
This is the **published version** of the bachelor thesis:

Pariante Perea, Daniel; Franco Puntos, Daniel, dir. Hub de sensores y controladores IoT con analítica Cloud y control desde app móvil. 2021. (958 Enginyeria Informàtica)

This version is available at <https://ddd.uab.cat/record/257799>

under the terms of the  license

Hub de sensores y controladores IoT con analítica Cloud y control desde app móvil

Daniel Pariente Perea

Resumen — Este trabajo busca crear una automatización en un entorno doméstico, el cual debe de ser capaz de realizar tareas de forma automática a partir de ciertos eventos en su entorno. Dependiendo del tipo de entorno donde se utilice, el sistema tendrá unas tareas diferentes con respecto a las anteriores. Para realizar la automatización de tareas, se utilizarán diferentes sensores y aplicaciones, los cuales permitirán obtener información a través de los sensores, tratarla y realizar acciones mediante las distintas aplicaciones, todo a través del repositorio cloud, que servirá como enlace entre sensores y las aplicaciones móviles.

Palabras clave — Hub, sensores, analítica, cloud, automatización, app móvil.

1 INTRODUCCIÓN

La tecnología ha permitido avanzar hasta el punto de crear sistemas domóticos sencillos, mediante los cuales se pueden controlar desde casas domésticas hasta grandes factorías. Mediante estos sistemas, es posible recoger todo tipo de información, y utilizarla para casi cualquier beneficio que se pueda imaginar.

Debido a la sencillez de estos sistemas, el uso de esta tecnología ha aumentado de una forma considerable, permitiendo la especialización de algunas empresas en ofrecer este tipo de servicios. Estas empresas suelen ofrecer sistemas propietarios, o sistemas de suscripción para ofrecer la domotización, como puede suceder con empresas de alarmas, las cuales ofrecen la instalación del sistema propietario y ofrecer el servicio con su propio hardware.

Esto genera una problemática, y es la dependencia de la empresa para utilizar el sistema, el cual no se puede adaptar a nuevas necesidades o añadir nuevos dispositivos si no se hace a través de la empresa.

De esta manera, existen diferentes opciones de sistemas con software y hardware libre, mediante el cual se puede desarrollar un entorno domótico, el cual se puede modificar y adaptar al entorno, con la posibilidad de utilizar el hardware deseado, permitiendo libertad a la hora de diseñar, cambiar o modificar el sistema (o parte de él), y reduciendo de una amplia forma el coste de la domotización deseada.

2. OBJETIVOS

2.1 Objetivo principal

El objetivo principal de este proyecto es dar solución a este problema, desarrollar y comparar un sistema open-source domótico con capacidad de automatización, ofreciendo una alternativa a los sistemas cerrados ofrecidos por empresas dedicados a ellos, y mostrando la viabilidad

de dicha alternativa.

Se han utilizado dos sistemas domóticos para la automatización de tareas, los cuales son Home Assistant y OpenHab, ya que ambos son sistemas *open source*, y ofrecen una gran variedad plug-in y un gran soporte por parte de ambas comunidades.

También se ha comprobado el grado de automatización con herramientas externas al sistema, como es el caso de IFTTT junto al sistema HomeAssistant, que ha permitido guardar un registro de movimientos cerca del sensor en una hoja de Google Sheets.

2.2 Objetivos específicos

1. Existen objetivos más específicos que han permitido un desarrollo correcto del proyecto, son los siguientes:
2. Definir beneficios del proyecto respecto a alternativas propietarias.
3. Comprobar el funcionamiento correcto de los sensores.
4. Configurar el sistema con OpenHAB y desarrollar la automatización de tareas desde este entorno.
5. Configurar el sistema con Home Assistant y desarrollar la automatización de tareas desde este entorno.
6. Ofrecer una comparación del sistema desarrollado respecto a alternativas de domotización propietarias.
7. Integrar al sistema una opción para visualizar los datos mediante el cloud a través de Microsoft Portal Azure.
8. Desarrollar un método para tener acceso y control desde un dispositivo móvil fuera de la red local del sistema mediante los plug-in de HomeAssistant y al servicio de DuckDNS.
9. Desarrollar una aplicación móvil a través de una herramienta low-code, como Microsoft Power-Apps.
10. Desarrollar una documentación de las aplicaciones y de las herramientas utilizadas en el trabajo en una

plataforma on-line como GitHub.

3. METODOLOGÍA Y PLANIFICACIÓN

Para este trabajo se ha utilizado una metodología Kanban, una metodología visual y ágil, que ha permitido ver el estado actual del proyecto y de las tareas, tanto en progreso como por realizar y finalizadas.

Como herramienta, se ha utilizado la opción gratuita de Trello, que permite la creación de tableros con tarjetas, donde con los nombres de "To do", "In Progress" y "Done" en los títulos de las columnas, se puede emular un tablero Kanban.

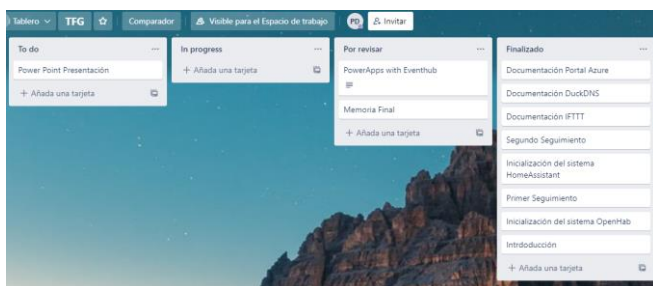


Imagen 1: Tablero Kanban en Trello.

