



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

– **TELECOM** ESCUELA
TÉCNICA **VLC** SUPERIOR
DE INGENIERÍA DE
TELECOMUNICACIÓN

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería de
Telecomunicación

SIMULACIÓN DE REDES IoT CON PACKET TRACER

Trabajo Fin de Grado

Grado en Ingeniería de Tecnologías y Servicios de
Telecomunicación

AUTOR/A: Sánchez Martínez, Anna

Tutor/a: Romero Martínez, José Oscar

CURSO ACADÉMICO: 2022/2023



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

— **TELECOM** ESCUELA
TÉCNICA **VLC** SUPERIOR
DE INGENIERÍA DE
TELECOMUNICACIÓN

PORTADA GENERADA AUTOMÁTICAMENTE EN EBRON

Escuela Técnica Superior de Ingeniería de Telecomunicación
Universitat Politècnica de València
Edificio 4D. Camino de Vera, s/n, 46022 Valencia
Tel. +34 96 387 71 90, ext. 77190
www.etsit.upv.es

VLC/
CAMPUS
VALENCIA, INTERNATIONAL
CAMPUS OF EXCELLENCE



SIMULACIÓN DE REDES IoT CON PACKET TRACER

El objetivo principal de este trabajo será explorar las capacidades de Packet Tracer como herramienta de simulación para redes IoT y demostrar su utilidad en diferentes escenarios.

En primer lugar, se presentará una visión general de los conceptos básicos de IoT, incluyendo definiciones, arquitectura y tecnologías involucradas. También se desarrollará la herramienta Packet Tracer utilizada, incluyendo sus características, funcionalidades y capacidades para la simulación de redes.

Además, se explicará la metodología para el diseño de redes IoT utilizando la versión 8 del Packet Tracer, teniendo en cuenta la selección de dispositivos, la configuración de parámetros de red y la integración de sensores y actuadores.

El enfoque principal consistirá en mostrar cómo simular redes IoT en Packet Tracer, incluyendo la configuración de escenarios, la generación de datos y la interacción entre dispositivos, y en analizar los resultados obtenidos de las simulaciones, evaluando el rendimiento de la red y la eficiencia de la comunicación.

Se presentarán diferentes casos de estudio donde se aplique la simulación en diferentes contextos y se resumirán las principales conclusiones, proporcionando recomendaciones para futuras investigaciones en el campo de la simulación de redes IoT.

Palabras clave: SIMULADOR / PACKET TRACER / IoT/ DISPOSITIVOS / SENSORES

SIMULACIÓ DE REDS IoT AMB PACKET TRACER

L'objectiu principal d'aquest treball serà explorar les capacitats de Packet Tracer com a ferramenta de simulació per a reds IoT i demostrar la seua utilitat en diferents escenaris.

En primer lloc, es presentarà una visió general dels conceptes bàsics de IoT, incloent definicions, arquitectura i tecnologies involucrades. També es desenvoluparà la ferramenta Packet Tracer utilitzada, incloent les seues característiques, funcionalitats i capacitats per a la simulació de reds IoT.

A més, s'explicarà la metodologia per al disseny de reds IoT utilitzant la versió 8 del Packet Tracer, tenint en compte la selecció de dispositius, la configuració de paràmetres de red i la integració de sensors i actuadors.

L'enfocament principal consistirà en mostrar com simular reds IoT en Packet Tracer, incloent la configuració d'escenaris, la generació de dades i la interacció entre dispositius, i en analitzar els resultats obtesos de les simulacions, evaluant el rendiment de la red i l'eficiència de la comunicació.

Es presentaran diferents casos d'estudi on s'aplique la simulació en diferents contextos i es resumiran les principals conclusions, proporcionant recomanacions per a futures investigacions en el camp de la simulació de reds IoT.



Paraules clau: SIMULADOR / PACKET TRACER / IoT / DISPOSITIUS / SENSORS

IoT NETWORK SIMULATION WITH PACKET TRACER

The main objective of this work will be to explore the capabilities of Packet Tracer as a simulation tool for IoT networks and demonstrate its usefulness in different scenarios.

Firstly, an overview of the basic concepts of IoT will be presented, including definitions, architecture, and involved technologies. The utilized Packet Tracer tool will also be developed, including its features, functionalities, and capabilities for network simulation.

Furthermore, the methodology for designing IoT networks using Packet Tracer version 8 will be explained, considering device selection, network parameter configuration, and integration of sensors and actuators.

The focus will be on demonstrating how to simulate IoT networks in Packet Tracer, including scenario configuration, data generation, device interaction, and analyzing the obtained simulation results, evaluating network performance and communication efficiency.

Different case studies will be presented where simulation is applied in different contexts, and the main conclusions will be summarized, providing recommendations for future research in the field of IoT network simulation.

Keywords: SIMULATOR / PACKET TRACER / IoT / DEVICES / SENSORS



Índice

I

Capítulo 1.	INTRODUCCIÓN:	1
Capítulo 2.	OBJETIVOS.....	2
Capítulo 3.	MARCO TEÓRICO	3
Capítulo 4.	DESARROLLO	4
Capítulo 5.	SIMULACIONES	32
5.1	ESCENARIO 1	32
5.1.1	Condiciones.....	38
5.1.2	Emails.....	42
5.2	ESCENARIO 2.	44
5.2.1	Placa SBC.....	49
5.2.2	Placas MCU.....	52
Capítulo 6.	RESULTADOS	56
Capítulo 7.	CONCLUSIONES	64
7.1	Ampliación futura	64
Capítulo 8.	BIBLIOGRAFÍA.....	65



Capítulo 1. INTRODUCCIÓN

En la era digital en la que nos encontramos, los simuladores se han vuelto sumamente importantes en diversos ámbitos. Mediante estas herramientas de software es posible recrear y modelar fenómenos complejos, sistemas físicos y virtuales, y eventos en un entorno controlado y seguro. El proceso de simulación consiste en intentar representar de forma exacta la realidad y predecir el comportamiento de los sistemas previamente a su implementación física, dentro de áreas de tan distintas como pueden ser la industria, la medicina, la ingeniería o los videojuegos.

Actualmente, una de las corrientes más recientes en el ámbito tecnológico y que ha contribuido enormemente en el avance de los simuladores, es la llamada Internet de las Cosas (IoT). Este concepto se entiende como la interconexión de elementos cotidianos a través de redes digitales, posibilitando la comunicación y el intercambio de información entre ellos. Todo ello ha resultado en un incremento exponencial de la cantidad de datos generados, desencadenando así la necesidad de implementar herramientas que permitan simular y analizar el comportamiento de estos sistemas IoT.

Hoy en día, pocos simuladores disponen de dicha tecnología, ya que se trata de una rama bastante novedosa. En este proyecto, la herramienta de simulación empleada será la versión 8.2.0 del Packet Tracer, que permitirá simular redes IoT y analizar su funcionamiento previamente a su implementación en el mundo real. Este tipo de simuladores IoT son muy útiles en aquellas aplicaciones que son de vital importancia para la seguridad, y en las que los errores pueden acarrear consecuencias graves. Permiten testear y optimizar las soluciones antes de implementarlas físicamente, ahorrando tiempo y recursos, así como probar distintos escenarios y evaluar el rendimiento en cada uno de ellos. Además, posibilitan la recopilación y análisis de extensas cantidades de datos generados por los dispositivos simulados, permitiendo así la identificación de patrones, la optimización de procesos y la toma de decisiones informadas. Asimismo, estas simulaciones pueden estudiar el consumo energético de los dispositivos, evaluar la calidad de los servicios y analizar la seguridad de la red.

Por tanto, los simuladores IoT se han convertido en una pieza fundamental dentro del desarrollo y el análisis de sistemas, permitiendo la investigación, diseño y optimización de estos, así como la anticipación a posibles errores y la mejora del rendimiento para su implementación en el mundo real.



Capítulo 2. OBJETIVOS

El objetivo principal de este proyecto será explorar las capacidades de Packet Tracer como herramienta de simulación para redes IoT y demostrar su utilidad en diferentes escenarios. Para ello, se determinará necesario plantear una serie de metas específicas que dirigirán el desarrollo y la ejecución de este.

En primer lugar, habrá que proporcionar una visión general de los fundamentos básicos de IoT, para así introducir cuales son los principios teóricos y los aspectos prácticos de esta tecnología. Con el objetivo de entender mejor el funcionamiento de la herramienta Packet Tracer utilizada, se detallarán los conceptos necesarios; la conexión de dispositivos, los protocolos de comunicación, la gestión de datos y la seguridad de las redes, entre otros.

Además, se deberán desarrollar las nuevas funcionalidades que ofrece la versión 8 de dicha herramienta de simulación, destacando aquellas que estén relacionadas con las redes IoT. Se expondrá también la capacidad de esta para diseñar escenarios de red, modelar dispositivos IoT, establecer conexiones y configurar parámetros relevantes para la simulación.

También se tendrán que explicar los aspectos necesarios para la selección y configuración de dispositivos IoT, así como para la distribución de estos en el proceso de creación de topologías de red. A su vez, se abordará el tema de la simulación, describiendo los procesos contenidos en esta, como la configuración de escenarios, generación de datos o realización de pruebas de funcionalidad.

Asimismo, se determinará necesaria la evaluación del rendimiento de las redes simuladas en diferentes escenarios y condiciones. Para ello, se tendrán en cuenta las medidas de diferentes parámetros, como la eficiencia energética, la calidad del servicio o la latencia, que permitirán identificar limitaciones y aplicar mejoras en el diseño de las redes.

Por último, se expondrán las conclusiones obtenidas, destacando tanto las ventajas como las limitaciones de la utilización de esta herramienta de simulación para redes IoT.



Capítulo 3. MARCO TEÓRICO

El Packet Tracer es la herramienta de simulación con la que se pretende trabajar en este proyecto, ampliamente utilizada para redes y sistemas de comunicación.

Incluye una gran variedad de dispositivos IoT virtuales, así como varios protocolos de comunicación con los que configurar dichos aparatos. Permite la creación de diferentes topologías de red IoT mediante la conexión y configuración de los dispositivos que la componen. Además, ofrece la posibilidad de crear sensores y actuadores personalizados a través de la programación de estos, y proporciona herramientas de análisis y depuración que permiten identificar problemas, supervisar el tráfico y evaluar el rendimiento de la red simulada. También facilita la creación de escenarios de simulación con los que probar diferentes situaciones de la red, y se integra correctamente con los dispositivos y escenarios del mundo real.

En conclusión, se trata de una herramienta versátil para la simulación de redes IoT, que permite probar diferentes configuraciones y escenarios antes de implementarlos en entornos reales.

Capítulo 4. DESARROLLO

Para la realización de este proyecto se ha diseñado un escenario de simulación que emula una vivienda familiar, con el objetivo de explorar como interactúan los distintos dispositivos y componentes, y descubrir la gran variedad de aparatos inteligentes que convierten una casa en un hogar conectado y eficiente.



Ilustración 1. Escenario de simulación

A continuación, se presentarán los diferentes dispositivos incorporados a la vivienda, detallando sus funciones, especificaciones, atributos y posibles configuraciones:

- Monitor de Nivel de Agua:



Ilustración 2. Monitor de Nivel de Agua

Se trata de un dispositivo diseñado para medir la cantidad de agua en pulgadas o centímetros.

Está equipado con una función de detección del nivel de agua a través de la lectura del valor WATER_LEVEL establecido en el objeto 'Entorno', que permite monitorear de manera precisa los cambios en el nivel esta.

No dispone de control directo, aunque si de local, en el que el objeto ‘Entorno’ es el encargado de controlar el valor WATER_LEVEL que se está capturando. En cuanto a las especificaciones de los datos, el formato del mensaje que proporciona el monitor es [nivel], donde “nivel” representa cualquier valor de salida mayor o igual a cero.

En la imagen a continuación se mostrará el apartado ‘Attributes’ del dispositivo, en el que aparecerán los valores asociados al tiempo medio entre fallos, el coste, la fuente de energía, las unidades de rack y el voltaje.

Name	Attribute
1 MTBF	26280
2 cost	40
3 power source	0
4 rack units	1
5 wattage	5

Property	Value
1 PROGRAMMING_EDITING_DIR	
2 level	0

Ilustración 3. Monitor de Nivel de Agua. Attributes

- Sensor de Movimiento:



Motion Detector

Ilustración 4. Sensor de movimiento

Este dispositivo ofrece funcionalidades específicas para la detección de movimiento a través del seguimiento del movimiento del ratón; se activa tan pronto como se detecta movimiento y se desactiva automáticamente después de 5 segundos de inactividad.

Para un control directo, se interactúa con el dispositivo utilizando la combinación de la tecla Alt y movimientos del ratón para gestionar su funcionamiento. En cambio, no dispone de control local ni remoto.

El sensor emite mensajes indicando el estado actual de este en un formato que sigue la estructura [estado], donde “estado” puede ser HIGH (activo) o LOW (inactivo).

La figura a continuación mostrará los datos asociados a los elementos del apartado ‘Attributes’ del sensor de movimiento:

Name	Attribute
1 MTBF	26280
2 cost	40
3 power source	0
4 rack units	1
5 wattage	5

Property	Value
1 PROGRAMMING_EDITING_DIR	
2 state	0

Ilustración 5. Sensor de movimiento. Attributes

- Puerta:



Door

Ilustración 6. Puerta

Se trata de un dispositivo que se puede abrir, cerrar, bloquear y desbloquear. Afecta al argón, monóxido de carbono, dióxido de carbono, hidrógeno, helio, metano, nitrógeno, O₂, ozono, propano y humo. Al abrirse la puerta, estos gases se reducen a un máximo del 2% del cambio total y la tasa de transferencia de humedad y temperatura aumenta en un 25%. Además, la tasa de transferencia de gas aumenta en un 100%.

Para un control directo de la misma existen una serie de acciones específicas. El bloqueo y desbloqueo se realiza clicando sobre la tecla Alt al mismo tiempo que sobre la cerradura. En cambio, para la apertura o cierre de la puerta, se clicará sobre esta a la vez que sobre la tecla Alt. Estas dos últimas funciones no son controlables de forma remota.

El dispositivo emite mensajes en un formato que sigue la estructura [puerta], [cerrojo], donde “puerta” puede tomar los valores 0 (cerrada), 1 (abierta) o -1 (no importa), y “cerrojo” puede estar a 0 (desbloqueado), 1 (bloqueado) o -1 (no importa).

En la siguiente imagen aparecerán los valores asociados a los campos del apartado ‘Attributes’ de la puerta:

Name	Attribute
1 MTBF	262800
2 cost	150
3 power source	0
4 rack units	5
5 wattage	5

Property	Value
1 PROGRAMMING_EDITING_DIR	
2 door state	0
3 lock state	0

Ilustración 7. Puerta. Attributes

- Aspersor:



Lawn Sprinkler

Ilustración 8. Aspersor

Es un dispositivo diseñado específicamente para el riego del césped, cuya función principal consiste en contribuir al aumento del nivel de agua en el área donde se utiliza. Afecta a una velocidad de 0,1 cm por segundo.

