitoriza el estado de la conexión para que, en caso de reconexión, vuelva a emitir el anuncio mDNS.

²⁰ https://es.wikipedia.org/wiki/Inversor_compresor (Junio de 2020)

²¹ https://en.wikipedia.org/wiki/Multicast_DNS (Junio de 2020)

5. Comparación de funcionamiento antes y después

En este punto, se presentan las diferencias de funcionamiento usando el sistema desarrollado y sin usar, haciendo una comparativa de la evolución de la temperatura en una de las habitaciones. La temperatura de consigna se ha establecido a 25°C en el modo de refrigeración, y se ha partido de una temperatura inicial de 26,9°C. En la siguiente gráfica se muestran los resultados obtenidos:

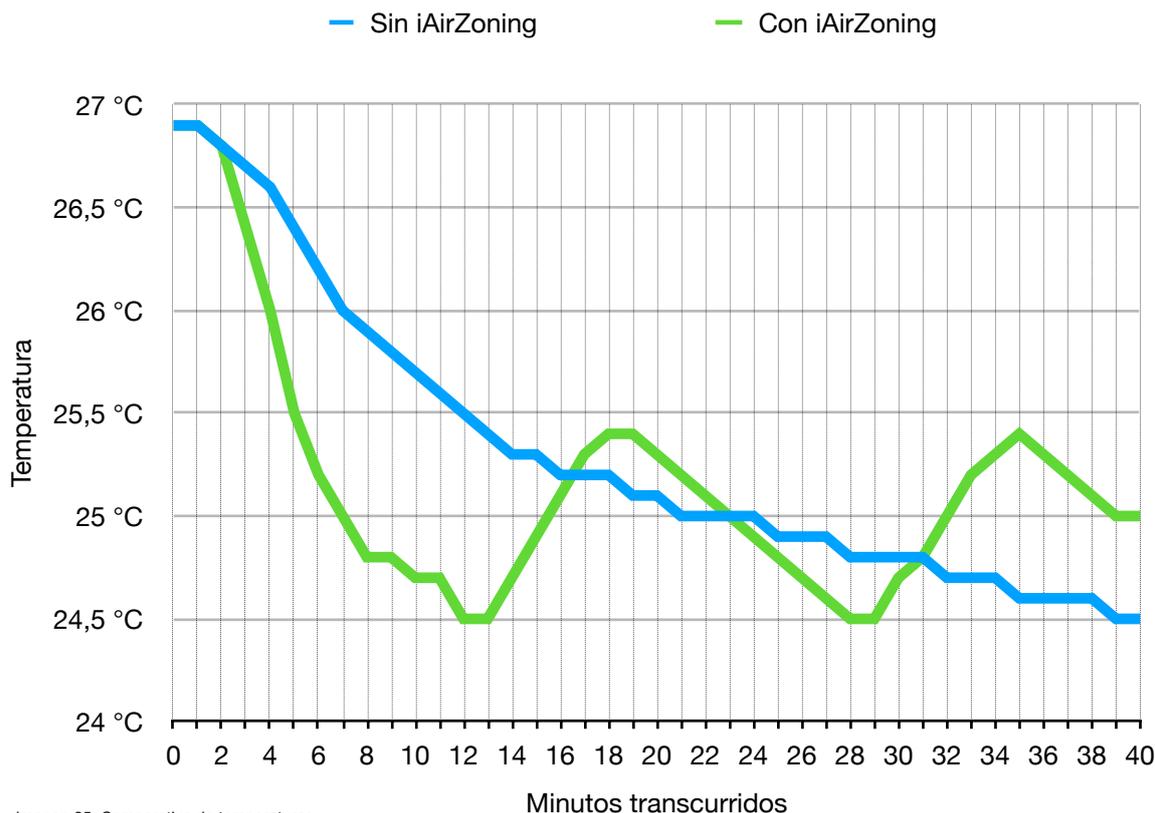


Imagen 25: Comparativa de temperaturas

Se puede observar que la evolución de la temperatura utilizando el sistema desarrollado es más pronunciada, llegando a la temperatura de consigna rápidamente, ya que sólo actúa sobre una de las 4 estancias, y manteniendo su temperatura ambiente entre 24,5°C y 25,5°C. Sin embargo, utilizando el sistema convencional sin iAirZoning, la temperatura baja más lentamente, ya que tiene que refrigerar las 4 estancias. Además, aunque no se muestra en la gráfica, la temperatura siguió descendiendo hasta los 23,4°C, momento en el que la máquina de A/C cambió de régimen de funcionamiento para dejar de enfriar.

Usando el sistema iAirZoning, la temperatura se mantiene de forma más agradable, ya que mantiene esta con 0,5°C de margen, además de que el ventilador reduce su velocidad, generando menos ruido; detalle que no ocurre cuando no se utiliza el sistema desarrollado, y al

final es el usuario el que tiene que estar constantemente regulando el A/C manualmente con el control remoto.

6. Monitorización del consumo eléctrico

Uno de los principal objetivos de este proyecto es la reducción del consumo eléctrico que supone la zonificación de una instalación de aire acondicionado por conductos para más de una estancia. Para poder llevar a cabo un estudio más detallado acerca del consumo eléctrico de la máquina de A/C, se ha optado por la adquisición de un medidor comercial de consumo Shelly EM²².

Este aparato permite realizar mediciones tanto en la tensión de entrada como en la corriente que circula por la fase de la línea de alimentación de la máquina de A/C. Además, todas las mediciones se guardan en un registro histórico para poder realizar una comparativa después de varios días de uso.

6.1. Instalación

La instalación del monitor de consumo ha de hacerse en el cuadro eléctrico general. Puesto que este sistema en particular es capaz de monitorizar 2 líneas de forma independiente, se ha optado por monitorizar el consumo total de la vivienda, además de la máquina de A/C de la planta alta, que es la involucrada en el desarrollo del presente proyecto.

A continuación, se observa el detalle del monitor de consumo (derecha) junto a una de las "pinzas" (izquierda) encargadas de medir el paso de la corriente de una de las fases:



Imagen 26: Sistema de monitor de consumo

²² <https://shelly.cloud/shelly-energy-meter-with-contactor-control-wifi-smart-home-automation/> (Abril de 2020)

Una vez instalado en el cuadro eléctrico, el sistema ya está listo para empezar a monitorizar el consumo. La instalación de la "pinza" se ha realizado sobre la línea que alimenta tanto a la máquina interior, como a la unidad exterior del sistema de A/C de la planta alta.

