UNIVERSIDAD POLITÉCNICA DE CARTAGENA

Escuela Técnica Superior de Ingeniería de Telecomunicación

SISTEMAS DE ALARMAS PARA LOCALIZACIÓN DE INTERIORES

TRABAJO FIN DE GRADO

GRADO EN INGENIERÍA TELEMÁTICA

Autor: Francisco José Martínez Vidal

Director: José Fernando Cerdán Cartagena

Codirector: Diego García Sánchez

Cartagena, diciembre 2020

Autor	Francisco José Martínez Vidal
E-mail del autor	franmartinezvidal@gmail.com
Director	José Fernando Cerdán Cartagena
E-mail del director	fernando.cerdan@upct.es
Codirector	Diego García Sánchez
E-mail del codirector	diego.garcia@ingeniatic.com

Resumen

En este TFE estamos trabajando en un sistema de alarmas para localización de interiores, estamos usando un gateway, 5 coins y 3 assets. Hemos realizado diversas comprobaciones para ver el correcto funcionamiento de los dispositivos. El objetivo ha sido la comprobación de la precisión con la que los assets iban moviéndose de un coin a otro, es decir, el tiempo que tardan en actualizar su nueva posición, así como la distancia a la que perdían la cobertura dos coins. También hemos podido observar mediante el uso del programa Wireshark y de nFR sniffer las tramas BLE que se producen entre los distintos dispositivos así como sus distintas fases de conexión y los distintos valores que forman cada trama.

Titulación	Grado en Ingenería Telemática
Departamento	Tecnologías de la Información y las Comunicaciones
Fecha de presentación	Diciembre 2020

Agradecimientos

A los directores del proyecto, José Fernando y Diego, por su implicación y apoyo en el proyecto y por las soluciones tomadas para los problemas que iban saliendo, a pesar del confinamiento y la situación sanitaria que estamos viviendo.

A mi familia por haberme apoyado durante todos los años de la carrera y durante la realización del proyecto.

ÍNDICE

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

1. INTRODUCCIÓN

1.1 Introducción

Cuando nuestros mayores viven solos, es normal entre sus familiares preocuparse por ellos. Con cierta frecuencia, nos preocupamos por no saber si están bien nuestros padres o abuelos en un determinado momento. Es normal querer llamarles con frecuencia para asegurarnos que se encuentran bien y sin ningún tipo de problema pero las llamadas no siempre son la mejor opción. Es por ello que la tecnología es una gran alternativa para saber cómo se encuentran nuestros mayores.

Se han producido muchos casos de personas mayores que caen y yacen en el suelo sin poder moverse esperando a que alguien les auxilie, provocando en muchas ocasiones esperas de horas hasta que los familiares se dan cuenta, esto es típico en situaciones como los ictus, problemas de corazón o simplemente un tropiezo que produce una caída en la que puede romperse la cadera o alguna otra parte del cuerpo. Otro caso es el de las veces en las que llamamos a nuestro familiar y no responde al teléfono, nos asustamos y nos imaginamos el peor de los casos y, al final, simplemente no escuchó el teléfono, [1].

Para saber si nuestros mayores se encuentran bien se han usado a lo largo de los años distintos tipos de tecnologías para poder vigilar o controlar sus movimientos y cuando se observe algún comportamiento extraño, poder darte cuenta a tiempo. Un ejemplo de estas tecnologías son las cámaras de videovigilancia, para poder vigilar a nuestro familiar y así poder ahorrarnos más de un susto. Otro tipo de tecnología es una pulsera con botón de alarma con el que al pulsarlo proporciona una conexión con los profesionales en caso de necesidad facilitando una respuesta o intervención inmediata.

Tanto las tecnologías nombradas anteriormente como otras ayudan a poder realizar un seguimiento de las personas mayores a través de un móvil o de un ordenador, por lo que las ventajas que aportan son bastante considerables ya que prácticamente en caso de necesidad pueden ser auxiliados inmediatamente.