Aproximadamente cada semana se han tenido reuniones con el tutor del trabajo, para comentar el estado de las tareas en progreso, comentar las siguientes tareas por realizar y obtener feedback.

Para marcar las fechas con las tareas y poner umbrales de tiempo, se han utilizado las entregas de los informes.

1. Preparar los sistemas OpenHab y HomeAssistant y conectar los sensores y actuadores al sistema. (10/10/2021)
2. Crear las primeras automatizaciones del sistema. (7/11/2021)
3. Implementar la herramienta DuckDNS y crear su documentación. (14/11/2021)
4. Implementar la herramienta IFTT y crear su documentación. (5/12/2021)
5. Enviar los datos a la plataforma Portal Azure y crear su documentación. (19/12/2021)
6. Crear la aplicación con Power Apps y conectarla con la plataforma Portal Azure. (10/1/2022)
7. Redactar el informe final. (23/1/2022)
8. Preparar la presentación (8/2/2022)

4. PROPUESTA DE SOLUCIÓN

4.1 Objetivos principales

Como ha quedado reflejado en el apartado de los objetivos, el principal objetivo del proyecto es la automatización

de un entorno domótico a través de los sistemas OpenHab y Home Assistant, el tratamiento de los datos obtenidos a través de los sensores y las automatizaciones y realizar un estudio de ambos sistemas.

El objetivo principal se ha conseguido, debido a que se ha podido generar un sistema domótico que, desde la obtención de datos por parte de los sensores, pasando por el sistema de domotización HomeAssistant, se ha conseguido generar automatizaciones a nivel del sistema con sus propias reglas (también en OpenHab), como también se ha conseguido generar automatizaciones a través de herramientas externas (a través del servicio IFTTT), y también se ha conseguido mandar y mostrar los datos en una herramienta cloud como Microsoft Azure.

Sobre los objetivos específicos, se han podido desarrollar una gran mayoría, con la excepción de los puntos 7 y 9.

Sobre el punto 7, se ha conseguido mandar la información recibida de los sensores a través del sistema Home Assistant a la plataforma Portal Azure, pero no ha sido posible trabajar con los datos almacenados.

Como se puede observar en la imagen siguiente, los datos son obtenidos por la plataforma, pero no se ha encontrado la forma para trabajar ni la forma de extraerlos, debido a que la única opción que parece viable es aumentar la suscripción a una categoría superior a la que se ha estado usando hasta el momento.

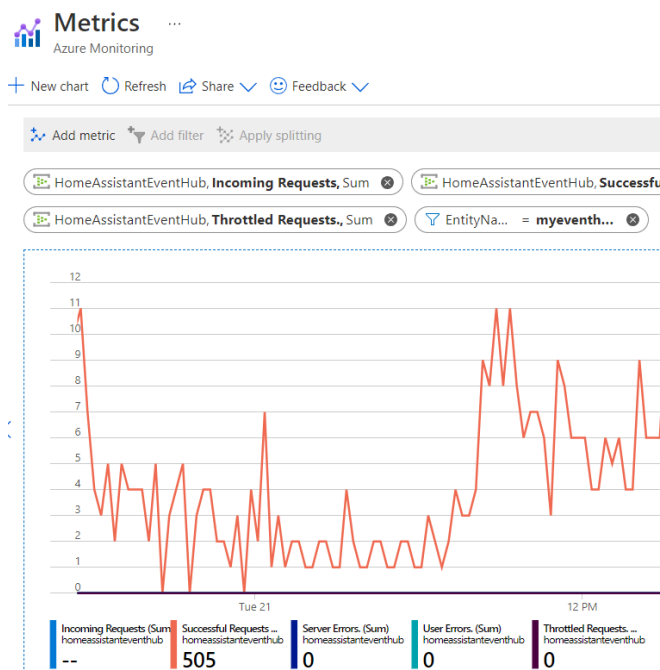


Imagen 2: Métricas de Portal Azure.

Sobre el punto 9, se ha creado la aplicación móvil a través de Microsoft Power-Apps como estaba planificado,

pero no se ha podido realizar el objetivo principal de esta, ya que era la inclusión de los datos obtenidos a través del Portal Azure y mostrarlos en la aplicación, mostrando lo que podría haber sido una integración total con las herramientas de Microsoft, y la ejecución completa del ciclo, desde el sensor hasta la aplicación pasando por Portal Azure.

Como alternativa, se ha creado una aplicación multiplataforma con Power Apps donde se ha aprovechado la integración que este tiene con Google Sheets, y se ha utilizado la hoja de cálculo que se ha creado a través de la automatización mediante IFTTT. De esta manera, se ha creado un ciclo completo desde el sensor hasta una aplicación, pasando por el sistema de Home Assistant y la automatización de IFTTT, pese a ser diferente a la propuesta principal, ya que no se utiliza Portal Azure en esta cadena completa. En la siguiente foto se puede observar el resultado de la aplicación creada mostrando los datos de esta hoja de cálculo de Google.