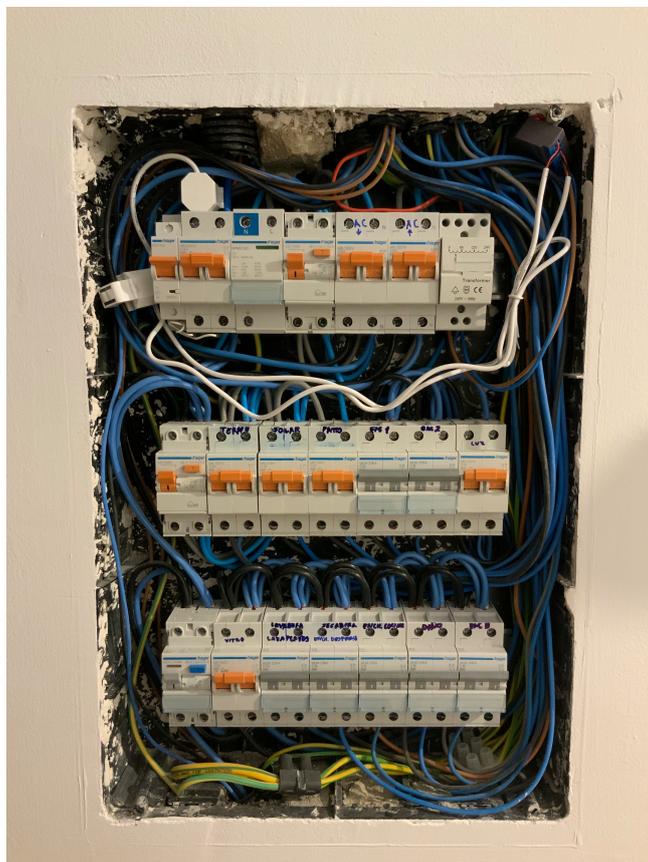


Imagen 27: Sistema de monitor de consumo instalado

Después de realizar una cuantas pruebas, se observa que las lecturas de consumo son fiables y acordes al consumo de la máquina de A/C (2,02kW), según las especificaciones del fabricante para la máquina Mitsubishi Electric MSPEZ-71VJA²³.

²³ <https://www.mitsubishielectric.es/aire-acondicionado/series/conductos-pead/#especificaciones> (Abril de 2020)

6.2. Consumo eléctrico antes y después

Una vez instalado el sistema desarrollado, se han realizado mediciones de consumo energético tanto con el nuevo sistema funcionando, como sin este. Seguidamente, se muestra una gráfica donde se puede apreciar la gran diferencia de consumo eléctrico entre ambos.