Tal y como indica el resumen, en el proyecto vamos a centrarnos en un sistema de alarmas para localización de interiores, es decir, nos centraremos en un sistema formado por 5 coins, 3 assets y un gateway conectados entre sí a través de bluetooth, y que se envían tramas BLE dependiendo de la posición de donde se encuentre el asset. Esto está orientado para, a través de la aplicación web, poder saber en qué zona de la casa se encuentra nuestro familiar y si se cambia de habitación o está durante un periodo de tiempo sin moverse, nos envía una alerta al correo. Esto es importante para conocer el estado en que se encuentra, ya que si la persona se mueve con normalidad significa que no ha tenido ningún problema pero si se encuentra durante un tiempo prolongado inactivo, puedes realizar una llamada o ir a comprobar para saber si se encuentra bien y que en caso contrario, se pueda pedir ayuda en el menor tiempo posible.

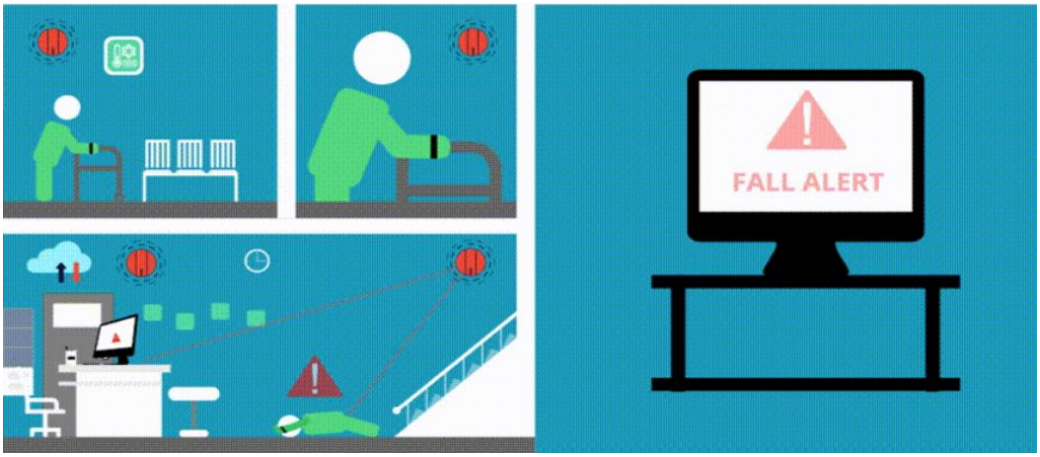

Figura 1: Ejemplo de funcionamiento de nuestro sistema.

Las opciones que nos permite nuestro sistema son las siguientes:

- La geofence: Nos permite establecer límites. Puede ser un círculo alrededor de un punto específico o dos polígonos que representan lados.
- Activar una alerta: Para supervisar fácilmente la ubicación de una persona. cuando una persona sale o entra en el área asignada, se dispara una alarma.
- Ver actividad en el panel: El mapa del piso se puede importar en nuestro tablero y se puede planificar la posición de las Coins.
- Ubicación en tiempo real: Para obtener la ubicación exacta de una persona dentro del área definida. Además, determina la cantidad de usuarios en un área predefinida.
- Industrias: Nuestras soluciones de geofence y seguimiento de personas se pueden utilizar para industrias como hospitales, industrias farmacéutica y química, transporte y muchas industrias de alto riesgo que tienen una necesidad imperiosa de monitorear la ubicación precisa y la seguridad de varios empleados a la vez.
- Fácil integración : Se puede calcular la cobertura de cada Coin y crear zonas de geofences virtuales, [2].

1.2 Objetivos

El principal objetivo de este TFG es la evaluación y el estudio del funcionamiento de un sistema completo de posicionamiento en interiores basado en la tecnología inteligente de BLE. Para ello se ha estudiado el funcionamiento del protocolo BLE y de la API del servidor de localización proporcionado.

También se proporcionará una estimación de la fiabilidad del sistema cuando se cambia de localización dentro del interior.

1.3 Fases del proyecto

Para llevar a cabo esta tarea son necesarias las siguientes fases:

1. Estudiar y entender el protocolo BLE
2. Estudiar y entender la API del servidor de localización BLE.
3. Ajustar el mapa al lugar donde se realizarán las pruebas y colocación de todos los dispositivos usados.
4. Pruebas de cambio de posición de los distintos Assets para ver si el sistema funciona de forma correcta.
5. Análisis de resultados.