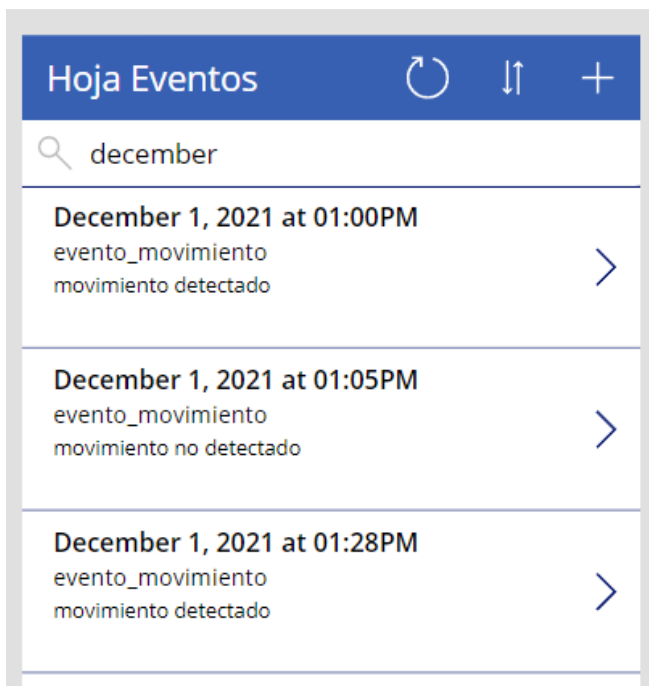


Imagen 3: Aplicación creada mediante PowerApps.

4.2 Dispositivos hardware y conexiones

Para la realización del trabajo se ha utilizado el siguiente Hardware facilitado por el tutor:

- Raspberry Pi 3 model B+.
- Aeotec Z-Stick Gen5.
- Aeotec Door/Window Sensor 7.
- Aeotec MultiSensor 6.
- Aeotec Dual Nano Switch.
- 2 tarjetas microSD.

Para la conexión con la Raspberry y el sistema, se ha

utilizado el siguiente router: Router Livebox Fibra de Jazztel.

Como actuador junto al Dual Nano Switch, se ha utilizado una lámpara básica, para comprobar que, al activarse la automatización, la lámpara se enciende. La conexión a la toma de corriente se ha realizado de la siguiente manera:

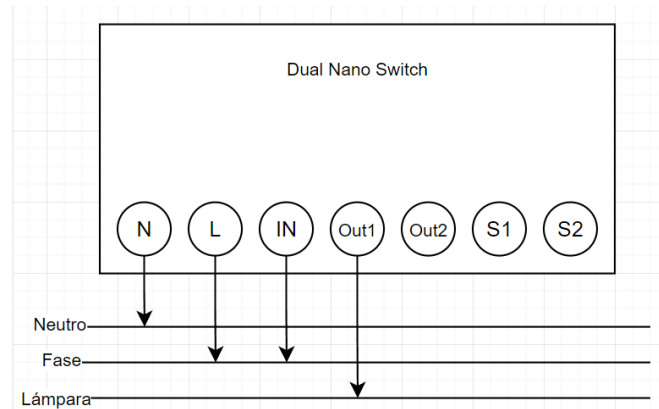


Imagen 4: Conexiones de Dual Nano Switch.

Antes de conectar el Z-Stick a la Raspberry, se deben de emparejar con los demás sensores y actuadores para poder utilizarlos. Para hacerlo, se debe de apretar el botón hasta que la luz se parpadee en azul, y una vez lo este, emparejar a los demás sensores y actuadores mediante los botones correspondientes a cada uno.

Una vez están emparejados todos los sensores y actuadores, el Z-Stick se conecta a la Raspberry Pi mediante el puerto USB, y a su vez, la Raspberry se conecta al router para obtener internet a través de un cable de red.

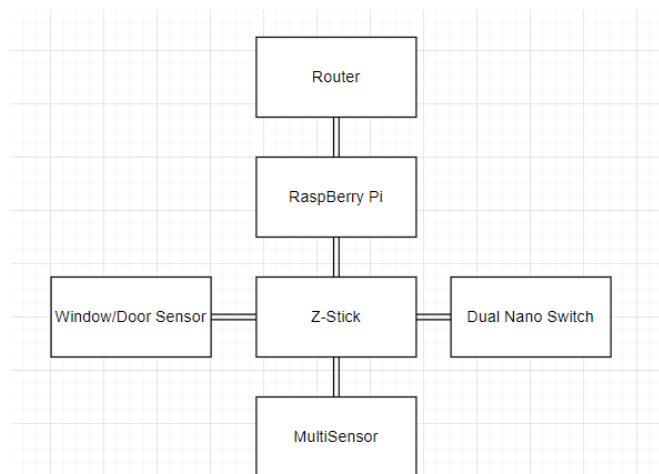


Imagen 5: Conexiones Hardware.

4.3 Diseño e instalación del sistema

Para la instalación del sistema doméstico (tanto OpenHab como Home Assistant), como primer paso se ha de

instalar el sistema en la tarjeta MicroSD para la Raspberry. Para hacer esta tarea se ha utilizado el programa Balena Etcher. Se han descargado las imágenes de OpenHab y HomeAssistant disponibles en sus respectivas páginas oficiales.

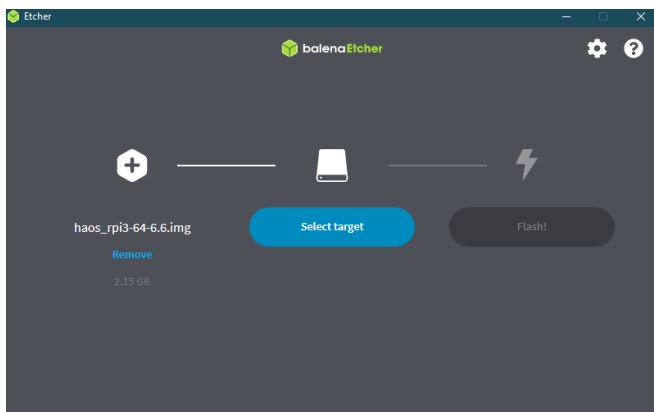


Imagen 6: Balena Etcher.