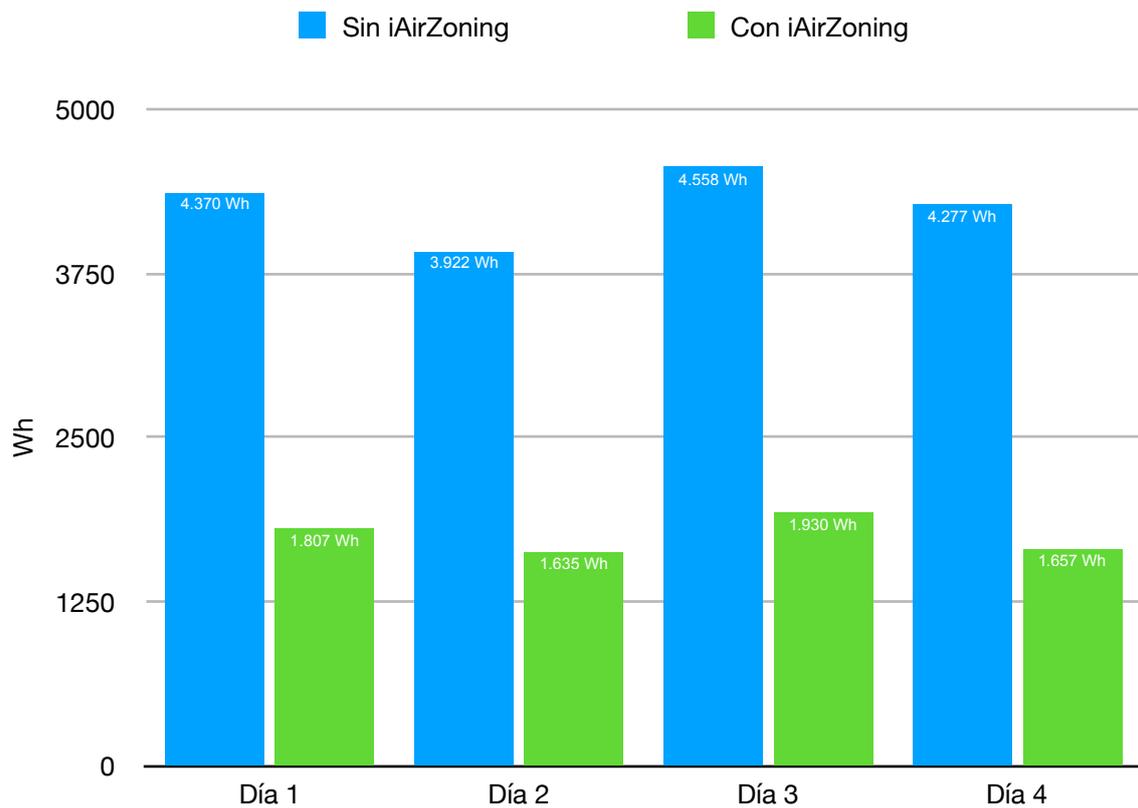


Imagen 28: Comparativa de consumo

Las mediciones se han realizado en días en los que la temperatura exterior ha sido similar, usando como temperatura de consigna 24°C, y manteniendo el sistema en funcionamiento durante 3 horas. Para el caso de las pruebas con iAirZoning, se ha activado en una de las 4 habitaciones, puesto que ese es el uso habitual.

En la gráfica, se puede observar que la reducción de consumo energético es considerable.

7. Viabilidad técnica

El sistema desarrollado finalmente se puede considerar viable desde un punto de vista técnico. Su capacidad funcional ha sido probada durante varios días, con unos resultados que podrían equipararse a los de cualquier producto comercial.

A continuación, se detallan sus puntos fuertes:

- **Fácil alimentación:** el control centralizado sólo necesita una pequeña fuente de alimentación microUSB (5VDC) de al menos 1A de corriente.
- **Sensores fácilmente reemplazables:** Los sensores de temperatura usados son fácilmente reemplazables en caso de avería.
- **Apagado automático en caso de fallo:** El sistema abre todas las compuertas y apaga la máquina de A/C automáticamente en caso de que alguno de los sensores dé más de 3 lecturas erróneas seguidas. En caso de que el sistema falle y se reinicie, el resultado también es la apertura de compuertas y el apagado del A/C.
- **Sobrecontrol de emergencia:** Es posible manejar la máquina de /AC usando su mando original.
- **Interfaz de usuario:** Cualquier cliente HomeKit autorizado es capaz de controlar el sistema, tanto local como remotamente.
- **Seguridad en las comunicaciones:** Gracias al uso de HAP, todas las comunicaciones WiFi van cifradas y autenticadas, de forma que un cliente no autenticado no puede dar ni