Es posible interactuar directamente con el dispositivo pulsando sobre este al mismo tiempo que se clicla sobre la tecla Alt.

Además, utiliza la ranura de entrada D0 para los datos y los mensajes que emite siguen el formato [estado], donde “estado” puede ser 0 (apagado) o 1 (encendido).

La figura siguiente mostrará los datos asociados a los atributos del aspersor:

Name	Attribute
1 MTBF	43800
2 cost	40
3 power source	0
4 rack units	2
5 wattage	5

Property	Value
1 PROGRAMMING_EDITING_DIR	
2 state	0

Ilustración 9. Aspersor. Attributes

- Puerta de garaje:



Garage Door

Ilustración 10. Puerta de garaje

Se trata de un dispositivo que se puede abrir y cerrar. Afecta al argón, monóxido de carbono, dióxido de carbono, hidrógeno, helio, metano, nitrógeno, O₂, ozono, propano y humo. Cuando se abre la puerta, estos gases se reducen a un máximo del 4% del cambio total, y la tasa de transferencia de humedad y temperatura aumenta en un 50%. Y, la tasa de transferencia de gas aumenta en un 100%.

Para un control directo de la puerta hay que pulsar a la vez que se clicca sobre la tecla Alt.

Para los datos, utiliza la ranura de entrada D0, y los mensajes que emite siguen el formato [estado], donde “estado” puede ser 0 (cerrada) o 1 (abierta).

En la siguiente imagen aparecerán los valores asociados a los campos del apartado ‘Attributes’ de la puerta de garaje:

Name	Attribute
1 MTBF	87600
2 cost	200
3 power source	0
4 rack units	10
5 wattage	5

Property	Value
1 PROGRAMMING_EDITING_DIR	
2 state	0

Ilustración 11. Puerta de garaje. Attributes

- Coche Antiguo:



Old Car

Ilustración 12. Coche Antiguo

Se trata de un dispositivo que tiene la opción de encenderse y apagarse según sea necesario. Además, afecta al monóxido de carbono a una tasa del 1% por hora, al dióxido de carbono a una tasa del 2% por hora y al humo a razón del 3% por hora. También afecta la temperatura ambiente a razón del 1% por hora.

No dispone de control local ni tampoco remoto. En cambio, permite interactuar directamente con él pulsando sobre este a la vez que se clicca sobre la tecla Alt.

La figura a continuación mostrará los datos asociados a los elementos del apartado 'Attributes' del automóvil:

Specifications		Physical		Config		Attributes	
Attributes:							
Name				Attribute			
1 MTBF				8760			
2 cost				2500			
3 power source				0			
4 rack units				1			
5 wattage				5			
Properties:							
Property				Value			
1 PROGRAMMING_EDITING_DIR							
2 state				0			
Refresh							
Top							
Advanced							

Ilustración 13. Ccoche antiguo. Attributes

- Detector de Monóxido de Carbono:



Carbon Monoxide Detector

Ilustración 14. Detector de Monóxido de Carbono

Es un dispositivo diseñado para detectar el nivel de monóxido de carbono (CO) en el entorno, cuya alarma se activará cuando este detecte un nivel de CO igual o superior al 20%.

Para la utilización de este detector se determina necesaria la presencia de un automóvil antiguo, ya que este hace variar el nivel de monóxido de carbono.

No dispone de ningún tipo de control; ni directo, ni local, ni remoto.

Los mensajes emitidos por el dispositivo siguen el formato [estado], [nivel], donde “estado” puede ser 0 (alarma apagada) o 1 (alarma encendida), y “nivel” es un número positivo.

En la siguiente imagen aparecerán los valores asociados a los campos del apartado ‘Attributes’ del detector:

Specifications		Physical		Config		Attributes	
Attributes:							
	Name			Attribute			
1	MTEF			43800			
2	cost			25			
3	power source			0			
4	rack units			1			
5	wattage			5			
Properties:							
	Property			Value			
1	PROGRAMMING_EDITING_DIR						
2	level			0			
3	state			0			
Refresh							
<input type="checkbox"/> Top							
Advanced							

**Ilustración 15. Detector de Monóxido de Carbono.
Attributes**

- Detector de Dióxido de Carbono:



Carbon Dioxide Detector

**Ilustración 16 Detector de
Dióxido de Carbono**

Se trata de un dispositivo diseñado para detectar el nivel de dióxido de carbono (CO₂) en el entorno, cuya alarma se activa cuando este detecte un nivel de dióxido de carbono igual o superior al 60%.

Para poder utilizarlo se necesita de un automóvil antiguo que cambie el nivel de CO₂.

No dispone de ningún tipo de control, ni directo, ni local, ni tampoco remoto.

Los mensajes emitidos por el detector siguen el formato [estado], [nivel], donde “estado” puede ser 0 (alarma apagada) o 1 (alarma encendida) y “nivel” es un número positivo.

La figura siguiente mostrará los datos asociados a los atributos del detector:

The screenshot shows a configuration window with tabs for Specifications, Physical, Config, and Attributes. The Attributes tab is active, displaying a table with 5 rows. Below it is a Properties section with a table showing 3 rows. At the bottom, there are 'Refresh', 'Top', and 'Advanced' buttons.

Name	Attribute
1 MTBF	43800
2 cost	25
3 power source	0
4 rack units	1
5 wattage	5

Property	Value
1 PROGRAMMING_EDITING_DIR	
2 level	0 035999998450279236
3 state	0

**Ilustración 17. Detector de Dióxido de Carbono.
Attributes**

- Cafetera:



Appliance

**Ilustración 18.
Cafetera**

Se trata de un electrodoméstico utilizado en el hogar que tiene la capacidad de ser encendido o apagado según las necesidades.

Para poder utilizarlo es necesario conectarlo a un SBC o botón y enviar un comando “customWrite” para encenderlo o apagarlo.

Utiliza la ranura de entrada 0 para indicar su estado. donde el valor 0 representa “Apagada” y el valor 1 “Encendida”.

La figura a continuación mostrará los datos asociados a los elementos del apartado ‘Attributes’ del electrodoméstico:

Name	Attribute
1 MTBF	26280
2 cost	75
3 power source	0
4 rack units	3
5 wattage	500

Property	Value
1 PROGRAMMING_EDITING_DIR	
2 state	0

Ilustración 19. Cafetera. Attributes

- Detector de humo:



Smoke Detector

Ilustración 20. Detector de humo

Es un dispositivo diseñado para detectar la presencia de humo en un entorno. Al detectar un nivel de humo igual o superior al 40%, activará una alarma.

Con el fin de utilizar este detector, es necesario hacer uso de un “Automóvil Antiguo” para cambiar el nivel de humo, ya que no dispone de ningún tipo de control; ni directo, ni local, ni tampoco remoto.

Los mensajes emitidos por el dispositivo siguen el formato [estado], [nivel], donde “estado” puede ser 0 (alarma apagada) o 1 (alarma encendida), y “nivel” es un número positivo.

En la siguiente imagen aparecerán los valores asociados a los campos del apartado ‘Attributes’ del detector:

Attributes:	
Name	Attribute
1 MTBF	43800
2 cost	40
3 power source	0
4 rack units	2
5 wattage	5

Properties:	
Property	Value
1 PROGRAMMING_EDITING_DIR	
2 level	0
3 state	0

Ilustración 21. Detector de humo. Attributes

- Ventana:



Window

Ilustración 22. Ventana

Se trata de un dispositivo que puede abrirse o cerrarse, y tiene la capacidad de ventilar ciertos gases. Afecta al argón, monóxido de carbono, dióxido de carbono, hidrógeno, helio, metano, nitrógeno, O₂, ozono, propano y humo. Al abrirse la ventana, estos gases se reducen a un máximo del 1% del cambio total, la tasa de transferencia de humedad y temperatura aumenta en un 20%. y la tasa de transferencia de gas aumenta en un 100%.

La ventana funciona en conjunto con el objeto Entorno; lee las variables CARBON_DIOXIDE y CARBON_MONOXIDE configuradas en el objeto y las cambia al activar los usuarios la apertura o cierre de esta.

Para una interacción directa con el dispositivo, los usuarios pueden abrirlo y cerrarlo pulsando sobre este a la vez que clican sobre la tecla Alt.

Los mensajes que emite siguen el formato [estado], donde “estado” puede ser 0 (cerrada) o 1 (abierta).

La figura siguiente mostrará los datos asociados a los atributos de la ventana:

Specifications		Physical		Config		Attributes	
Attributes:							
	Name			Attribute			
1	MTBF	87600					
2	cost	100					
3	power source	0					
4	rack units	10					
5	wattage	5					
Properties:							
	Property			Value			
1	PROGRAMMING_EDITING_DIR						
2	state	0					
Refresh							
<input type="checkbox"/> Top							
Advanced							

Ilustración 23. Ventana. Attributes

- Altavoz de Bluetooth:



Bluetooth Speaker

Ilustración 24. Altavoz de Bluetooth

Es un dispositivo diseñado para reproducir sonidos a través de una conexión Bluetooth desde un “Reproductor de Música Portátil”. Por tanto, puede conectarse y emparejarse con otros dispositivos a través de Bluetooth.

Para poder utilizar este dispositivo, primero es necesario emparejarlo con un “Reproductor de Música Portátil” mediante la conexión Bluetooth a través de la pestaña de Configuración.

Para habilitar el reproductor y enviar música de ruido blanco al altavoz, hay que presionar sobre el reproductor a la vez que se clicca sobre la tecla Alt. A partir de ahí, este envía los datos de audio al altavoz a través de la conexión Bluetooth.

No dispone de ningún tipo de control; ni directo, ni local, ni tampoco remoto.

En la siguiente imagen aparecerán los valores asociados a los campos del apartado ‘Attributes’ del Altavoz de Bluetooth:

Name	Attribute
1 MTBF	43800
2 cost	35
3 power source	0
4 rack units	1
5 wattage	5

Property	Value
1 PROGRAMMING_EDITING_DIR	
2 active	0
3 state	0

Ilustración 25. Altavoz de Bluetooth. Attributes

- Reproductor de Música Portátil:



Portable Music Player

Ilustración 26. Reproductor de Música Portátil

Es un dispositivo diseñado para reproducir música a través de Bluetooth, que también le permite conectarse y emparejarse con otros dispositivos. Además, tiene la capacidad de ser encendido y apagado según las necesidades.

Para poder utilizarlo, es necesario primero emparejarlo con un Altavoz Bluetooth mediante la pestaña de configuración. Y, para habilitarlo y enviar música de ruido blanco al altavoz, hay que presionar sobre este a la vez que se clica sobre la tecla Alt.

Este último procedimiento comentado es el que se utiliza también para controlar de forma directa el dispositivo.

La figura a continuación mostrará los datos asociados a los elementos del apartado 'Attributes' del reproductor:

Name	Attribute
1 MTBF	26260
2 cost	40
3 power source	0
4 rack units	1
5 wattage	5

Property	Value
1 PROGRAMMING_EDITING_DIR	
2 state	0

**Ilustración 27. Reproductor de Música Portátil.
Attributes**

- Webcam:



Webcam

**Ilustración 28.
Webcam**

Se trata de un dispositivo que tiene la capacidad de grabar y enviar datos. Puede ser encendida y apagada cuando sea necesario.

Los usuarios pueden interactuar directamente con esta pulsando sobre ella a la vez que clican la tecla Alt.

Los mensajes emitidos por la cámara siguen el formato [estado], donde “estado” puede ser 0 (apagada) o 1 (encendida).

En la siguiente imagen aparecerán los valores asociados a los campos del apartado ‘Attributes’ del dispositivo:

Attributes:	
Name	Attribute
1 MTBF	26280
2 cost	20
3 power source	0
4 rack units	1
5 wattage	5

Properties:	
Property	Value
1 PROGRAMMING_EDITING_DIR	
2 state	0

Ilustración 29Webcam. Attributes

- Desagüe de agua:



Ilustración 30. Desagüe de agua

Es un dispositivo diseñado para drenar el agua en un área específica a una velocidad de 0,5 cm por hora.

Los usuarios pueden interactuar directamente con este dispositivo haciendo clic con la tecla Alt.

Los mensajes emitidos por el desagüe siguen el formato [estado], donde “estado” puede ser 0 (apagado) o 1 (encendido).

La figura siguiente mostrará los datos asociados a sus atributos:

Specifications Physical Config **Attributes**

Attributes:

	Name	Attribute
1	MTBF	26280
2	cost	40
3	power source	0
4	rack units	1
5	wattage	5

Properties:

	Property	Value
1	PROGRAMMING_EDITING_DIR	
2	state	0

Top

Ilustración 31. Desagüe de agua. Attributes

- Rociador de fuego:



Fire Sprinkler

Ilustración 32. Rociador de fuego

Se trata de un dispositivo diseñado para apagar el fuego, cuya función principal es aumentar el nivel de agua en una zona específica, al cual afecta a una velocidad de 0,1 cm por segundo.

Se puede interactuar directamente con este dispositivo haciendo clic con la tecla Alt.

Los mensajes que emite este aparato siguen el formato [estado], donde “estado” puede ser 0 (apagado) o 1 (encendido).

En la siguiente imagen aparecerán los valores asociados a los campos del apartado ‘Attributes’ del rociador:

Specifications		Physical		Config		Attributes	
Attributes:							
	Name		Attribute				
1	MTBF		43800				
2	cost		40				
3	power source		0				
4	rack units		1				
5	wattage		5				
Properties:							
	Property		Value				
1	PROGRAMMING_EDITING_DIR						
2	state		0				
<input type="button" value="Refresh"/>							
<input type="checkbox"/> Top <input type="button" value="Advanced"/>							

Ilustración 33. Rociador de fuego. Attributes

- Humidificador:



Humidifier

**Ilustración 34.
Humidificador**

Es un dispositivo cuya función principal es aumentar la humedad en el ambiente donde se encuentra, y lo hace a razón de 1% por hora. Puede encenderse y apagarse cuando sea preciso.

Los usuarios pueden interactuar directamente con este pulsando sobre él a la vez que clican sobre la tecla Alt.

Utiliza la ranura de entrada digital D0 para recibir señales de control, y emite mensajes que siguen el formato [estado], donde “estado” puede ser apagado (0) o encendido (1).

La figura a continuación mostrará los datos asociados a los elementos del apartado ‘Attributes’ del humidificador:

Name	Attribute
1 MTBF	26280
2 cost	40
3 power source	0
4 rack units	3
5 wattage	5

Property	Value
1 PROGRAMMING_EDITING_DIR	
2 state	0

Ilustración 35. Humidificador. Attributes

- Monitor de Humedad:

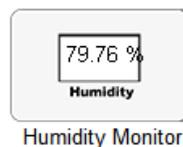


Ilustración 36. Monitor de humedad

Se trata de un dispositivo diseñado para detectar y mostrar el nivel de humedad en el entorno.

Para utilizarlo, no dispone de ningún tipo de control; ni directo, ni local, ni tampoco remoto. Este simplemente lee los valores del entorno para determinar la humedad y muestra el resultado. Además, también permite visualizarlo en un servidor de registro.

Los mensajes que emite este aparato siguen el formato [nivel], donde “nivel” puede ser cualquier valor de salida mayor o igual a 0.