El sistema sigue una estructura como la que se muestra en el siguiente esquema:

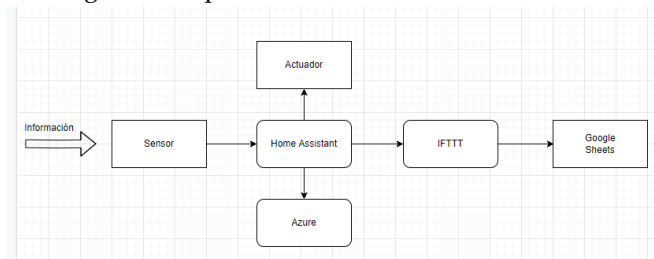


Imagen 7: Esquema del sistema.

Como se puede observar, el nexo es el sistema domótico (en este caso, HomeAssistant), ya que funciona como nexo entre los sensores y la automatización, o entre los sensores y IFTTT, o los sensores y Portal Azure.

4.4 Desarrollo e implementación

Como método de documentación para este trabajo, se ha utilizado la plataforma [Github](#), y se hará referencia en este apartado.

Se ha decidió utilizar esta plataforma tanto para aligerar la carga del anexo de este trabajo, como para almacenar públicamente los datos del sistema en una plataforma popular.

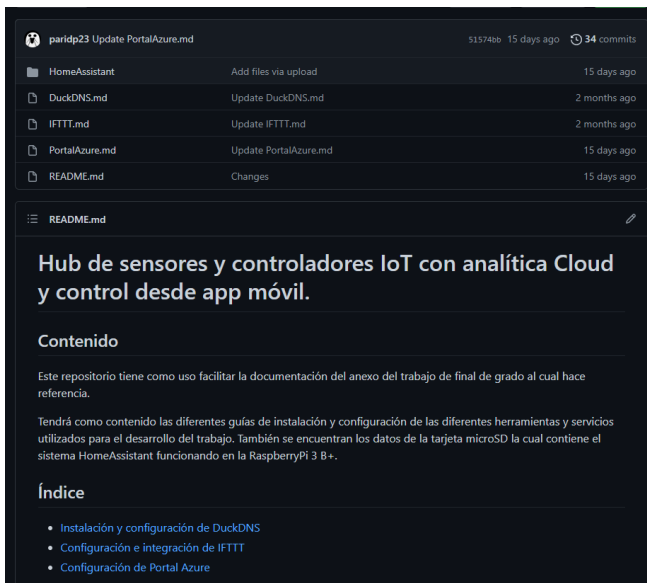


Imagen 8: Página principal del proyecto de Github.

Para el funcionamiento del Z-Stick con OpenHab y HomeAssistant, se ha de instalar el addon Z-Way-Binding y Z-Wave JS en cada uno de los sistemas respectivamente.

Una vez instalado este Add-On, el sistema ya recibe los datos de los diferentes sensores a través del Z-Stick. En el caso de Home Assistant, no será necesario hacer nada, debido a que se configuran de forma automática. Con OpenHab, se deben de crear los buses necesarios para cada sensor.

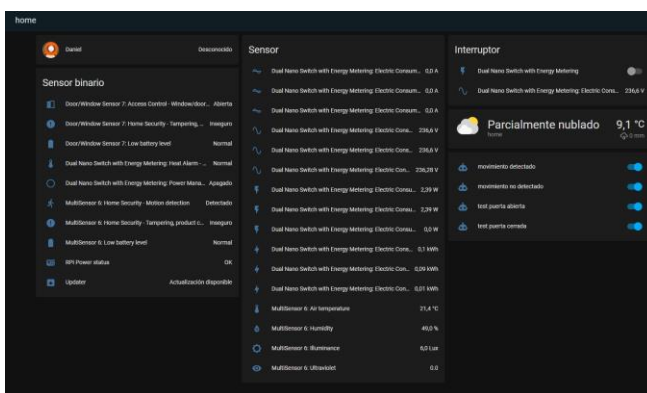


Imagen 9: Página principal de HomeAssistant.

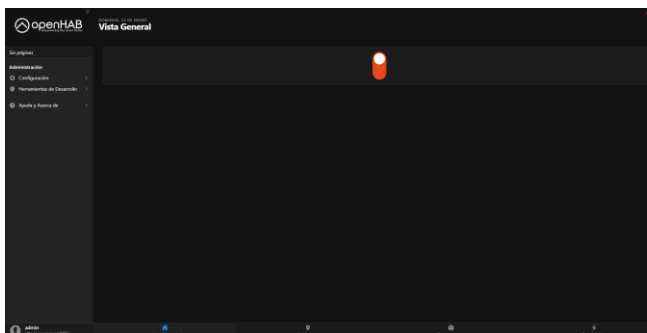


Imagen 10: Página principal de OpenHab.

4.4.1 Automatizaciones del sistema

Para la creación de las automatizaciones básicas, se han realizado utilizando las herramientas que los propios sistemas facilitan.

En el caso de OpenHab, primero se deben de pasar los sensores y actuadores de Things, que es como primero se reconocen, a ítems, para poder generar la automatización:

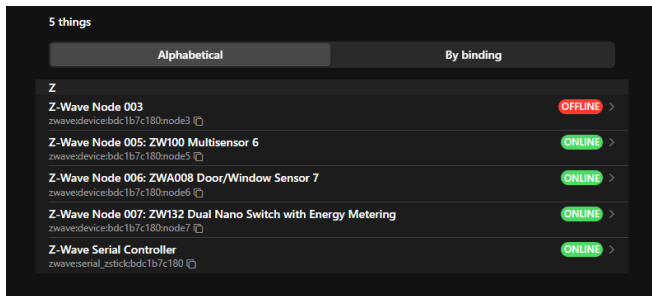


Imagen 11: Aparatado de Things.