Id	Descripción	Prob.	Imp.	Total	Acciones	Conclusión
1	Avería en el <i>hardware</i> del prototipo	1	9	9	Usar pieza de reemplazo	Hay que tener repuestos de todo
2	Funcionamiento anómalo de los sensores de temperatura debido a la distancia	1	9	9	Alimentarlos a 5V y usar un divisor de voltaje (2 resistencias) en el pin de datos	Retrasará ligeramente la planificación
3	Avería del ordenador de trabajo	1	9	9	Usar otro	Bajo control
4	Pérdida de datos del proyecto	1	9	9	Disponer de múltiples copias de seguridad en línea y fuera de línea	Bajo control

Tabla 5: Evaluación de Riesgos

recibir órdenes del control centralizado.

Entre los puntos débiles, se podrían destacar:

- **Ecosistema domótico cerrado:** Al usar HAP, el usuario no puede interactuar con el

Impacto en el proyecto	9	9	18	27	36	45
	7	7	14	21	28	35
	5	5	10	15	20	25
	3	3	6	9	12	15
	1	1	2	3	4	5
		1	2	3	4	5
Probabilidad de riesgo						

Valor	Probabilidad	Impacto	Valor		
1	Improbable	Muy bajo	1		
2	Extraño	Bajo	3		
3	A veces	Moderado	5	Riesgo alto	22 - 45
4	A menudo	Alto	7	Riesgo moderado	11 - 21
5	Frecuente	Muy alto	9	Riesgo bajo	1 - 10

Tabla 6: Leyenda de la evaluación de riesgos

sistema de ninguna otra forma que no sea mediante HomeKit.

- **Dependiente de una conexión WiFi:** Es necesario que, tanto el control centralizado, como los clientes HomeKit, estén conectados a la misma red WiFi para poder interactuar con el sistema. De todas formas, en caso de que la conexión WiFi dejara de estar disponible, es posible crear un punto de acceso con los mismos datos de conexión (SSID y contraseña) usando un iPhone o un iPad, al que el control centralizado pueda conectarse y así interactuar con él.