En la siguiente imagen aparecerán los valores asociados a los campos del apartado ‘Attributes’ del monitor:

Specifications		Physical		Config		Attributes	
Attributes:							
	Name			Attribute			
1	MTBF	26280					
2	cost	40					
3	power source	0					
4	rack units	1					
5	wattage	5					
Properties:							
	Property			Value			
1	PROGRAMMING_EDITING_DIR						
2	level	75.01850128173828					
<input type="button" value="Refresh"/>							
<input type="checkbox"/> Top <input type="button" value="Advanced"/>							

Ilustración 37. Monitor de humedad. Attributes

- Lámpara:



**Ilustración 38.
Lámpara**

Es un dispositivo de iluminación que se puede encender y apagar cuando sea necesario. Ofrece también la posibilidad de atenuar la luz para ajustar su intensidad luminosa. Y, su función principal es emitir luz visible en el entorno donde se encuentra, a la cual afecta con una potencia máxima del 20%.

Este aparato permite una interacción directa haciendo clic con la tecla Alt.

Los mensajes emitidos por este dispositivo siguen el formato [estado], donde “estado” puede ser 0 (apagado), 1 (atenuado) o 2 (encendido).

La figura siguiente mostrará los datos asociados a los atributos de la lámpara:

Specifications		Physical		Config		Attributes	
Attributes:							
	Name			Attribute			
1	MTBF	43800					
2	cost	40					
3	power source	0					
4	rack units	3					
5	wattage	5					
Properties:							
	Property			Value			
1	PROGRAMMING_EDITING_DIR						
2	state	0					
Refresh							
Top							
Advanced							

Ilustración 39. Lámpara. Attributes

- Aire Acondicionado:



Air Conditioner

Ilustración 40. Aire Acondicionado

Se trata de un dispositivo de enfriamiento diseñado para reducir la temperatura y la humedad en un espacio a -10°C y en un 2% por hora respectivamente.

Para utilizar este aparato, se tiene que conectar D0 en la ranura D2 del termostato, ya que no dispone de control directo.

Los mensajes emitidos por el aparato de aire acondicionado siguen el formato [estado], donde “estado” puede ser 0 (AC apagado) o 1 (AC encendido).

La entrada D0 puede adquirir los valores “HIGH” o “LOW” para controlar el encendido o apagado del AC.

En la siguiente imagen aparecerán los valores asociados a los campos del apartado ‘Attributes’ del dispositivo:

Specifications		Physical		Config		Attributes	
Attributes:							
	Name			Attribute			
1	MTBF	300000					
2	cost	250					
3	power source	1					
4	rack units	2					
5	wattage	5					
Properties:							
	Property			Value			
1	PROGRAMMING_EDITING_DIR						
2	VOLUME_AT_RATE	100000					
Refresh							
<input type="checkbox"/> Top Advanced							

Ilustración 41. Aire Acondicionado. Attributes

- Termostato:



**Ilustración 42.
Termostato**

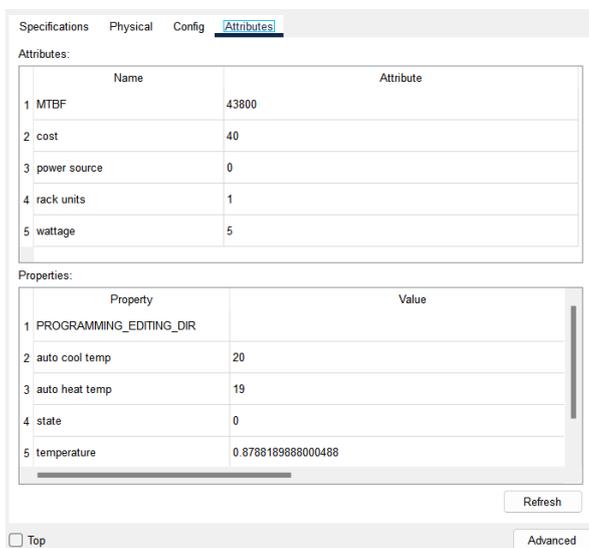
Es un dispositivo de control de temperatura que cuenta con diferentes modos de funcionamiento. Puede estar en modo de enfriamiento para reducir la temperatura, en modo de calefacción para aumentarla o en modo automático para ajustarla según parámetros específicos. También puede estar en modo apagado, lo que significa que no realiza acciones de calefacción o enfriamiento.

Para utilizar este aparato, se debe conectar un horno en D1 y un Aire Acondicionado (AC) en D2. También se puede conectar un sensor de temperatura para mostrar cambios en la temperatura ambiente.

Permite una interacción directa pulsando sobre él a la vez que se pulsa sobre la tecla Alt.

La ranura D0 se utiliza para recibir datos de temperatura, y los mensajes que emite el termostato siguen el formato [estado], [temperatura], [temperaturaAutoEnfriamiento], [temperaturaAutoCalefaccion], donde “estado” puede ser 0 (apagado), 1 (enfriamiento), 2 (calefacción) o 3 (automático), “temperatura” representa la temperatura actual en grados Celsius y “temperaturaAutoEnfriamiento” y “temperaturaAutoCalefaccion” son umbrales para el modo automático.

La figura a continuación mostrará los datos asociados a los elementos del apartado ‘Attributes’ del termostato:



Specifications Physical Config **Attributes**

Attributes:

	Name	Attribute
1	MTBF	43800
2	cost	40
3	power source	0
4	rack units	1
5	wattage	5

Properties:

	Property	Value
1	PROGRAMMING_EDITING_DIR	
2	auto cool temp	20
3	auto heat temp	19
4	state	0
5	temperature	0.8788189888000488

Refresh

Top

Ilustración 43. Termostato. Attributes

- Aparato de calefacción:



Furnace

Ilustración 44. Aparato de calefacción

Se trata de un dispositivo diseñado para aumentar la temperatura en 10°C por hora y reducir la humedad en un 2% por hora en el espacio en el que esté instalado.

Para utilizar este dispositivo, se debe conectar D0 en la ranura D1 del termostato, ya que no dispone de control directo.

Los mensajes emitidos por el aparato de calefacción siguen el formato [estado], donde “estado” puede ser 0 (apagado) o 1 (encendido).

La entrada D0 del dispositivo puede adquirir los valores “HIGH” o “LOW” para controlar el encendido y apagado de este.

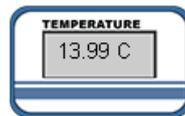
En la siguiente imagen aparecerán los valores asociados a los campos del apartado ‘Attributes’ del aparato de calefacción:

Name	Attribute
1 MTBF	262800
2 cost	500
3 power source	0
4 rack units	5
5 wattage	5

Property	Value
1 PROGRAMMING_EDITING_DIR	
2 VOLUME_AT_RATE	100000
3 state	0

Ilustración 45Aparato de calefacción. Attributes

- Monitor de temperatura:



Temperature Monitor

Ilustración 46. Monitor de temperatura

Es un dispositivo diseñado para recopilar datos sobre la temperatura del entorno y presentarlos de manera legible.

No dispone de ningún tipo de control; ni directo, ni local, ni tampoco remoto.

La figura siguiente mostrará los datos asociados a los atributos de este monitor:

Specifications Physical Config **Attributes**

Attributes:

	Name	Attribute
1	MTBF	43800
2	cost	40
3	power source	0
4	rack units	1
5	wattage	5

Properties:

	Property	Value
1	PROGRAMMING_EDITING_DIR	
2	level	5.555560111999512

Refresh

Top Advanced

Ilustración 47. Monitor de temperatura. Attributes

- Sirena:



Siren

Ilustración 48.
Sirena

Se trata de un dispositivo que emite sonidos y puede ser activado o desactivado.

Se puede interactuar directamente con este aparato pulsando sobre él mientras se clicla la tecla Alt.

La sirena utiliza la ranura de entrada D0 para los datos, y emite mensajes que siguen el formato [estado], donde “estado” puede ser 0 (apagada) o 1 (encendida).

En la siguiente imagen aparecerán los valores asociados a los campos del apartado ‘Attributes’ del dispositivo:

Specifications		Physical		Config		Attributes	
Attributes:							
Name				Attribute			
1 MTBF				26280			
2 cost				40			
3 power source				0			
4 rack units				1			
5 wattage				5			
Properties:							
Property				Value			
1 PROGRAMMING_EDITING_DIR							
2 state				0			
Refresh							

Ilustración 49. Sirena. Attributes

- Ventilador de techo:



Fan

Ilustración 50. Ventilador de techo

Se trata de un dispositivo que puede ser controlado en términos de encendido y velocidad.; puede ser apagado para detener su funcionamiento, y puede configurarse para funcionar a una velocidad baja o alta.

Se puede interactuar directamente con este aparato pulsando sobre él al mismo tiempo que se pulsa la tecla Alt.

Los mensajes que emite este dispositivo siguen el formato [estado], donde “estado” puede ser 0 (apagado), 1 (velocidad baja) o 2 (velocidad alta).

En la siguiente imagen aparecerán los valores asociados a los campos del apartado ‘Attributes’ del ventilador:

Name	Attribute
1 MTBF	87600
2 cost	100
3 power source	0
4 rack units	3
5 wattage	5

Property	Value
1 PROGRAMMING_EDITING_DIR	
2 state	0

Ilustración 51. Ventilador de techo. Attributes

Para controlar localmente los dispositivos comentados, es necesaria la conexión de estos a un SBC/MCU/Thing, así como la utilización de la API “customWrite” de acuerdo con las especificaciones de datos. Y, para hacerlo de forma remota, hay que conectarlos al Servidor de Registro a través del apartado de Configuración de cada uno de ellos.

Además de todos los dispositivos definidos anteriormente, también se han incorporado 3 dispositivos finales: un portátil, una impresora y una computadora.



Ilustración 52. Dispositivos finales

A través de la opción ‘Personalizar icono en vista lógica’ del apartado ‘Physical’ de estos dispositivos, se ha modificado la imagen de cada uno de ellos:



Ilustración 53. Dispositivos finales. Iconos personalizados

Las figuras siguientes mostrarán los valores asociados a los campos del apartado ‘Attributes’ de cada uno de los 3 dispositivos finales:

Name	Attribute
1 MTBF	43800
2 cost	1000
3 power source	0
4 rack units	3
5 wattage	150

Property	Value
1 PROGRAMMING_EDITING_DIR	

Refresh

Top

Ilustración 55. Dispositivos finales. Computadora. Attributes

Name	Attribute
1 MTBF	26280
2 cost	1000
3 power source	0
4 rack units	2
5 wattage	60

Property	Value
1 PROGRAMMING_EDITING_DIR	

Ilustración 54. Dispositivos finales. Portátil. Attributes

Physical Config Attributes

Attributes:

	Name	Attribute
1	MTEF	17520
2	cost	1500
3	power source	0
4	rack units	3
5	wattage	30

Properties:

Property	Value
----------	-------

Top

**Ilustración 56. Dispositivos finales. Impresora.
Attributes**

Capítulo 5. SIMULACIONES

Una vez establecido el escenario de simulación inicial, se han implementado una serie de configuraciones y programaciones sobre este y los dispositivos que lo forman. Estas acciones realizadas han sido divididas en dos escenarios de simulación distintos.

En el primero de ellos, el objetivo ha sido establecer una serie de condiciones sobre los dispositivos para determinar cuándo activar cada uno de ellos. Y también, crear un servicio de correo electrónico dentro del hogar para que los dispositivos que pertenezcan a él puedan comunicarse entre sí.

En el segundo, la finalidad ha sido la utilización y programación de placas SBC y MUC que activen los dispositivos conectados a ellas.

5.1 ESCENARIO 1

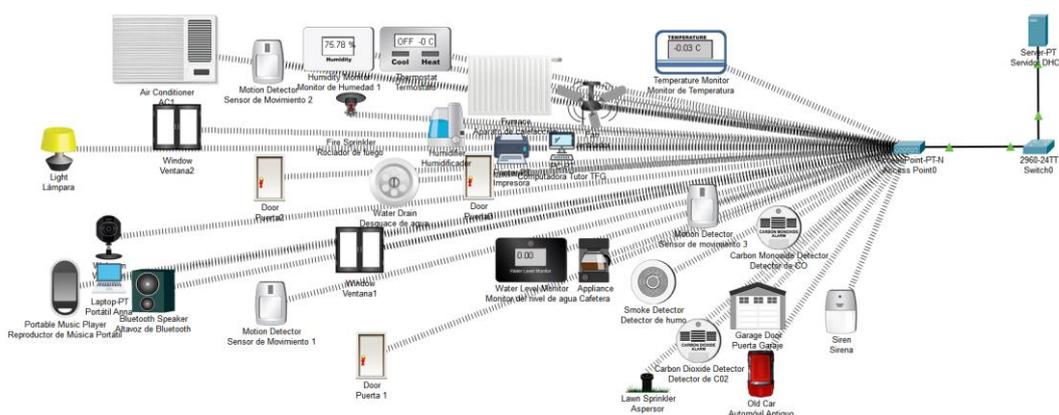


Ilustración 57. Escenario 1

En este proyecto se han añadido un Servidor, un Switch y un Access Point, para así poder conectar a través de internet todos los dispositivos. Para ello, se han llevado a cabo las siguientes configuraciones:

- Configuración del servidor:

En primer lugar, se conecta al switch a través del puerto FastEthernet0 mediante un cable de cobre directo. Luego, a través del apartado 'IP Configuration' del 'Desktop' del servidor, se le asigna a este la dirección IPv4 192.168.0.1, y se establece la misma como Servidor DNS.

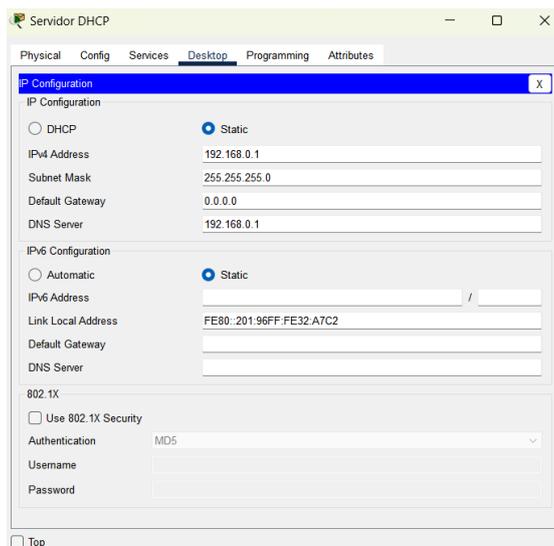


Ilustración 58. Escenario 1. Servidor. IP Config

En el apartado ‘Services’ se activan el servicio IoT, el DNS y el DHCP, en el cual se configura la dirección establecida del Servidor DNS para que este pueda dar direcciones IP a todos los dispositivos conectados. Además, se establece la dirección inicial desde la que el servidor puede empezar a asignar a los dispositivos.