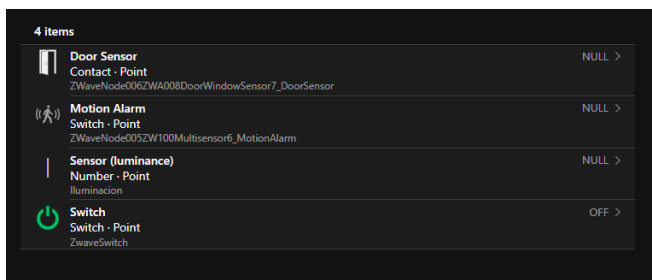


Imagen 12: Apartado de Items.

Una vez tenemos los sensores y actuadores como Items, se generan las reglas de automatización deseadas, como la que se muestra a continuación, que es la apertura de la puerta y el encender el actuador cuando esto sucede.

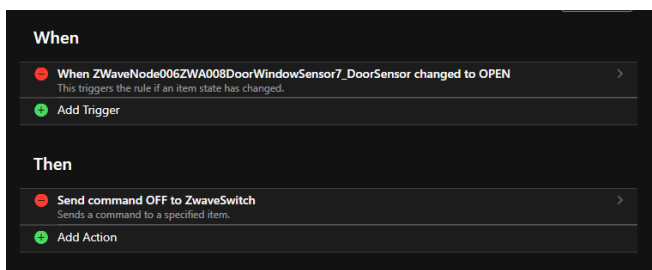


Imagen 13: Automatización en OpenHab.

En el caso de HomeAssistant las automatizaciones son más sencillas de configurar que en el caso anterior, ya que los sensores y actuadores se definen como entidades de forma automática, y no es necesario tratarlos como en OpenHab.

En el propio apartado de Automatizaciones del sistema se definen los desencadenantes y las acciones, como se muestran en las siguientes dos imágenes:

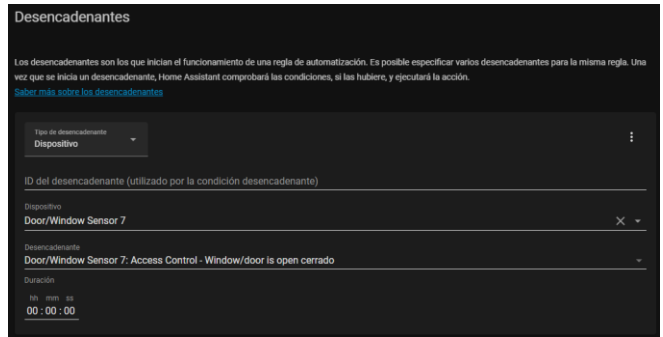


Imagen 14: Desencadenantes.

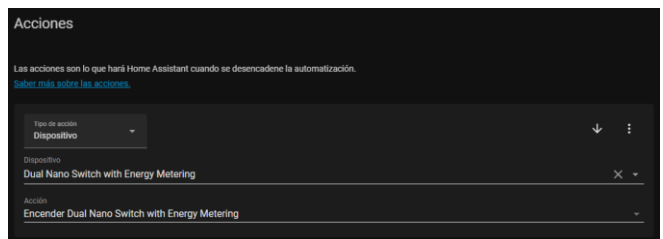


Imagen 15: Acciones.

4.4.2 IFTTT

También se han añadido las automatizaciones mediante la herramienta externa al sistema IFTTT. La configuración de este servicio se puede consultar en la documentación de GitHub, en el siguiente enlace: [Configuración e integración de IFTTT](#). También es posible acceder mediante el enlace que aparece en la sección Anexo con el mismo nombre.

IFTTT (abreviación de If This Then That) es un servicio web que permite crear automatizaciones o tareas, con integración a una gran cantidad de dispositivos y servicios, la cual ha sido una de las opciones más relevantes a la hora de crear automatizaciones. Las posibilidades y el soporte de este servicio han sido las razones por las cuales se ha decidido usar para este trabajo. El servicio no es completamente gratuito, pero permite su uso básico con tres Applets sencillos, con un solo desencadenante y una sola acción por Applet.

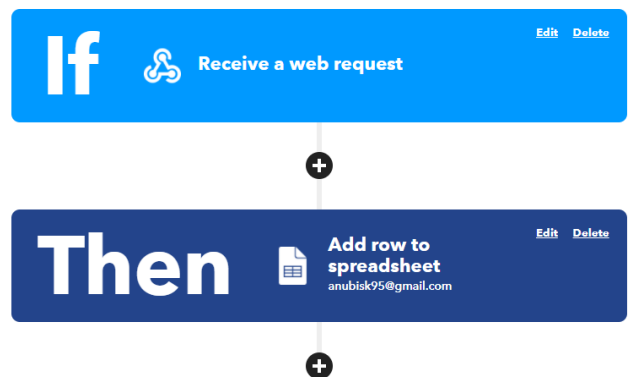


Imagen 16: Applet de IFTTT.

Con IFTTT es posible implementar diferentes automatizaciones, aumentando las posibilidades que permiten los propios sistemas domóticos, y en el caso de este trabajo, se ha aumentado la potencia de la propia automatización del sistema. Mediante dicho servicio, se ha podido añadir una capa más a la automatización creada, permitiendo que, una vez se detecte movimiento en el sensor o se deje de detectar movimiento, no solo el Nano Switch encienda o apague la lampara respectivamente, sino que permite añadir un registro a una hoja de cálculo de Google Sheets con el horario y el evento realizado. De la misma forma, se ha añadido esta funcionalidad al Window/Door sensor.