7.1. Valoración de riesgos

Puesto que la realización de este proyecto implica una instalación real de *hardware*, todas las tecnologías que han sido utilizadas ya han sido probadas previamente. Si bien es cierto que existen ciertos riesgos, como la elevada longitud de los cables que conectarán con los sensores de temperatura (unos 5 metros), en principio no debería haber problema según las especificaciones del fabricante, que otorga hasta un máximo de 20 metros de longitud de cable siempre que la alimentación sea de 5VDC.

Los posibles problemas que pueden surgir son principalmente aquellos derivados del *hardware* y un mal funcionamiento, normalmente producidos por averías, como puede ser una conexión errónea que estropea el dispositivo y hay que esperar al reemplazo. Es por ello que se ha decidido disponer de todo el material por duplicado, excepto de las compuertas motorizadas,

Descripción	Cantidad	Precio	Total
Placa NodeMCU Lolin v3	1	15,00 €	15,00 €
Sensor de temperatura y humedad DHT22	4	10,00 €	40,00 €
Placa de 8 relés libres	1	12,00 €	12,00 €
Interfaz MCP23017	1	6,00 €	6,00 €
LED IrDA de alta potencia	1	2,00 €	2,00 €
Compuerta motorizada	4	130,00 €	520,00 €
Fuente de alimentación 12VCD de 100W	1	16,00 €	16,00 €
Fuente de alimentación microUSB	1	15,00 €	15,00 €
Componentes electrónicos de bajo coste	1	10,00 €	10,00 €
Desarrollo del sistema	160	15,00 €	2.400,00 €
Montaje e instalación	45	8,00 €	360,00 €
		TOTAL	3396,00 €

Tabla 7: Valoración económica

dado su elevado coste (130€ por unidad). En este último caso se cuenta con una de reserva.



Imagen 29: Placa de desarrollo ESPDuino

Además, para las tareas iniciales de desarrollo, no se ha utilizado la placa controlador definitiva, sino que se ha hecho uso de una placa ESPDuino de DOIT²⁴, la cual utiliza el mismo chip controlador ESP8266. De esta forma se preserva el *hardware* definitivo del proyecto hasta que sea necesaria su utilización.

²⁴ <http://doit.am> (Abril de 2020)

A continuación se muestra la leyenda correspondiente a la tabla de evaluación de riesgos:



8. Valoración económica

Seguidamente se expone una tabla donde se refleja el coste que ha supuesto el desarrollo del sistema iAirZoning para una zonificación de 4 estancias.

El presupuesto de mantenimiento no se considera relevante para este proyecto, puesto que el sistema ha sido diseñado para no necesitar ningún tipo de mantenimiento. Aunque es posible que requiera de alguna reparación puntual, pero la cuantía de esta vendrá determinada por el número de horas y los materiales necesarios para llevarla a cabo. No es lo mismo reemplazar el servo averiado de una compuerta, que sustituir el LED emisor de infrarrojos.

8.1. Materiales utilizados

En la siguiente tabla se presentan los materiales utilizados para la construcción del *hardware* prototipo del proyecto. Además, ha sido necesario cableado variado, resistencias,

Descripción	Cantidad	Precio	Total
Microcontrolador ESP8266EX	1	5,00 €	5,00 €
PCB	1	1,00 €	1,00 €
Relé	8	0,75 €	6,00 €
Interfaz MCP23017	1	3,00 €	3,00 €
LED IrDA de alta potencia	1	1,00 €	1,00 €
Conectores varios	1	1,00 €	1,00 €
Componentes de la fuente de alimentación	1	5,00 €	5,00 €
Componentes electrónicos de bajo coste	1	5,00 €	5,00 €
Sensor de temperatura DHT22	4	7,00 €	28,00 €
Carcasa de ABS	1	1,00 €	1,00 €
TOTAL			56,00 €

Tabla 9: Costes de industrialización

condensadores, conectores, placas prototipo, etc.