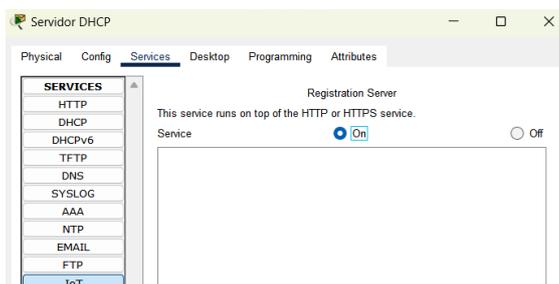


Ilustración 59. Escenario 1. Servidor. Services. IoT

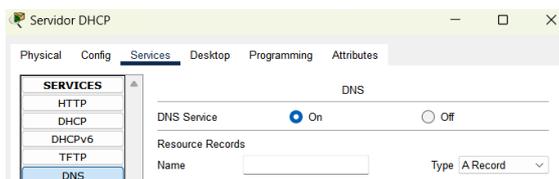


Ilustración 60. Escenario 1. Servidor. Services. DNS

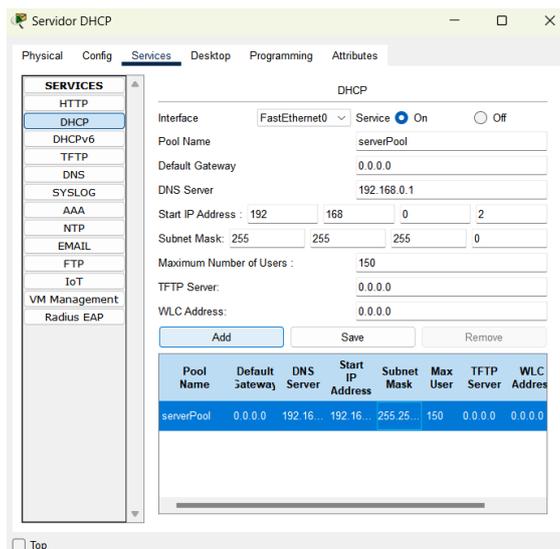


Ilustración 61. Escenario 1. Servidor. Services. DHCP

En el apartado 'IoT Monitor' del Desktop del servidor, se establece la dirección de este como dirección del servicio IoT y se configuran unas credenciales de usuario y contraseña para poder acceder a este servicio.

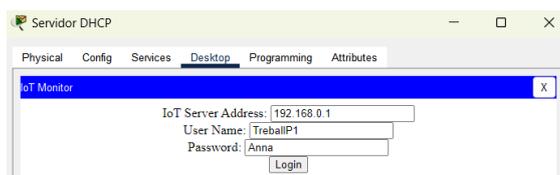


Ilustración 62. Escenario 1. Servidor. IoT Monitor

- Configuración del Access Point:

En primer lugar, se conecta al switch a través del puerto 0 mediante un cable de cobre directo. Luego, se accede al apartado de configuración del puerto 1 y se configuran el SSID y las credenciales de autenticación.

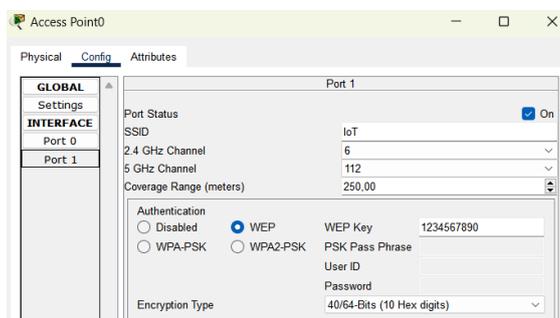


Ilustración 63. Escenario 1. Access Point. Port 1

- Configuración de los dispositivos:

Primeramente, se accede al apartado I/O Config de la configuración avanzada de los dispositivos para asegurarse de que el adaptador de red seleccionado es el inalámbrico.



Ilustración 64. Escenario 1. Dispositivos. Network Adapter

En el caso de los dispositivos finales; la computadora, el portátil y la impresora, se entra al apartado ‘Physical’, se apaga el dispositivo, se cambia el adaptador existente por el inalámbrico y se vuelve a encender el dispositivo.

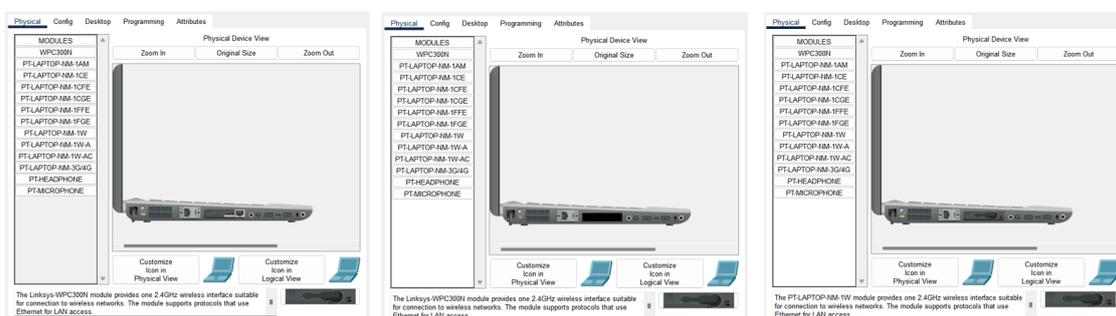


Ilustración 65. Escenario 1. Dispositivos finales. Physical

Luego, se accede al apartado ‘Wireless0’ de la configuración y se ingresan el SSID y las credenciales de acceso establecidas en el punto de acceso.

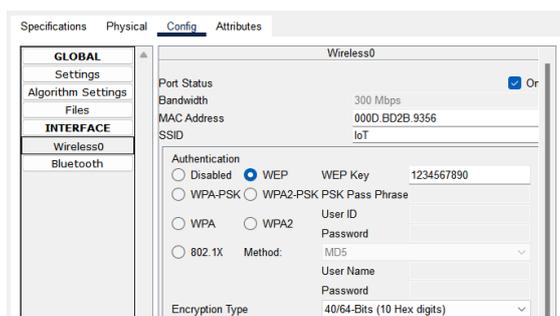


Ilustración 66. Escenario 1. Dispositivos. Wireless0

Además, en el servicio IoT del apartado de ‘Settings’ de la configuración, se selecciona la opción de ‘Remote Service’ y se añaden la dirección y las credenciales del servicio IoT establecido en el servidor.

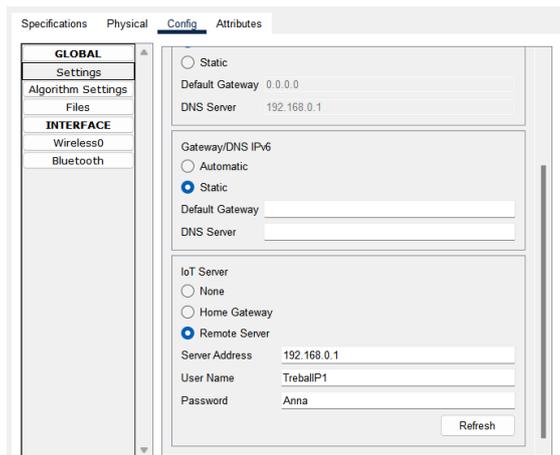


Ilustración 67. Escenario 1. Dispositivos. IoT Server

De esta manera, ya están todos los dispositivos del hogar conectados a internet a través del Access Point, cada uno con su propia dirección IP asignada por el servidor DHCP dentro del rango de direcciones 192.168.0.0/24.

Además, el servicio IoT establecido permite controlar desde el servidor todos y cada uno de los dispositivos IoT configurados. Para ello, hay que acceder al 'Web Browser o al 'IoT Monitor' del apartado 'Desktop' del servidor.

The image displays 12 screenshots of the IoT Monitor web interface, arranged in a 4x3 grid. Each screenshot shows a different device being controlled or monitored. The interface includes a navigation bar with 'Home', 'Conditions', 'Editor', and 'Log Out' options. Each device card shows its name, ID, and current status, along with control buttons like 'Unlock', 'Dim', 'On', 'Off', 'Cooling', 'Heating', 'Auto', and 'Set'.

Ilustración 68. Escenario 1. Servidor IoT Monitor. Control dispositivos

5.1.1 Condiciones

Este servicio contiene un apartado denominado ‘Conditions’ en el que se han establecido una serie de condiciones sobre los dispositivos:

- Condición 1: Encender Sensor Mov 1.

Al pulsar sobre la tecla Alt a la vez que se clicla sobre el Sensor de Movimiento 1 para activarlo, se abrirán la Puerta 1 y la Ventana 1, se encenderán el Reproductor de Música Portátil y la Webcam, y se abrirá el Desguace de agua.

- Condición 2: Apagar Sensor Mov 1.

Si el Sensor de Movimiento 1 está desactivado, todos los dispositivos comentados en la condición anterior, también lo estarán.

Name: Encender Sensor Mov 1
Enabled:

If:

Match: All	Sensor de Movimiento 1	On	is	true	+ Condition	+ Group
------------	------------------------	----	----	------	-------------	---------

Then set:

Puerta 1	Lock	to	Unlock	+ Action
Ventana1	On	to	true	-
Reproductor de Música Portátil	On	to	true	-
Desguace de agua	Status	to	true	-
Webcam	On	to	true	-

Ilustración 69. Escenario 1. Condición 1

Name: Apagar Sensor Mov 1
Enabled:

If:

Match: All	Sensor de Movimiento 1	On	is	false	+ Condition	+ Group
------------	------------------------	----	----	-------	-------------	---------

Then set:

Puerta 1	Lock	to	Lock	+ Action
Ventana1	On	to	false	-
Reproductor de Música Portátil	On	to	false	-
Desguace de agua	Status	to	false	-
Webcam	On	to	false	-

Ilustración 70. Escenario 1. Condición 2

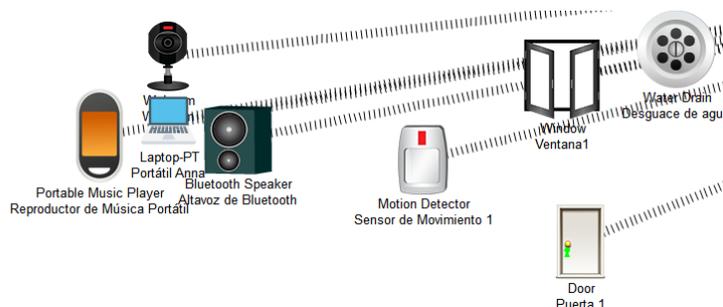


Ilustración 71. Escenario 1. Demostración Condición 1

- Condición 3: Encender Sensor Mov 2.

Al pulsar sobre el Sensor de Movimiento 2 para activarlo, a la vez que se clicca sobre la tecla Alt, se abrirán la Puerta 2 y la Ventana 2, se encenderán la Lámpara, el AC1, el Termostato y el Aparato de calefacción, y se activarán el Humidificador y el Rociador de fuego.

- Condición 4: Apagar Sensor Mov 2.

Si el Sensor de Movimiento 2 no está activado, tampoco lo estarán ninguno de los dispositivos mencionados en la condición anterior.

Name Encender Sensor Mov 2
Enabled

If:

Match All

Sensor de Movimiento 2 On is true

Then set:

Lámpara	Status	to	On
Ventana2	On	to	true
Puerta2	Lock	to	Unlock
Humidificador	Status	to	true
AC1	On	to	true
Termostato	Status	to	Auto
Rociador de fuego	Status	to	true
Aparato de calefacción	On	to	true

Ilustración 72. Escenario 1. Condición 3

Name Apagar Sensor Mov 2
Enabled

If:

Match All

Sensor de Movimiento 2 On is false

Then set:

Lámpara	Status	to	Off
Ventana2	On	to	false
Puerta2	Lock	to	Lock
Humidificador	Status	to	false
AC1	Termi		
Rociador de fuego	Status	to	false
Aparato de calefacción	On	to	false

Ilustración 74. Condición 4

Ilustración 73. Escenario 1. Condición 4

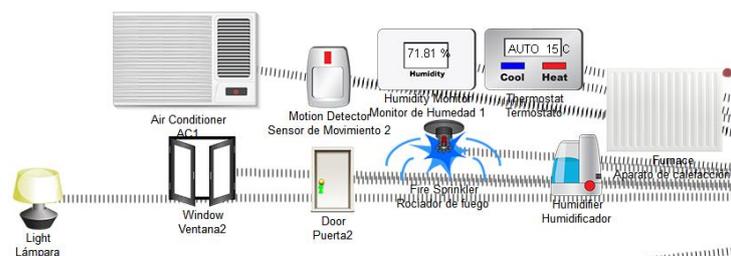


Ilustración 75. Escenario 1. Demostración Condición 3

- Condición 5: Encender Sensor Mov 3.

Al pulsar sobre la tecla Alt a la vez que se clica sobre el Sensor de Movimiento 1 para activarlo, se abrirán la Puerta 3 y la Puerta del garaje, se encenderán la cafetera y el ventilador, y se activará el aspersor.

- Condición 6: Apagar Sensor Mov 3.

Si el Sensor de Movimiento 3 está desactivado, todos los dispositivos comentados en la condición anterior, también lo estarán.

Name: Encender Sensor Mov 3
Enabled:

If:

Match: All	Sensor de movimiento 3	On	is	true	+ Condition	+ Group
------------	------------------------	----	----	------	-------------	---------

Then set:

Aspersor	Status	to	true	+ Action
Puerta Garaje	On	to	true	-
Cafetera	On	to	true	-
Ventilador	Status	to	High	-
Puerta3	Lock	to	Unlock	-

Ilustración 76. Escenario 1. Condición 5

Name: Apagar Sensor Mov 3
Enabled:

If:

Match: All	Sensor de movimiento 3	On	is	false	+ Condition	+ Group
------------	------------------------	----	----	-------	-------------	---------

Then set:

Aspersor	Status	to	false	+ Action
Puerta Garaje	On	to	false	-
Puerta3	Lock	to	Lock	-
Cafetera	On	to	false	-
Ventilador	Status	to	Off	-

Ilustración 77. Escenario 1. Condición 6

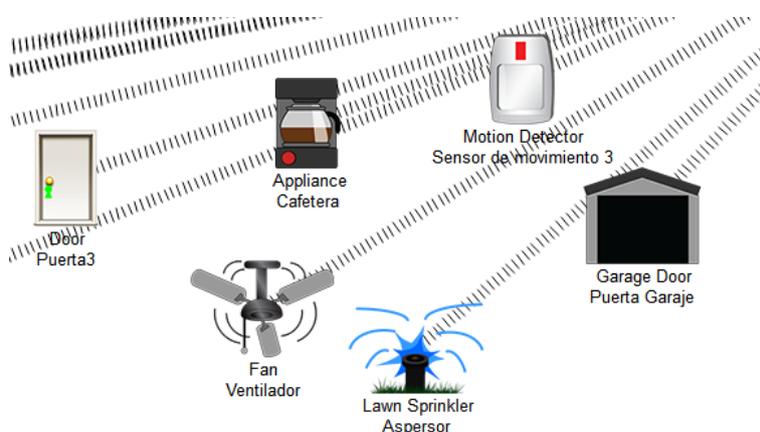


Ilustración 78. Escenario 1. Demostración Condición 5

- Condición 7: Encender Sirena.