323	November 30, 2021 at 08:56AM	evento_movimiento	movimiento no detectado
324	November 30, 2021 at 08:58AM	evento_movimiento	movimiento detectado
325	November 30, 2021 at 09:07AM	evento_movimiento	movimiento no detectado
326	November 30, 2021 at 09:36AM	evento_movimiento	movimiento detectado
327	November 30, 2021 at 09:51AM	evento_puerta	puerta cerrada
328	November 30, 2021 at 09:51AM	evento_puerta	puerta abierta
329	November 30, 2021 at 09:52AM	evento_puerta	puerta cerrada
330	November 30, 2021 at 09:52AM	evento_puerta	puerta abierta
331	November 30, 2021 at 10:00AM	evento_movimiento	movimiento no detectado
332	November 30, 2021 at 10:01AM	evento_movimiento	movimiento detectado
333	November 30, 2021 at 10:13AM	evento_movimiento	movimiento no detectado

Imagen 17: Registro de eventos.

4.4.3 DuckDNS

Debido a que nuestro sistema se encuentra conectado a la red local mediante un cable ethernet, a priori no es posible acceder al dashboard mediante una red externa, a no ser que se contrate el servicio de pago que ofrece HomeAssistant. Debido a esto, se han buscado soluciones gratuitas y open-source para poder acceder mediante, por ejemplo, nuestro dispositivo a través de la red móvil.

Para dar solución, se han estudiado diferentes opciones de redirección de DNS, lo que nos permite convertir nuestra IP por la cual accedemos al dashboard en una dirección la cual se podrá usar desde fuera de la red interna. Este DNS redirecciona a la IP que nosotros indicamos (IP y puerto en este caso).

La opción escogida ha sido **DuckDNS**, debido a que tiene un gran soporte por parte de los desarrolladores, una integración sencilla a través de un plug-in en el sistema y además es una herramienta completamente gratuita.

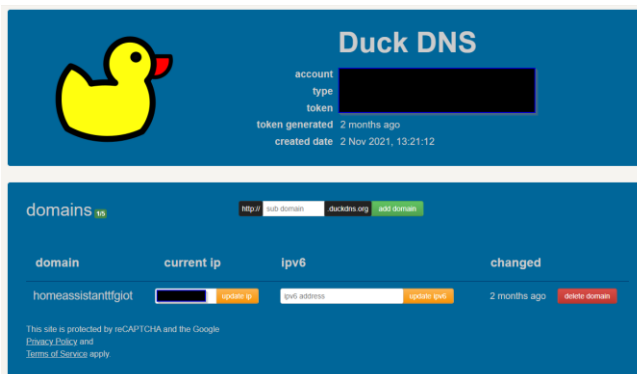


Imagen 18: Portal DuckDNS.

Con esta herramienta, hace posible acceder a nuestro sistema desde el exterior de la red y desde cualquier navegador, utilizando la URL que hemos generado o utilizando la aplicación de Home Assistant oficial, y especificando la URL generada.

La documentación de este servicio ha sido realizada en Github, y se puede acceder mediante el siguiente enlace: [Configuración e integración de DuckDNS](#). También es posible acceder mediante el enlace que aparece en la sección Anexo con el mismo nombre.

4.4.4 Portal Azure

Hasta este punto, nuestro sistema domótico es capaz de detectar eventos, tratarlos a través de automatizaciones simples ofrecidas por el sistema, o más complejas como las ofrecidas por IFTTT, y podemos acceder al sistema a través de cualquier red conectada a internet mediante el servicio DNS ofrecido por DuckDNS.

Los datos de los eventos se guardan temporalmente dentro de nuestro sistema, y a través de la automatización de IFTTT, guardamos los eventos en una hoja de cálculo de Google, pero se busca la integración con una herramienta más potente para el estudio de los datos obtenidos de los sensores.

La herramienta escogida ha sido Portal Azure de Microsoft, mediante el cual se pueden recibir los datos enviados desde el sistema Home Assistant. Es una herramienta muy potente, la cual tiene una integración muy fuerte con las demás herramientas de Microsoft, y debido a que se utilizará la herramienta de Low-Code de Microsoft, se ha decidido la utilización de esta.

Portal Azure ofrece una cierta cantidad de servicios gratuitos, pero los que se utilizan para el desarrollo de este trabajo no lo son. Ofrece un primer mes con 200\$ gratuitos a gastar en el servicio, y a partir de aquí es necesaria una suscripción al servicio para poder utilizar ciertas herramientas, como el Event Hub que se utiliza en nuestro caso.

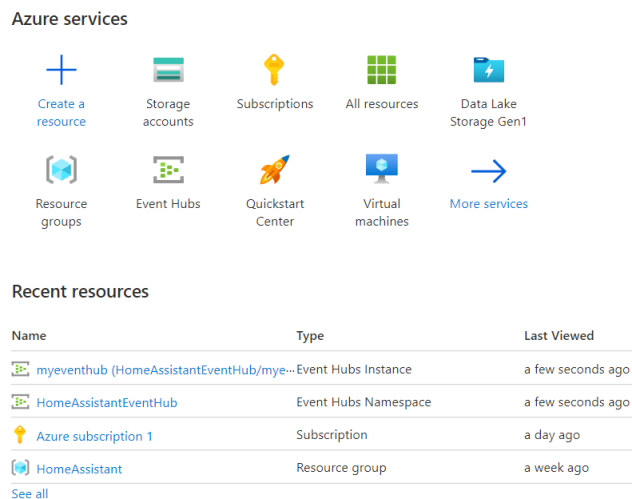


Imagen 19: Servicios y recursos de Portal Azure.