Descripción

Imagen

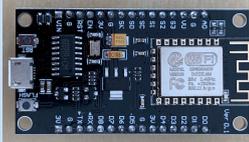
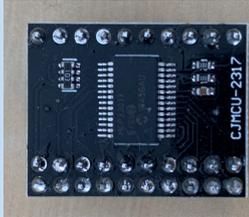
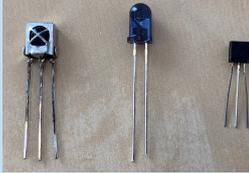
<p>Placa NodeMCU Lolin v3</p> <ul style="list-style-type: none"> • Integra el microcontrolador ESP8266. 	
<p>Sensor de temperatura DHT22</p> <ul style="list-style-type: none"> • Usa el protocolo 1-Wire de Adafruit Industries, soportado por esp-open-rtos. 	
<p>Placa de 8 relés libres</p> <ul style="list-style-type: none"> • Cada compuerta necesita 2 relés de este tipo para hacer la función de un <i>H-Bridge</i> para el control del motor DC. 	
<p>Interfaz de ampliación de Entrada/Salida MCP23017</p> <ul style="list-style-type: none"> • Usa el interfaz de comunicaciones I²C utilizando 2 GPIO del ESP8266. • Necesaria para poder controlar los 8 relés, ya que el ESP8266 no dispone de tantos GPIO libres. • El SDK cuenta con soporte por software para I²C usando 2 GPIO, pero ha sido necesario realizar el desarrollo del driver del MCP23017 para esp-open-rtos. 	
<p>LED IR de alta potencia, transistor 2N2222 y receptor IR a 38KHz</p> <ul style="list-style-type: none"> • Junto con un par de resistencias, sirven para capturar y enviar los códigos IrDA necesarios para controlar la unidad central de A/C. 	
<p>Compuerta motorizada rectangular de rejilla AirZone</p> <ul style="list-style-type: none"> • Funcionan a 12VDC: +12V para abrirla y -12V para cerrarla. 	
<p>Fuente de alimentación 240VAC - 12VDC de 100W</p> <ul style="list-style-type: none"> • Para alimentar los motores de las compuertas. 	

Tabla 8: Material necesario para la realización del proyecto

8.2. Industrialización

Para el cálculo económico de lo que sería un proceso de industrialización, se han estimado ciertos costes de producción en lo relativo a la fabricación del control centralizado, puesto que las compuertas motorizadas son ya un producto comercial, y la necesidad del número de estas vendrá determinado por la cantidad de zonas a configurar.

Hay que tener en cuenta que el desarrollo del prototipo se ha hecho en base a placas ya construidas y comercializadas, pero para un proceso de industrialización, se deberá tener en cuenta los componentes electrónicos utilizados por estas, así como la producción de una PCB diseñada para el sistema en cuestión. A todo esto habrá que añadir el diseño externo del producto acabado, y las certificaciones necesarias para su comercialización (CE²⁵, MFi²⁶...).

A continuación se muestra una tabla con una estimación de los posibles costes de producción por unidad para la fabricación de 1.000 unidades del control centralizado para 4 zonas:

Teniendo un coste de 56,00 € por unidad, el total para fabricar 1.000 unidades ascendería a 56.000,00 €. A esta cuantía habría que añadir el coste de las diferentes certificaciones, que variarán en función del mercado donde se comercializará el producto, así como el cumplimiento de la legislación que esté vigente en los países donde esté previsto su uso.

Es importante destacar el hecho de que esta industrialización es un caso hipotético que no puede llevarse a cabo, puesto que este proyecto ha sido desarrollado en base a las especificaciones no comerciales de HAP, las cuales prohíben explícitamente cualquier forma de comercialización y distribución. Apple Inc. indica que si se desea desarrollar, producir y comercializar productos HomeKit, es necesario estar afiliado a su programa MFi (Made For iPhone), y pagar las correspondientes tasas.