Al pulsar sobre el Automóvil Antiguo para activarlo a la vez que se clicla sobre la tecla Alt, este desprende humo y ciertos gases como el dióxido y el monóxido de carbono. Si el detector de CO₂ detecta una cantidad de este gas superior o igual a 0,12, el detector de CO detecta una cantidad de este mayor o igual a 0,04, y el detector de humo detecta una cantidad de este mayor o igual a 0,1, se activará la sirena.

- Condición 8: Apagar Sirena.

La sirena se apagará, si los detectores mencionados detectan una cantidad inferior de humo y gases a la establecida en la condición anterior.

Name: Encender Sirena
Enabled:

If:

Match: All	Detector de C02	Level	>=	0.12	+ Condition	+ Group
	Detector de CO	Level	>=	0.04	-	-
	Detector de humo	Level	>=	0.1	-	-

Then set:

Sirena On to true

+ Action

Ilustración 79. Escenario 1. Condición 7

Name: Apagar Sirena
Enabled:

If:

Match: All	Detector de C02	Level	<	0.12	+ Condition	+ Group
	Detector de CO	Level	<	0.04	-	-
	Detector de humo	Level	<	0.1	-	-

Then set:

Sirena On to false

+ Action

Ilustración 80. Escenario 1. Condición 8

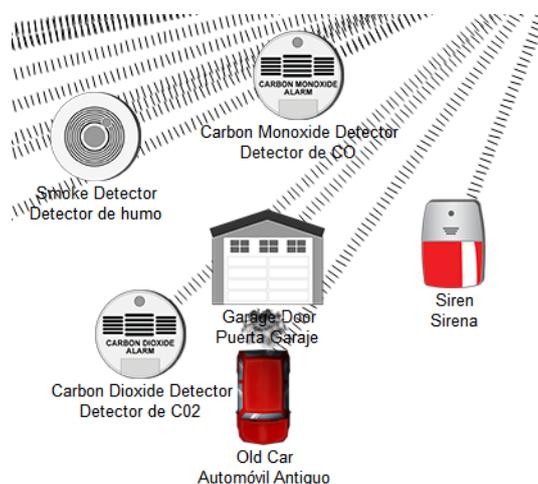


Ilustración 81. Escenario 1. Demostración Condición 7

5.1.2 Emails

También se ha diseñado un servicio DNS de correo electrónico para que el portátil del comedor y la computadora de la habitación se puedan comunicar a través de él:

- En primer lugar, se accede al apartado ‘Services’ del servidor y se crea dicho servicio DNS asignándole la dirección del DNS.

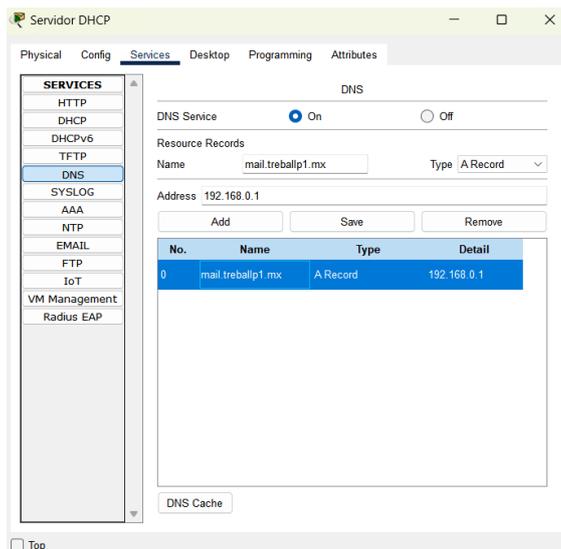


Ilustración 82. Escenario 1. Email. Servicio DNS

- Luego, se activa el servicio ‘EMAIL’ del mismo apartado, se añade el nombre de dominio del servicio creado y se configuran las credenciales de los usuarios que van a comunicarse a través del dispositivo.

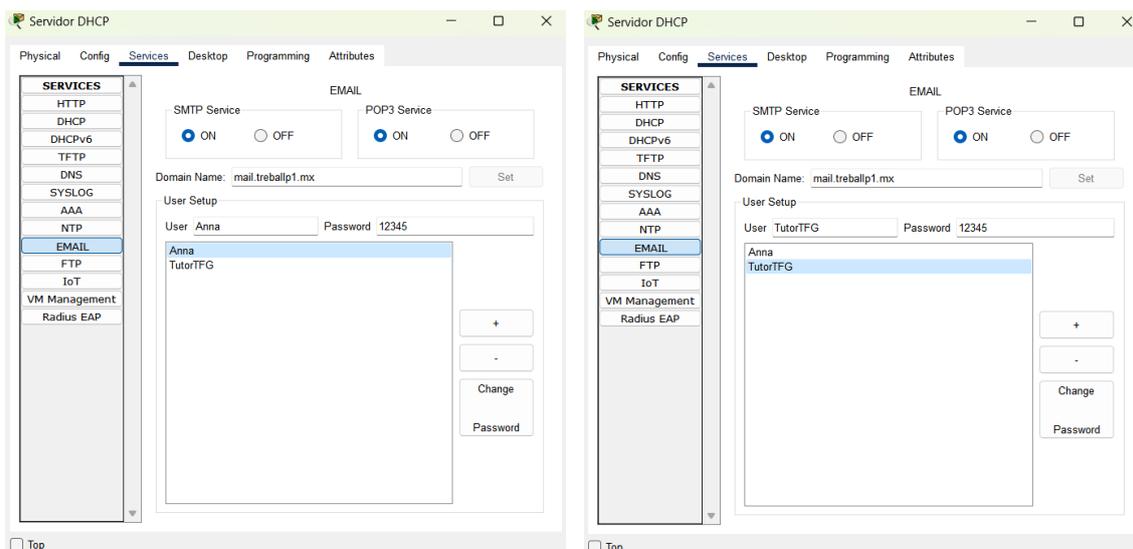


Ilustración 83. Escenario 1. Email. Servicio EMAIL

- Posteriormente, se accede al apartado ‘Email’ del ‘Desktop’ del portátil y de la computadora y se añade la información pertinente del servidor; nombre, apellidos y correo electrónico del usuario, servidor de correo de entrada y salida, y las credenciales de acceso configuradas en el servicio ‘EMAIL’ del servidor.

Portátil Anna

Physical Config Desktop Programming Attributes

Configure Mail X

User Information

Your Name: Anna Sánchez Martínez

Email Address: anna@mail.treballp1.mx

Server Information

Incoming Mail Server: mail.treballp1.mx

Outgoing Mail Server: mail.treballp1.mx

Logon Information

User Name: Anna

Password: ●●●●●●

Save Remove Clear Reset

Top

Ilustración 84. Escenario 1. Email. Info Portátil

Computadora Tutor TFG

Physical Config Desktop Programming Attributes

Configure Mail X

User Information

Your Name: Tutor TFG

Email Address: tutorfg@mail.treballp1.mx

Server Information

Incoming Mail Server: mail.treballp1.mx

Outgoing Mail Server: mail.treballp1.mx

Logon Information

User Name: TutorTFG

Password: ●●●●●●

Save Remove Clear Reset

Top

Ilustración 85. Escenario 1. Email. Info Computadora

5.2 ESCENARIO 2

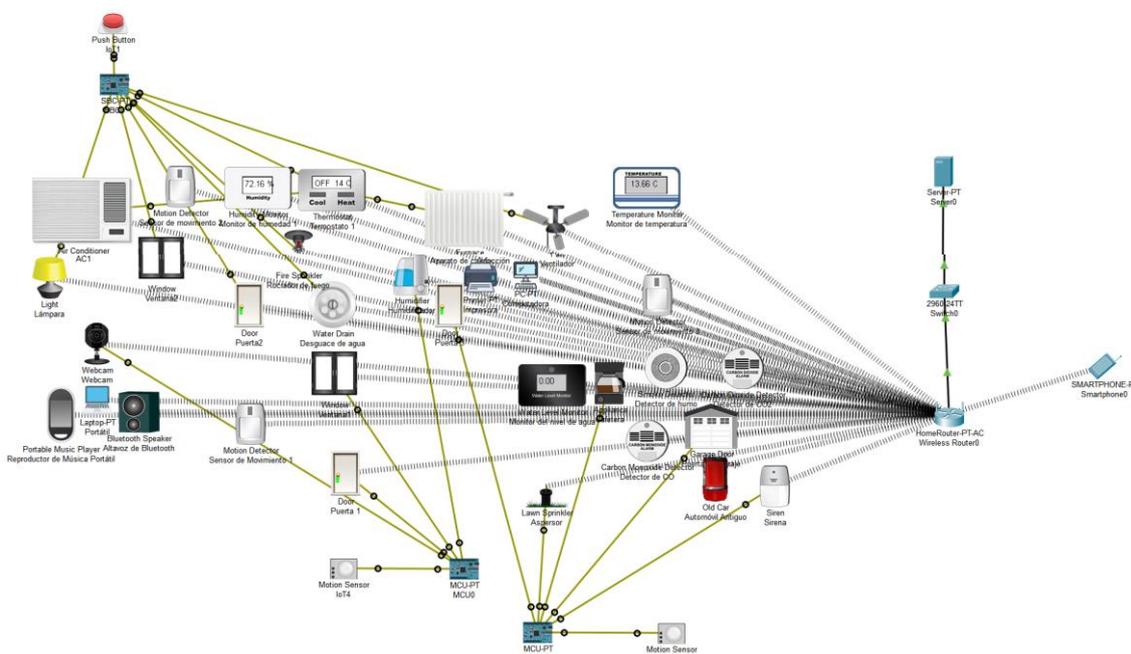


Ilustración 86. Escenario 2

En este proyecto se han añadido un Servidor, un Switch, un Router Inalámbrico y un Teléfono Inteligente, para así poder controlar a través de él todos los dispositivos IoT. Para ello, se han llevado a cabo las siguientes configuraciones:

- Configuración del Servidor:

Primeramente, desde el puerto FastEthernet0 y mediante un cable de cobre directo, se conecta al switch. Posteriormente, se le asigna al servidor la dirección IPv4 192.169.0.2 desde el apartado 'IP Configuration' su escritorio. Y también se establece la dirección 192.168.0.1 como puerta de enlace predeterminada.

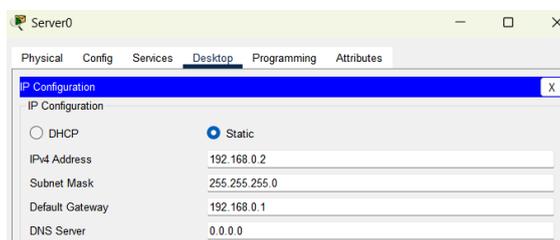


Ilustración 87. Escenario 2. Servidor. IP Config

Luego, se activan los servicios 'IoT' y 'DHCP', y en este último, se configura la dirección de la puerta de enlace predeterminada establecida en la configuración anterior. También se añade la dirección inicial desde la que el servidor puede empezar a asignar a los dispositivos.

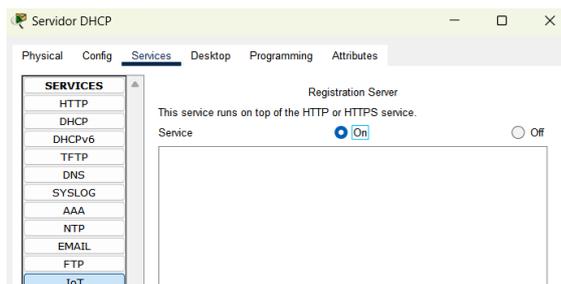


Ilustración 89. Escenario 2. Servidor. IoT

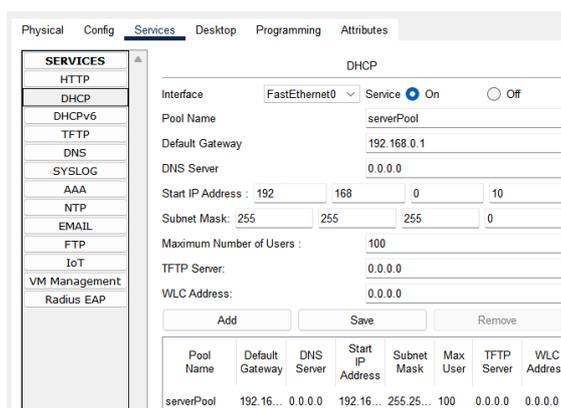


Ilustración 88. Escenario 2. Servidor. DHCP

Con el objetivo de configurar un servicio IoT, se accede al apartado 'IoT Monitor' del escritorio, se establece como dirección del servicio la del servidor y se configuran unas credenciales de ingreso para acceder a él.



Ilustración 90. Escenario 2. Servidor. Servicio IoT

- Configuración del Router Inalámbrico:

En primer lugar, desde el puerto GigabitEthernet1 y mediante un cable de cobre directo, se conecta al switch. Luego, en el apartado 'Setup' de la Interfaz Gráfica del Usuario (GUI), se establece Punto de Acceso Inalámbrico (Wireless AP) como tipo de conexión a internet.



Ilustración 91. Escenario 2. Router Inalámbrico. GUI. Wireless AP

Posteriormente, se accede al apartado ‘Wireless’ de la GUI y en la sección de 2.4 GHz, se añade el SSID.



Ilustración 92. Escenario 2. Router Inalámbrico. GUI. Wireless

Y luego, también en la sección de 2.4 GHz del apartado ‘Wireless Security’ se configuran las credenciales de autenticación. En este caso, se ha seleccionado mecanismo WPA2 Personal.

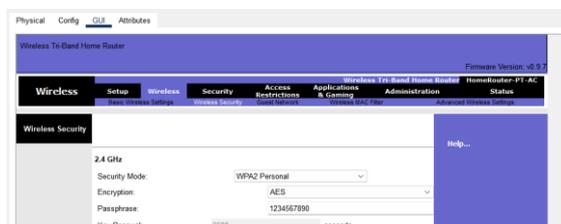


Ilustración 93. Escenario 2. Router Inalámbrico. GUI. Wireless Security

- Configuración del Smartphone:

Primeramente, se accede al apartado ‘IP Configuration’ del escritorio y se reinicia el servicio DHCP para que este le asigne una dirección IP dentro del rango establecido.

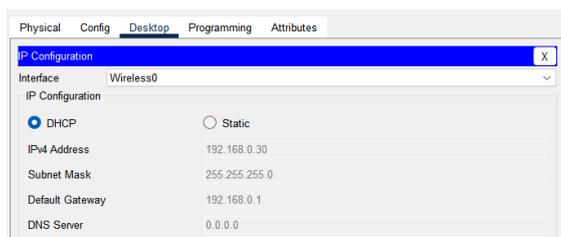


Ilustración 94. Escenario 2. Smartphone. IP Config

Luego, en la sección de ‘Wireless0’ de la configuración se ingresan el SSID y las credenciales de autenticación establecidas en el router inalámbrico.