La documentación que se ha creado para la obtención de la clave de Portal Azure y la configuración de Home Assistant para el envío de datos ha sido realizada en Github, en el siguiente enlace: [Configuración e integración de Portal Azure](#). También es posible acceder mediante el enlace que existe en el anexo.

4.4.5 PowerApps

Para el desarrollo de la aplicación móvil se han estudiado diferentes alternativas. Debido a que en el grado se ha estudiado la programación de aplicaciones básica, se ha estudiado una opción diferente, y es el uso de las herramientas Low-Code que se encuentran en auge en la actualidad.

Una herramienta Low-Code permite el desarrollo de aplicaciones (multiplataforma en la mayoría de los casos) a través de interfaces gráficas, de una forma más sencilla y rápida.

Debido a esto, y a la integración que tiene con la herramienta del Portal Azure, se ha elegido la herramienta de Microsoft, Power Apps.

Debido a las complicaciones surgidas con el almacenamiento y el tratamiento de los datos en Portal Azure, se ha decidido crear una aplicación con los datos recogidos en la hoja de cálculo de Google a través de IFTTT, permitiendo así generar un ciclo completo desde la recogida de datos de los sensores hasta la aplicación, pese a que estos no pasen por Portal Azure sino por IFTTT.

5. APLICACIONES DEL TRABAJO

El sistema que se ha generado con este trabajo es sencillo, dos sensores (tres contando el sensor de GPS del móvil para la prueba de geolocalización de IFTTT) y un actuador, pero permite una infinidad de opciones distintas.

Un ejemplo sencillo actual, con los casos de Coronavirus que estamos viviendo en el mundo en el momento de la redacción de esta memoria, una implementación interesante podría ser la siguiente. Mediante un termómetro láser con conexión a internet que efectuaría como sensor, sería posible tomar la temperatura de los trabajadores a distancia, y como actuación, abrir la puerta a aquellos que no tengan una temperatura superior a 37°, guardando también un registro de ello. Todo esto posibilita un trabajador menos midiendo las temperaturas de los trabajadores.

En el ámbito doméstico, la domotización del hogar puede ser una gran ventaja, tanto en términos de comodidad como en términos de ahorro de energía debido al gran control de los aparatos electrónicos.

Un ejemplo sencillo podría ser el control de la temperatura de la calefacción conforme a los datos obtenidos por un sensor de temperatura exterior, o haciendo referencia al ejemplo realizado con IFTTT, encender la calefacción una vez entremos en un radio cercano a nuestra casa, o una vez abandonemos nuestro puesto de trabajo.

La gran posibilidad de aplicaciones posibles para un sistema domótico de este tipo es casi infinita teniendo en cuenta la cantidad de sensores, actuadores y tareas posibles.

6. CONCLUSIONES

Sin ningún tipo de dudas, creo que nos encontramos ante una de las tendencias actuales más poderosas. La domotización de entornos, tanto domésticos como industriales o de otros entornos están en auge. De una forma muy sencilla se puede observar cómo, sistemas como Alexa de Amazon, altavoces como Google Nest y otras marcas diferentes apuestan por este tipo de sensores/actuadores, que, a su vez, funcionan como nexo del sistema. Estoy seguro de que, hoy en día, todos hemos escuchado el "Alexa play Despacio", pero la capacidad de integración con el hogar que pueden generar estos sistemas es muchísimo mayor.

Como aprendizaje personal, he puesto en práctica la instalación y configuración de un entorno domótico sencillo pero con gran potencial, como también he puesto en práctica metodologías estudiadas en el grado, y conceptos que también han sido impartidos en diferentes materias, añadiendo también el uso de herramientas, tanto open-source como de empresas de nivel como las de Microsoft, y he utilizado herramientas que contienen paradigmas Low-Code, que sin duda creo que cada vez tomarán más y más importancia en un futuro no muy lejano, debido a que permiten codificar aplicaciones hasta a personas que nunca han programado una sola línea de código.

Es sencillo también pensar que, con una cantidad

grande de tutoriales, documentación y guías, es relativamente sencillo preparar un sistema domótico como este, pero nada más alejado de la realidad, siempre se encuentran pequeños problemas, los cuales hay que saber como interpretar y resolver.

7. BIBLIOGRAFÍA

A. (2021a, julio 6). *Acceso remoto a Home Assistant con DuckDNS y certificado SSL emitido por Let's Encrypt*. Fertry Tech. Recuperado 10 de noviembre de 2021, de <https://fertry.tech/index.php/2021/07/03/acceso-remoto-a-home-assistant-con-duckdns-y-certificado-ssl-emitido-por-lets-encrypt/>

(2021b, julio 6). *Acceso remoto a Home Assistant con DuckDNS y certificado SSL emitido por Let's Encrypt*. Fertry Tech. Recuperado 12 de noviembre de 2021, de <https://fertry.tech/index.php/2021/07/03/acceso-remoto-a-home-assistant-con-duckdns-y-certificado-ssl-emitido-por-lets-encrypt/>

Assistant, H. (s. f.-a). *Azure Event Hub*. Home Assistant. Recuperado 4 de noviembre de 2021, de https://www.home-assistant.io/integrations/azure_event_hub/

Assistant, H. (s. f.-b). *Setup basic information*. Home Assistant. Recuperado 2 de noviembre de 2021, de <https://www.home-assistant.io/docs/configuration/basic>