En el caso de que realmente se quisiera industrializar este producto, habría que utilizar el microcontrolador ESP32, y realizar algunos cambios en el código para adaptarlo a la librería HomeKit certificada por Apple, a la cual sólo se puede tener acceso si se forma parte del programa MFi. Siendo así, sería totalmente factible producir y comercializar el producto, y contaría además con la certificación HomeKit de Apple. En este caso, el coste de producción sería similar, ya que aunque el microcontrolador ESP32 tiene un precio un poco más elevado

²⁵ https://europa.eu/youreurope/business/product-requirements/labels-markings/ce-marking/index_es.htm (Junio de 2020)

²⁶ <https://developer.apple.com/programs/mfi/> (Junio de 2020)

que el ESP8266EX, no sería necesaria la utilización de la interfaz de expansión MCP23017, puesto que el ESP32 dispone de GPIO libres suficientes para controlar los relés directamente.

9. Conclusiones

Como conclusión principal, cabría destacar que a la finalización del proyecto, todos los objetivos propuestos en la etapa inicial de este se han podido cumplir:

- Ahorro energético mediante zonificación de la climatización.
- Integración domótica en el ecosistema Apple HomeKit.
- Monitorización y presentación de la temperatura y humedad de cada estancia a través de las interfaces de Apple HomeKit.

9.1. Descripción

El desarrollo de este proyecto ha permitido profundizar en los sistemas empotrados de una manera bastante grande. Su grado de dificultad, que se podría valorar de alto, ha estado condicionado por los muchos y diferentes factores que han atañido tanto al diseño y construcción del *hardware*, como al desarrollo del *software*. Interactuar con productos comerciales, de los cuales algunos tienen un elevado coste (máquina de A/C y compuertas motorizadas) hace que haya que pensar y decidir muy bien la manera de proceder, dejando a un lado "técnicas" basadas en ensayo y error.

Ha sido una carrera de fondo, donde se han aplicado conocimientos ya aprendidos, y también se han adquirido otros, tales como una mejor forma de gestionar la memoria RAM de un sistema empotrado; algo muy valioso, dado que normalmente este tipo de *hardware* dispone de una cantidad de RAM muy limitada.

9.2. Autoevaluación

a) Tiempo dedicado a cada PEC

En la siguiente tabla se muestran, tanto el número de horas planificado para cada parte del desarrollo del proyecto, como las horas reales que finalmente ha supuesto cada una.

HORAS	PEC0	Propuesta	PEC1	PEC2	PEC3	Entrega Final	Memoria	Total
Planificado	0	20	25	90	70	65	60	330
Real	0	20	25	86	67	76	58	332

Tabla 10: Dedicación en horas

b) Reflexiones sobre los objetivos alcanzados

Todos los objetivos se han alcanzado con un elevado grado de satisfacción. Después de algunas semanas de uso, se puede considerar como un éxito total, puesto que el gasto

energético se ha reducido considerablemente, y las opciones de confort se han ampliado, gracias a la capacidad del sistema de poder seleccionar diferentes temperaturas de consigna para cada estancia, así como todas las posibilidades que aporta su gestión mediante HomeKit.

c) Análisis de la planificación y el seguimiento

A lo largo de todo el desarrollo del proyecto, se ha intentado seguir la planificación inicial, con especial énfasis en el cronograma, sin importar el hecho de que en algún día concreto haya sido necesario dedicar un mayor número de horas de las previstas. La intención de esta forma de trabajar ha sido el poder mantener la planificación del trabajo diario (cronograma).

Esta metodología ha supuesto cargas elevadas de trabajo durante más de una jornada, sobre todo para las tareas de instalación y montaje; sin embargo, la evolución del proyecto ha seguido la planificación inicial, puesto que un retraso en las fechas de finalización de los hitos hubiese tenido muy malas consecuencias, tales como no poder realizar las pruebas necesarias, o incluso la no terminación de la instalación, lo que hubiese supuesto abandonar el proyecto por no poder presentar ningún tipo de resultado.