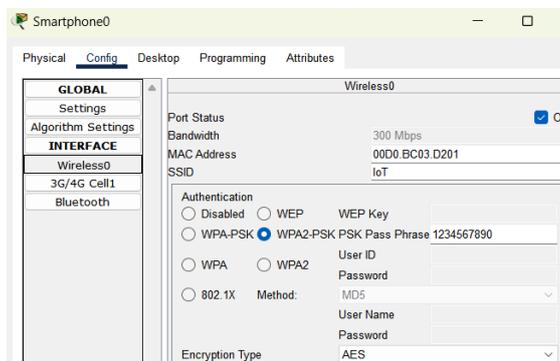


Ilustración 95. Escenario 2. Smartphone. Wireless0

- Configuración de los dispositivos:

Lo primero que se hace es acceder al apartado I/O Config de la configuración avanzada de los dispositivos y comprobar que el adaptador de red seleccionado es el inalámbrico.

Posteriormente, se accede al apartado ‘Wireless0’ de la configuración y se ingresan el SSID y las credenciales de autenticación establecidas en el router inalámbrico.

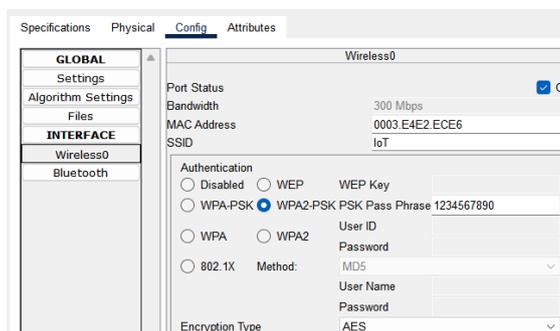


Ilustración 96. Escenario 2. Dispositivos. Wireless0

Y luego, en el apartado de ‘Settings’ de la configuración, se selecciona la opción de ‘Remote Server’ para el servicio IoT y se añaden la dirección y las credenciales del servicio Iot creado en el servidor.

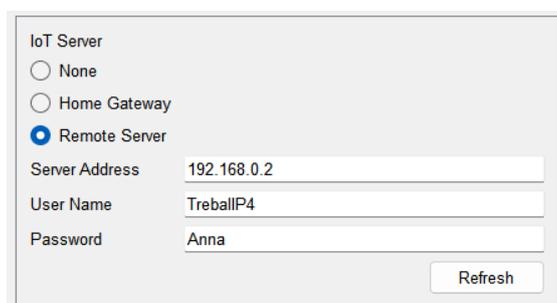


Ilustración 97. Escenario 2. Dispositivos. Servicio IoT

Por tanto, una vez configurados y conectados todos los dispositivos a internet y al servicio IoT creado, es posible controlarlos desde el mismo servidor y también desde el teléfono inteligente. Para ello, se accede al apartado 'Web Browser' o 'IoT Monitor' del 'Desktop' de estos dispositivos.



The screenshot displays a grid of IoT device control panels. Each panel includes a device name, its current status, and a control interface. The devices shown are:

- Water Level Monitor:** Water Level 12.1 cm.
- Motion Detector:** On.
- Door:** Open/Lock.
- Lawn Sprinkler:** Status.
- Garage Door:** On.
- Appliance:** On.
- Window:** On.
- Door:** Open/Lock.
- Humidifier:** Status.
- Door:** Open/Lock.
- Motion Detector:** On.
- Thermostat:** Status (Off/Cooling/Heating), Temperature 8.2 °C, Auto Cool Temperature N/A, Auto Heat Temperature N/A.
- Temperature Monitor:** Temperature 8.2 °C.
- Portable Music Player:** On.
- Fire Sprinkler:** Status.
- Furnace:** On.
- Smoke Detector:** Alarm Level 0.
- Window:** On.
- Bluetooth Speaker:** Connected/Playing.
- Ceiling Fan:** Status (Off/Low).
- Water Drain:** Status.
- Light:** Status (Off/Dim).
- Webcam:** On, Image view.
- Siren:** On.
- AC:** On.
- Motion Detector:** On.
- Humidity Sensor:** Humidity 70.9841 %.
- Carbon Monoxide Detector:** Alarm Level 0.
- Carbon Dioxide Detector:** Alarm Level 0.036.
- Automóvil Antiguo:** No remote control API.

Ilustración 98. Escenario 2. Smartphone. Servicio iot. Control de dispositivos

5.2.1 Placa SBC

A esta simulación, también se le ha añadido una tarjeta de programación SBC, junto con un botón para activarla. Esta, se ha programado de manera que al clicar sobre el botón se activen los dispositivos conectados a ella.

- En primer lugar, además del botón, se conectan a la tarjeta SBC la Lámpara, la Ventana 2, la Puerta 2, el Rociador de Fuego, el Desguace de Agua, el Ventilador y el Termostato. A este último, se le conectan el AC1 y el Aparato de calefacción a través de los puertos D2 y D1 respectivamente. Todas estas conexiones se realizan mediante cables personalizados IoT.

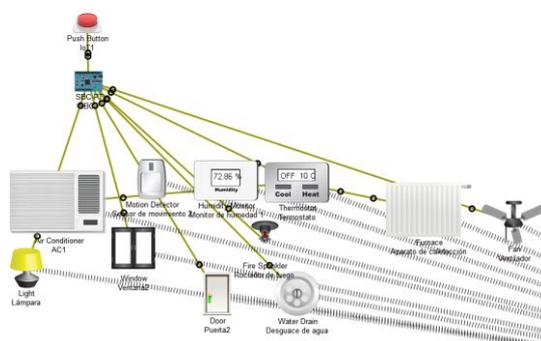


Ilustración 99. Escenario 2. Tarjeta SBC. Conexiones

- Luego, se programa la placa SBC a través del apartado de programación de esta, para que, al pulsar sobre el botón, se activen todos los dispositivos conectados a ella. En este caso, se ha decidido programar dicho dispositivo en Python.

Para ello, se crea un nuevo proyecto seleccionando la opción 'Empty-Phyton'.

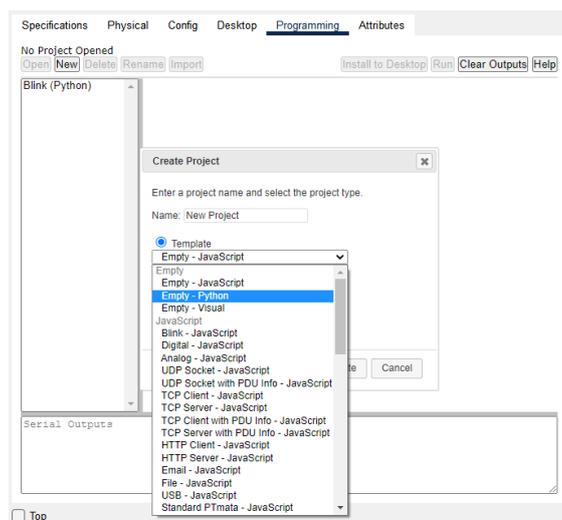


Ilustración 100. Escenario 2. Tarjeta SBC. Programación

- El código diseñado para programar dicha tarjeta es el siguiente:

```
from gpio import *
from time import *

def main():
    pinMode(0, OUT)
    pinMode(1, OUT)
    pinMode(2, OUT)
    pinMode(3, OUT)
    pinMode(4, OUT)
    pinMode(5, OUT)
    pinMode(6, OUT)
    pinMode(8, IN)

    while True:
        if digitalRead(8) == HIGH:
            customWrite(0, '1')
            customWrite(1, '2')
            customWrite(2, '2')
            customWrite(3, '1')
            customWrite(4, '1')
            customWrite(5, '3')
            customWrite(6, '1')

        else:
            customWrite(0, '0')
            customWrite(1, '0')
            customWrite(2, '0')
            customWrite(3, '0')
            customWrite(4, '0')
            customWrite(5, '0')
            customWrite(6, '0')
            delay(500)

if __name__ == "__main__":
    main()
```

Tabla 1. Escenario 2. tarjeta SBC. Código programación

Los valores asociados a cada dispositivo dependen del puerto de la tarjeta al que estén conectados. En este caso los dispositivos del hogar están conectados del 0 al 6 y son pines de salida, mientras que el botón está conectado al puerto 8 de la tarjeta y es un pin de entrada.

Hay que consultar en las especificaciones de cada dispositivo los posibles valores que pueden adoptar cada uno de ellos. En este caso:

- El botón puede tomar los valores 'LOW (no pulsado)' o 'HIGH' (pulsado).
- La lámpara puede adoptar los valores de estado 0 (apagado), 1 (atenuado) o 2 (encendido).



- La ventana puede tener los valores de estado 0 (cerrada) o 1 (abierta).
- La puerta puede tomar los valores 0 (cerrada) o 1 (abierta).
- El desguace de agua puede adoptar los valores de estado 0 (cerrado) o 1 (abierto).
- El rociador de fuego puede tener los valores 0 (cerrado) o 1 (abierto).
- El termostato puede tomar los valores 0 (apagado), 1 (enfriamiento), 2 (calefacción) o 3 (automático).
- El ventilador puede adoptar los valores de estado 0 (apagado), 1 (velocidad baja) o 2 (velocidad alta).

Primero se importan las librerías ‘gpio’ y ‘time’ y se define un proyecto main. Luego, se crea un bucle While infinito en el que se establece la siguiente condición: si el valor que se lee en el pin de lectura asociado al botón (8) es igual a ‘HIGH’, entonces se escribe en los demás pines de salida los valores asociados al encendido de cada dispositivo. Y, en caso de no cumplirse esta condición, si el valor que se lee en el pin de entrada es ‘LOW’, entonces se escribe en los demás pines los valores asociados al apagado de los dispositivos.

5.2.2 Placas MCU

Además, también se han añadido a esta simulación dos tarjetas MCU junto con dos sensores de movimiento para activarlas.

- Para empezar, a la primera tarjeta se conectan mediante cables personalizados IoT, la Webcam, la Puerta 1, la Ventana 1 y el Humidificador, mientras que a la segunda se conectan la Puerta 3, el Aspersor, la Cafetera, la Puerta del Garaje y la Sirena, también con el mismo tipo de cable.

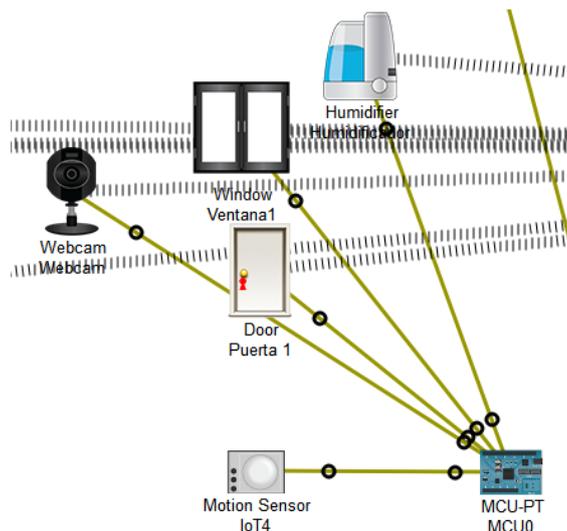


Ilustración 101. Escenario 2. MCU0. Conexiones

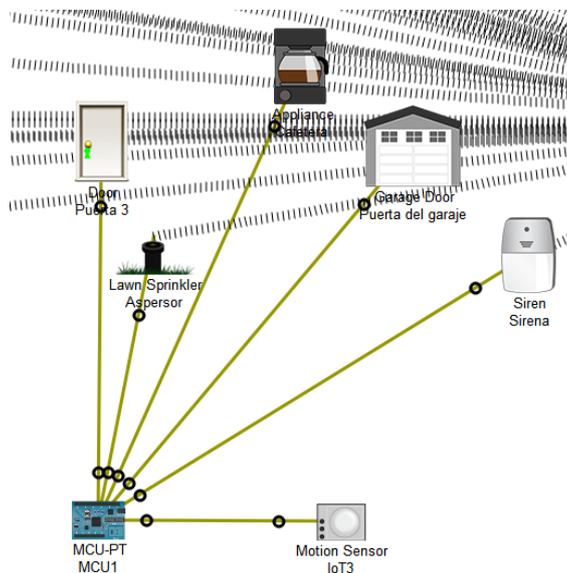


Ilustración 102. Escenario 2. MCU1. Conexiones

- Luego, se programan ambas placas MCU a través del apartado de programación de estas, para que, al pulsar sobre el sensor de movimiento correspondiente, se activen los dispositivos conectados a cada una de ellas. En este caso, se ha decidido programar los dispositivos en Blockly; un tipo de lenguaje de programación visual cuya interfaz se basa en bloques de arrastrar y soltar.

Para ello, se crea un nuevo proyecto seleccionando la opción ‘Blink-Visual’.

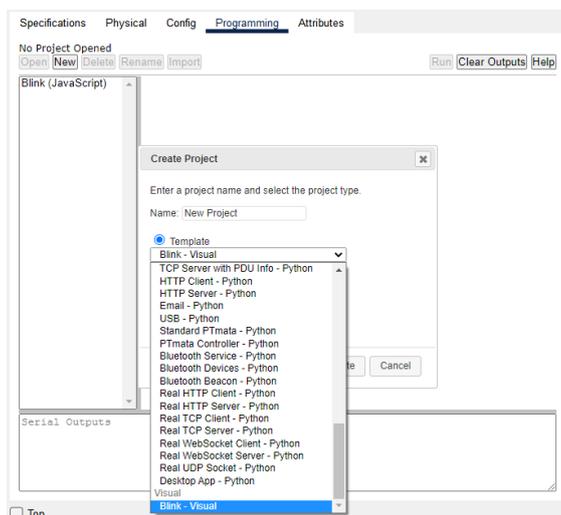


Ilustración 103. Escenario 2. MCUs. Programación

- Los bloques diseñados para programar dichas tarjetas son los siguientes:

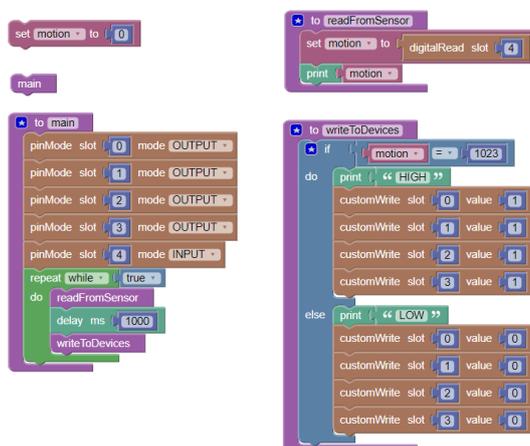


Ilustración 104. Escenario 2. MCU0. Código bloques

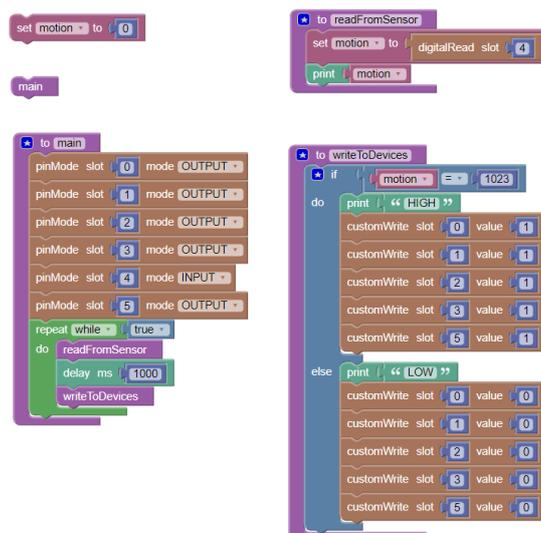


Ilustración 105. Escenario 2. MCU1. Códigos bloques

Estos códigos se comportan prácticamente de la misma manera que el diseñado en Python en el apartado anterior para la tarjeta SBC. En este caso, para la placa MCU0 los dispositivos del hogar están conectados del puerto 0 al 3 y son pines de salida, mientras que el sensor de movimiento está conectado al puerto 4 de la tarjeta y es un pin de entrada. Para la placa MCU1, los dispositivos del hogar están conectados a los puertos 0, 1, 2, 3 y 5, mientras que el sensor de movimiento está conectado al 4.