D. (s. f.-a). *GitHub - daniart1991/home-assistant-config: Home Assistant configuration running Hassio on a Raspberry Pi 2 B*. GitHub. Recuperado 10 de noviembre de 2021, de <https://github.com/daniart1991/home-assistant-config>

Duck DNS. (s. f.). DuckDNS. Recuperado 4 de noviembre de 2021, de <https://www.duckdns.org/domains>

González, D. M. (2020a, septiembre 4). *Controla tu casa desde cualquier sitio con DuckDNS*. daniart1991's Blog. Recuperado 10 de noviembre de 2021, de <https://www.danielmartingonzalez.com/es/controla-tu-casa-desde-cualquier-sitio-con-duckdns/>

González, D. M. (2020b, septiembre 4). *Controla tu casa desde cualquier sitio con DuckDNS*. daniart1991's Blog. Recuperado 10 de noviembre de 2021, de <https://www.danielmartingonzalez.com/es/controla-tu-casa-desde-cualquier-sitio-con-duckdns/>

IFTTT Maker Webhooks. (s. f.). IFTTT Maker. Recuperado 15 de noviembre de 2021, de <https://maker.ifttt.com/use/cEpiZ2tc4e7i8NkOh2FcKB>

Ifttt Webhook tutorial. (2021, 15 marzo). Reddit. Recuperado 25 de noviembre de 2021, de

https://www.reddit.com/r/homeassistant/comments/m5thg1/ifttt_webhook_tutorial/

openHAB. (s. f.). OpenHab. Recuperado 8 de octubre de 2021, de <https://www.openhab.org/>

P. (2021c, octubre 17). *Instalación de DuckDNS en Home Assistant. Acceso remoto desde internet*. - Paciencia Digital. Domótica. Paciencia Digital. Domótica, Estadística y Datos. Recuperado 4 de noviembre de 2021, de <https://www.pacienciadigital.com/instalacion-duckdns-home-assistant/>

P. (2021d, octubre 17). *Instalación Home Assistant + Raspberry Pi*. Paciencia Digital. Domótica, Estadística y Datos. Recuperado 7 de noviembre de 2021, de <https://www.pacienciadigital.com/instalacion-home-assistant-paso-a-paso/>

Progress Customer Community. (s. f.). Create IFTTT Action. Recuperado 2 de diciembre de 2021, de <https://community.progress.com/s/article/How-to-create-the-IFTTT-action>

R. (2021e, enero 14). *Duckdns config fails - not a file for dictionary value*. Home Assistant Community. Recuperado 13 de noviembre de 2021, de <https://community.home-assistant.io/t/duckdns-config-fails-not-a-file-for-dictionary-value/268225/4>

S. (s. f.-b). *Azure Quickstart - Create an event hub using the Azure portal - Azure Event Hubs*. Microsoft Docs. Recuperado 4 de diciembre de 2021, de <https://docs.microsoft.com/en-us/azure/event-hubs/event-hubs-create>

S. (2021f, diciembre 27). *Captura de eventos de streaming: Azure Event Hubs - Azure Event Hubs*. Microsoft Docs. Recuperado 10 de enero de 2022, de <https://docs.microsoft.com/es-es/azure/event-hubs/event-hubs-capture-overview>

S. (2021g, diciembre 27). *Creación de un centro de eventos con la captura habilitada: Azure Event Hubs - Azure Event Hubs*. Microsoft Docs. Recuperado 15 de enero de 2022, de <https://docs.microsoft.com/es-es/azure/event-hubs/event-hubs-resource-manager-namespace-event-hub-enable-capture>

T. (2021h, diciembre 15). *Configuración de la cadena de conexión - Azure Storage*. Microsoft Docs. Recuperado 15 de enero de 2022, de <https://docs.microsoft.com/es-es/azure/storage/common/storage-configure-connection-string>

T. (2022, 10 enero). *Inicio rápido: Carga, descarga y enumeración de blobs con Azure Portal - Azure Storage*. Microsoft Docs. Recuperado 10 de enero de 2022, de <https://docs.microsoft.com/es-es/azure/storage/blobs/storage-quickstart-blobs-portal#create-a-container>

8. ANEXO

La documentación de este trabajo ha sido realizada en la página web externa Github, como se indica en los apartados indicados. En ella también se encuentran todos los archivos de la tarjeta SD del sistema HomeAssistant y OpenHab para el libre uso. Los enlaces, claves de acceso a servicios y IP's han sido borrados de los archivos para evitar problemas de seguridad con mi red interna, también debido a que cada usuario deberá tener valores diferentes.

La página principal es la siguiente: <https://github.com/paridp23/TFGiot>

Las páginas de documentaciones creadas están dentro de la página principal, y son las siguientes.

Configuración e integración de IFTTT: <https://github.com/paridp23/TFGiot/blob/main/IFTTT.md>

Configuración e integración de DuckDNS: <https://github.com/paridp23/TFGiot/blob/main/DuckDNS.md>

Configuración de Portal Azure: <https://github.com/paridp23/TFGiot/blob/main/PortalAzure.md>

9. AGRADECIMIENTOS

En primer lugar, me gustaría agradecer este trabajo a una gran parte de los profesores que he tenido, los cuales me han contagiado su pasión por aprender, por la programación y por la tecnología.

A mis padres, por todo lo que me han enseñado, transmitido y por todo el amor que me han dado.

A Marina, por su incondicional apoyo en todo momento.

Y a mi abuelo, que este donde este, sé que está orgulloso de mi.