9.3. Líneas de trabajo futuro

A pesar de que en este trabajo no ha quedado pendiente ningún desarrollo, este proyecto tiene mucho potencial de ampliación, pudiéndose destacar las siguientes líneas de trabajo futuro a explorar:

- Crear una interfaz de configuración donde establecer los parámetros de la configuración de red, así como las características de las habitaciones, conexión de los sensores, etc. De esta manera se universalizaría el *firmware* para su distribución en masa sin necesidad de compilar una versión diferente para cada instalación.
- Adaptar el código a las librerías comerciales de Apple HomeKit para ser usado en un microcontrolador ESP32 de Espressif, haciendo uso de su HomeKit SDK²⁷. Esto permitiría desarrollar un producto HomeKit certificado por Apple, y poder así distribuirlo y venderlo bajo la licencia MFi.
- Añadir la posibilidad de tener los sensores conectados a microcontroladores independientes, y que estos envíen la información correspondiente al control centralizado. De esta forma se evitaría tener que usar cableado para el conexionado de los sensores, y permitiría también desarrollar la opción de integrar un control físico en cada estancia, con una pequeña pantalla informativa.

²⁷ <https://www.espressif.com/en/products/sdks/esp-homekit-sdk> (Junio de 2020)

10. Glosario

- **Firmware:** Parte lógica o de *software* que contiene todas las instrucciones a ejecutar por un microcontrolador.
- **FreeRTOS:** Sistema operativo en tiempo real para sistemas empotrados, gratuito y de código abierto.
- **GPIO:** *General Purpose Input Output*. Pin de un chip cuya función puede ser configurada a discreción del desarrollador, dependiendo de las necesidades del sistema.
- **HAP:** *HomeKit Accessory Protocol*. Protocolo de comunicaciones usado por los dispositivos HomeKit para el hogar.
- **HomeKit:** Sistema domótico desarrollado por Apple Inc. para el control del hogar.
- **I²C:** Bus de comunicaciones en serie, con capacidad de jerarquía, que hace uso de 2 pines para la comunicación (datos y reloj).
- **IrDA:** Estándar de comunicaciones mediante pulsos de luz infrarroja.
- **ISP:** *Internet Service Provider*. Proveedor de Servicios de Internet.
- **Microcontrolador:** Chip integrado, de reducido tamaño, con capacidad para ejecutar instrucciones.
- **Motor DC:** Motor de corriente continua.
- **PCB:** *Printed Circuit Board*. Placa de circuito impreso.
- **RF:** Radio frecuencia.
- **UART:** Bus de comunicaciones en serie, bidireccional, *full-duplex*, punto a punto, que hace uso de 2 pines para la comunicación (transmisión y recepción).
- **WatchDog:** Proceso, función o tarea encargado de la vigilancia y monitorización de una parte concreta del *software* o *hardware*, con capacidad para ejecutar acciones.
- **WiFi:** Tecnología de comunicación inalámbrica que soporta TCP/IP.

11. Bibliografía

- Espressif System. *ESP8266EX Datasheet*. Versión 6.4. Abril de 2020.
https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf
- Aosong Electronics Co., Ltd. *DHT22 Datasheet*. 2020.
<https://cdn-shop.adafruit.com/datasheets/DHT22.pdf>
- Microchip Technology Inc. *MCP23017/MCP23S17 Datasheet*. 2007.
<https://cdn-shop.adafruit.com/datasheets/mcp23017.pdf>
- AirZone. *Compuerta Motorizada Rectangular de Rejilla*. 2019.
https://doc.airzone.es/producto/actuales/Airzone/Difusion_motorizada/DM_CPRR_A4_ES.pdf
- Apple Inc. *HomeKit Accessory Protocol Specification. Non-Commercial Version*. Release R2. Julio de 2019. <https://developer.apple.com/homekit/specification/>