En este caso, los posibles valores que puede adoptar cada dispositivo son:

- El sensor de movimiento puede tomar los valores ‘LOW (no pulsado)’ o ‘HIGH’ (pulsado).
- La cámara puede adoptar los valores de estado 0 (apagada), 1 (encendida).
- La ventana puede tener los valores de estado 0 (cerrada) o 1 (abierta).
- La puerta puede tomar los valores 0 (cerrada) o 1 (abierta).
- El humidificador puede adoptar los valores de estado 0 (cerrado) o 1 (abierto).
- El aspersor puede tener los valores 0 (cerrado) o 1 (abierto).
- La cafetera puede tomar los valores 0 (apagada) o 1 (encendida).
- La puerta del garaje puede adoptar los valores de estado 0 (cerrada), 1 (abierta).
- La sirena puede tener los valores 0 (apagada) o 1 (encendida).

Primero se crea una variable denominada ‘motion’ y se establece su valor a 0. Luego se define el proyecto main y dentro de este los pines de entrada y salida asociados a cada dispositivo. Posteriormente, se añade un bucle While infinito que realiza las funciones ‘readFromSensor’ y ‘writeToDevices’ desarrolladas previamente con un retardo entre estas de 1000 ms.

La función readFromSensor asocia la variable ‘motion’, creada anteriormente, al pin de entrada del sensor de movimiento (4), e imprime el valor de esta. Y, la función writeToDevices está formada por una condición asociada al valor de la variable ‘motion’: si este valor es igual a 1023 (HIGH), esto implica que el sensor de movimiento está activo, y, por tanto, se escribe en los pines de salida los valores asociados al encendido de cada dispositivo. Pero, en caso de que la variable ‘motion’ sea igual a 0 (LOW), el sensor de



movimiento está inactivo y los valores que se escriben en los pines son los asociados al apagado de los dispositivos.

Capítulo 6. RESULTADOS

En el Escenario 1, al activarse los dispositivos tras pulsar sobre los sensores de movimiento, podemos observar cómo se actualizan los valores de los monitores existentes.

El Monitor del nivel del agua se enciende e incrementa su valor al activar los sensores de movimiento 2 y 3, ya que estos activan el rociador de fuego y el aspersor, respectivamente. En cambio, este monitor se apaga y toma el valor de 0 al activarse el sensor de movimiento 1, ya que este abre el desague de agua y esta desaparece a través de él.

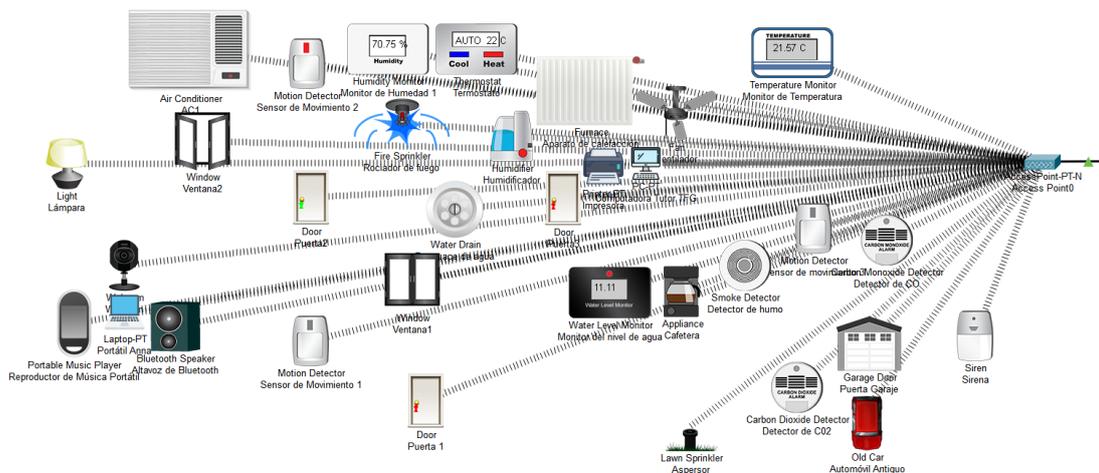


Ilustración 106. Escenario 1. Monitor del nivel del agua. Sensor Mov 2 ON

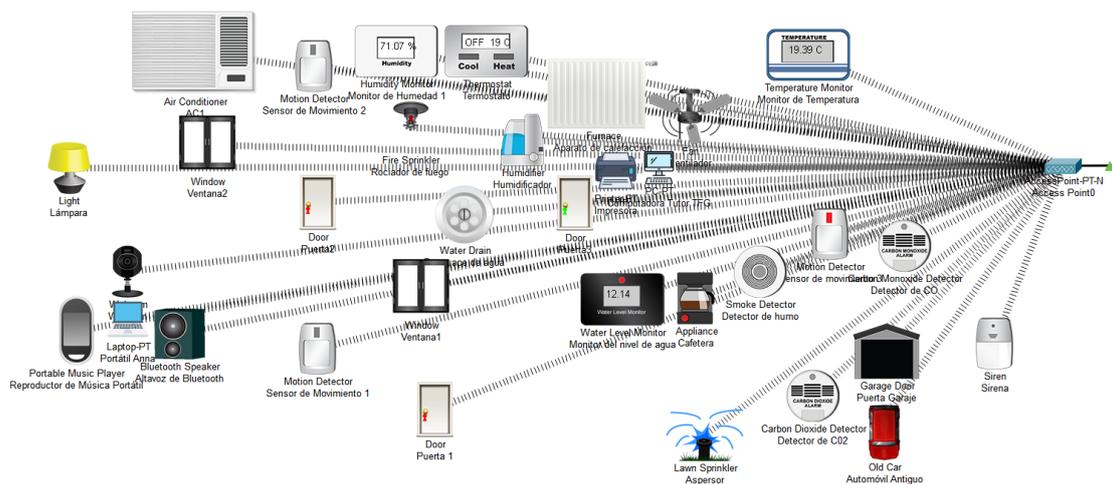


Ilustración 107. Escenario 1. Monitor del nivel del agua. Sensor Mov 3 ON

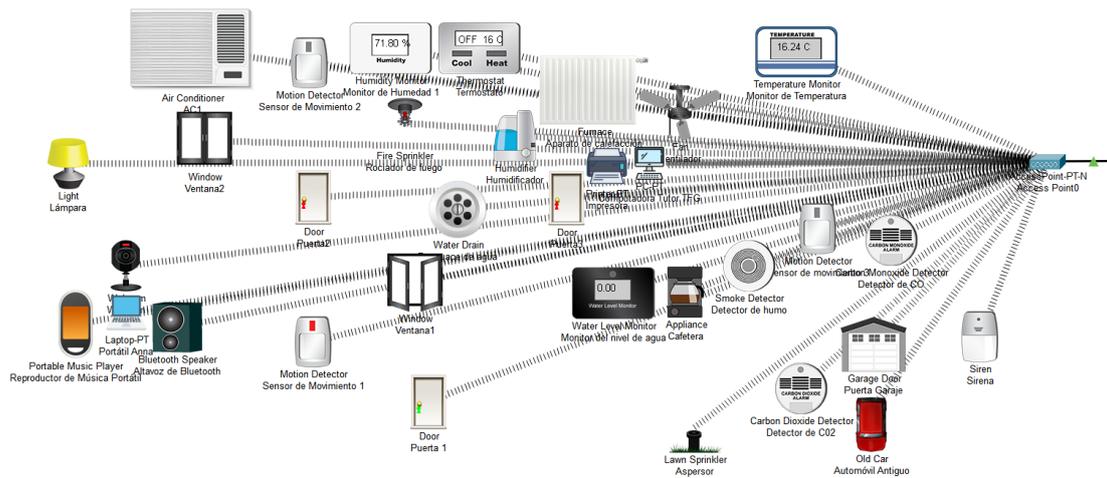


Ilustración 108. Escenario 1. Monitor del nivel del agua. Sensor Mov 1 ON

El monitor de humedad varía su valor al activar cualquiera de los 3 sensores de movimiento, ya que todos ellos encienden dispositivos que afectan a dicho valor, como las puertas y ventanas, el humidificador, el termostato, el ventilador...

En cambio, el valor contenido en el monitor de humedad coincide con el valor automático que establezca el termostato.

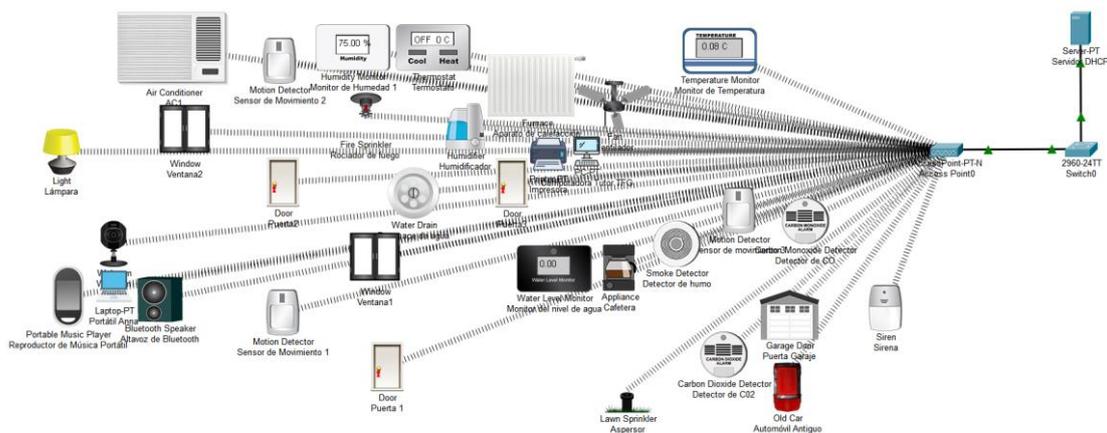


Ilustración 109. Escenario 1. Monitores de humedad y temperatura. Sensores desactivados

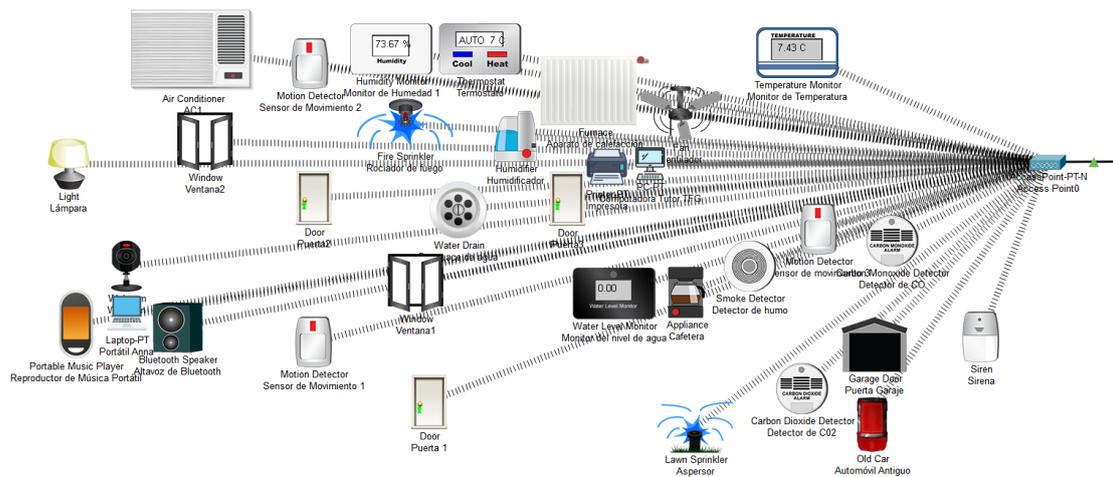


Ilustración 111. Escenario 1. Monitores de humedad y temperatura. Sensores activados I

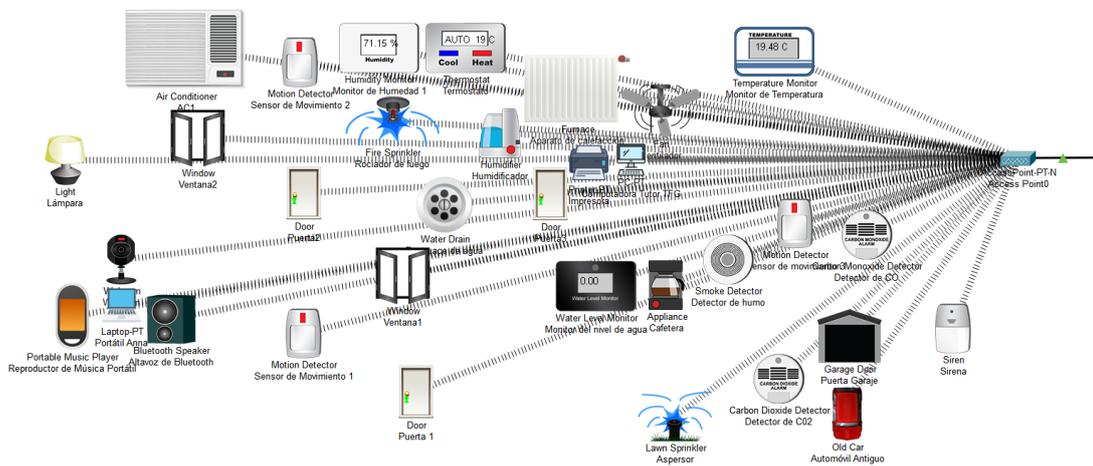


Ilustración 110. Escenario 1. Monitores de humedad y temperatura. Sensores activados II

Al establecerse dichas condiciones, estas predominan sobre el control que el servidor tiene de los dispositivos y, por tanto, ya no dispone de dicho control.

En cuanto al servicio DNS de correo electrónico creado, tras seguir el procedimiento comentado y realizar las configuraciones descritas, ambos dispositivos pueden comunicarse a través del él.

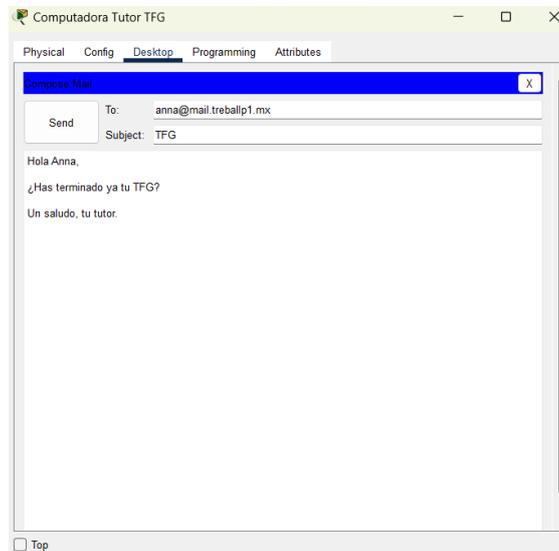


Ilustración 112. Escenario 1. Email. Envío correo I

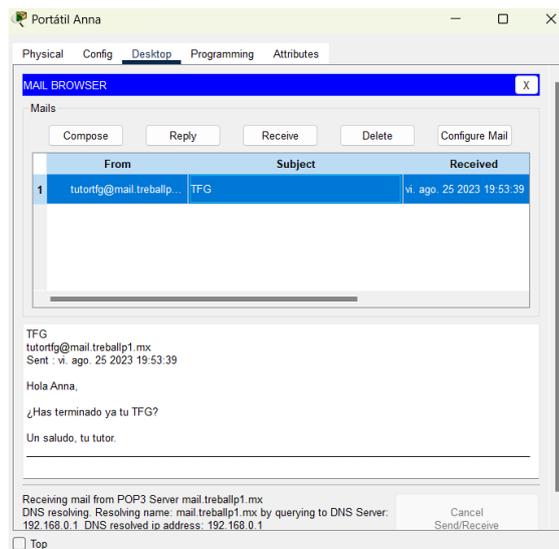


Ilustración 113. Escenario 1. Email. Recepción correo I

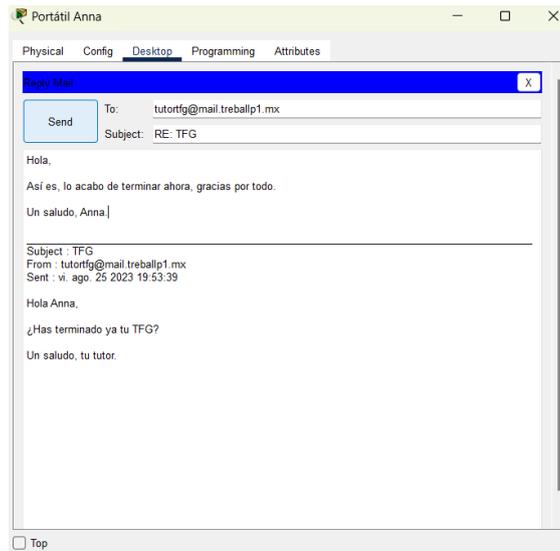


Ilustración 115. Escenario 1. Email. Envío correo II

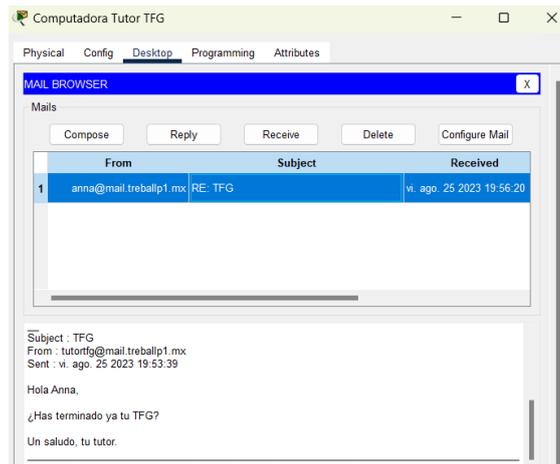


Ilustración 114. Escenario 1. Email. Recepción correo II

Las siguientes imágenes prueban el funcionamiento de las placas SBC y MCU que se han añadido al Escenario 2:

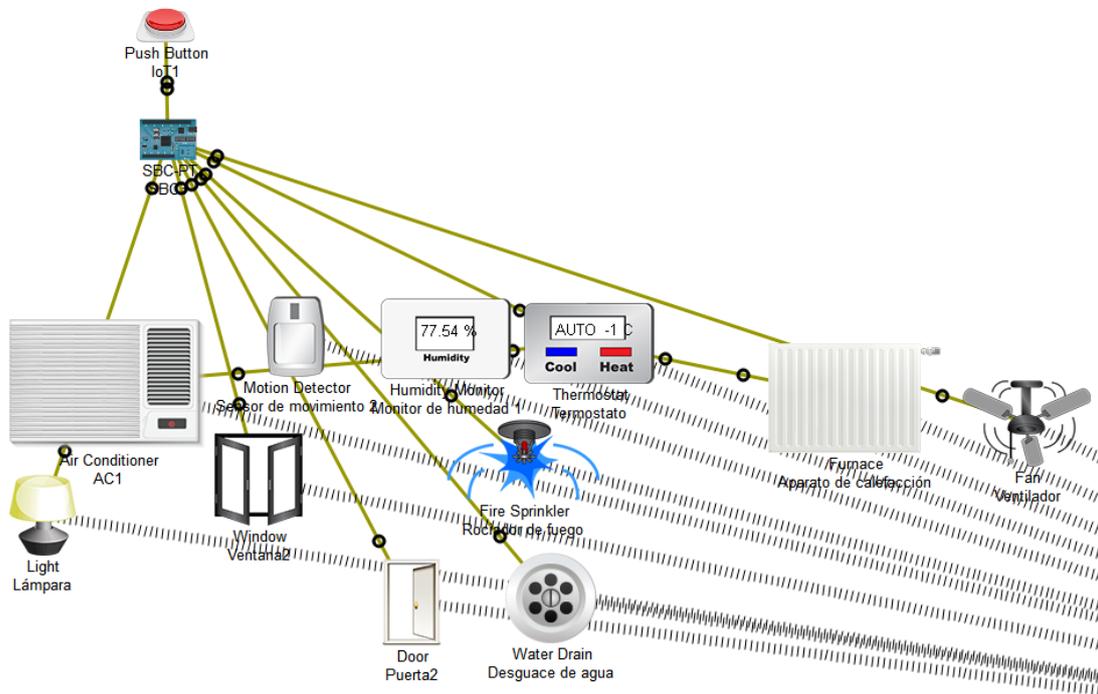


Ilustración 116. Escenario 1. SBC. Botón activado

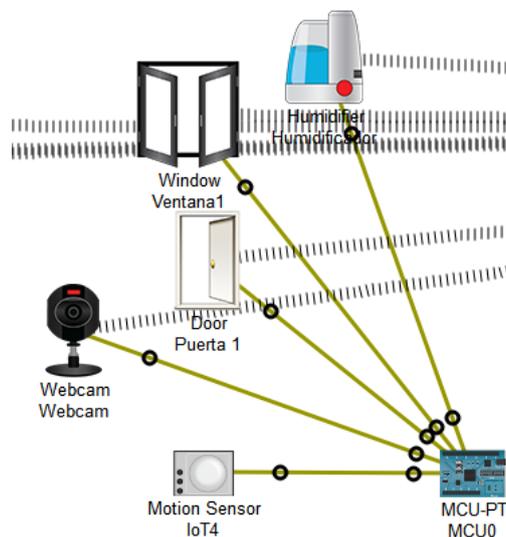


Ilustración 117. Escenario 2. MCU0. Sensor activado

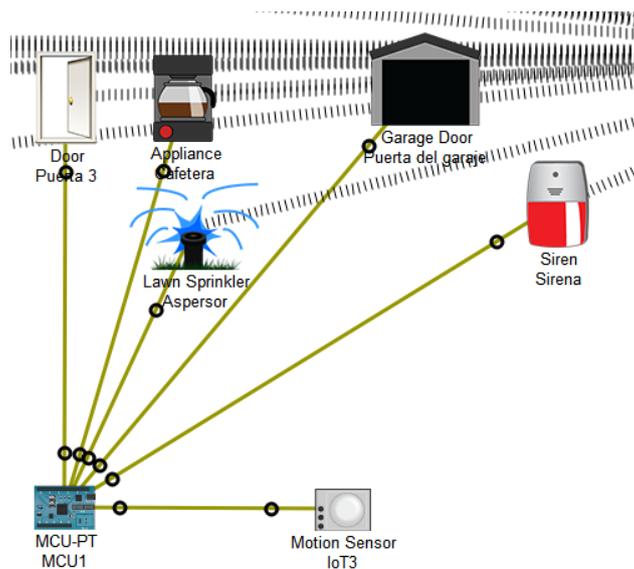


Ilustración 118. Escenario 2. MCU1. Sensor activado

Cabe destacar que, en este caso, la programación de las placas no predomina sobre el control del teléfono inteligente y del servidor. Si, por ejemplo, desde uno de estos dos dispositivos de control se bloquean las puertas, éstas no se abrirán al activar el botón o los sensores de movimiento.



Ilustración 119. Escenario 2. Bloqueo de puertas

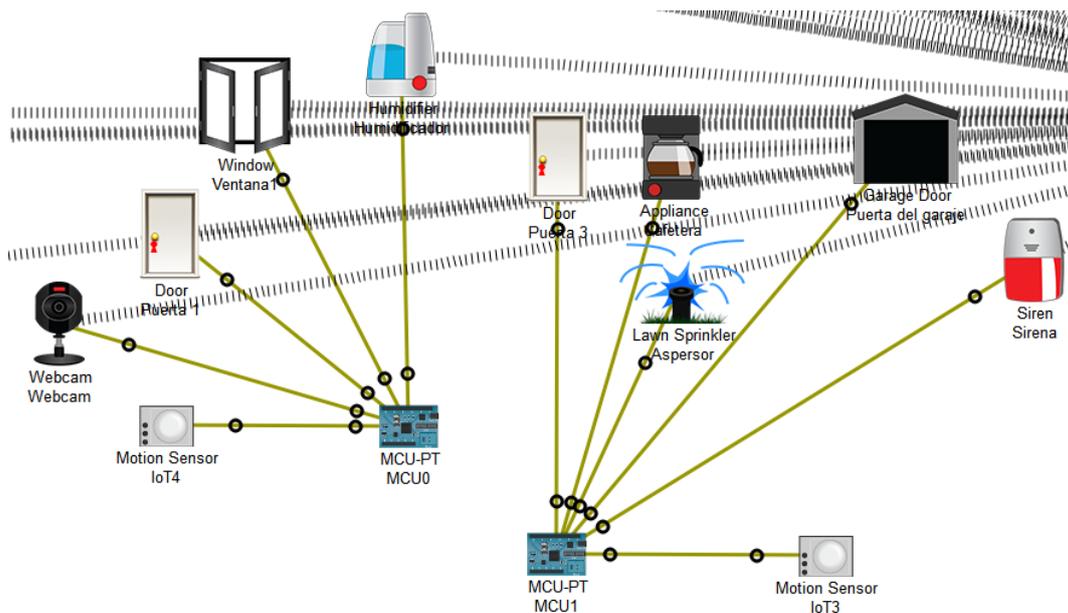


Ilustración 121. Escenario 2. Puertas 1 y 3 bloqueadas

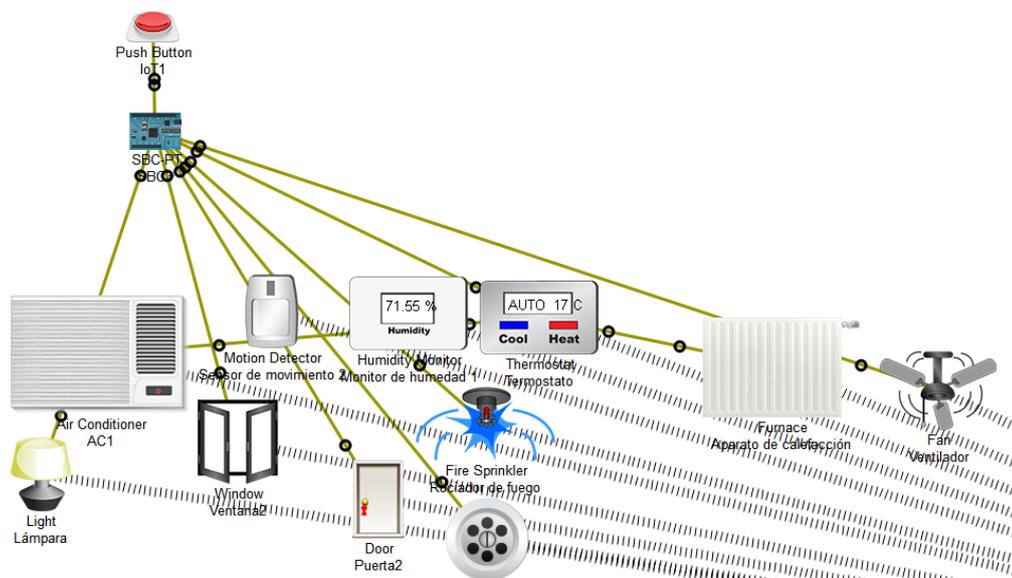


Ilustración 120. Escenario 2. Puerta 2 bloqueada



Capítulo 7. CONCLUSIONES

En este trabajo se ha abordado la creación, configuración y administración de redes IoT utilizando la herramienta de simulación Cisco Packet Tracer, enfocado en la administración de dispositivos y la implementación de servicios personalizados. A través de los dos escenarios implementados, se han explorado diversos aspectos de la configuración de redes, desde la asignación de direcciones IP hasta la automatización de dispositivos IoT y la programación de microcontroladores.

A lo largo del desarrollo de este proyecto se ha hecho patente la importancia de planificar cada paso, resolver rápidamente los problemas y prestar especial atención a todos los detalles. Y también se ha evidenciado la amplia gama de posibilidades para mejorar la eficiencia y funcionalidad de las redes que ofrece la capacidad de poder personalizar los servicios. Además, se ha resaltado la importancia de la seguridad y la gestión de recursos dentro de una red.

En resumen, ha sido una experiencia enriquecedora haber trabajado en este proyecto y considero que los conocimientos proporcionados son esenciales en el campo de las tecnologías de la información.

7.1 Ampliación futura

Una posible ampliación futura de este proyecto podría ser la implementación de medidas de seguridad más avanzadas mediante la configuración de firewalls, detección de intrusiones o políticas de seguridad más robustas.

Esta ampliación también podría centrarse en la adaptación de las redes simuladas a situaciones del mundo real, llevando los proyectos y las simulaciones a un nivel más práctico y aplicado.



Capítulo 8. BIBLIOGRAFÍA

- [1] <https://packettracer.network/packet-tracer-8-0-configurar-dispositivos-de-iot/>
- [2] <https://iot4beginners.com/how-to-simulate-iot-projects-using-cisco-packet-tracer/>
- [3] <https://www.scirp.org/journal/paperinformation.aspx?paperid=108495>
- [4] <https://www.youtube.com/watch?v=mAhkSQIS79g>
- [5] <https://www.youtube.com/watch?v=m1-3Ri6tNoo>
- [6] https://www.youtube.com/watch?v=_P3eUxcW664&t=1s
- [7] https://www.youtube.com/watch?v=pCoQE_gg-xk
- [8] <https://www.youtube.com/watch?v=N7QelVF4Ifc>