1.4 Estructura del proyecto

La memoria del proyecto se divide en los siguientes apartados:

- En el capítulo 1, se realiza una introducción, además de indicar los objetivos y las fases del proyecto.
- En el segundo capítulo, se lleva a cabo el estudio del protocolo BLE, su pila de protocolos, sus ventajas y desventajas y las tramas BLE.
- En el tercer capítulo, se indican las tecnologías y los equipamientos que se han utilizado a lo largo del desarrollo del proyecto.
- Seguidamente, en el capítulo 4, se indican las configuraciones de los distintos programas utilizados en el proyecto.
- En el capítulo 5 se encuentran las pruebas hechas y los resultados obtenidos en ellas.
- En el capítulo 6 encontramos los problemas que surgieron durante el desarrollo del proyecto y su resolución.
- En el capítulo 7 podemos encontrar las conclusiones del proyecto.
- Para terminar, en el capítulo 8 aparecen las bibliografías y referencias.

2. PROTOCOLO BLE

2.1 BLE, [3]

El Bluetooth de baja energía (Bluetooth Low Energy o BLE), es un subconjunto del estándar Bluetooth v4.0. Dispone de una pila de protocolos en referencia a la capa OSI completamente nueva y orientada a conexiones sencillas en aplicaciones de muy baja potencia (dispositivos dependientes de batería o pila).

Dentro de las oportunidades que ofrece esta tecnología, una de sus mayores ventajas reside en el hecho de la aceptación obtenida por parte de las grandes plataformas a día de hoy como IOs, Android, Microsoft o Linux entre otras y su compatibilidad con éstas. Cabe destacar del mismo modo alguna otra de sus características fundamentales, como su interoperabilidad en el mundo de los fabricantes de chipsets, tamaño reducido, requerimientos de potencia muy bajos y un aceptable rango de alcance en las comunicaciones.

2.2 Pila de protocolos BLE, [3]

La pila de protocolos para Bluetooth Low Energy sigue la estructura definida a continuación, [3]:

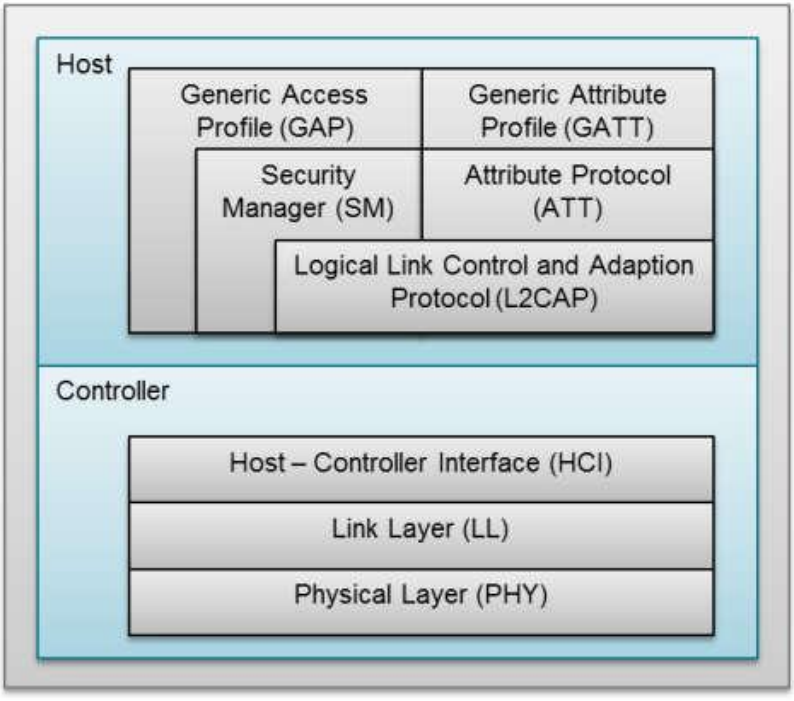

Figura 2: Pila de protocolos BLE

La capa física contiene la circuitería de comunicaciones capaz de realizar los procesos de modulación y demodulación de señales analógicas y posteriormente transformarlas en símbolos digitales. La tecnología BLE es capaz de utilizar hasta 40 canales de 2MHz en la banda ISM de 2.4 GHz. El estándar emplea la técnica “frequency hopping” o “saltos en frecuencia”, siguiendo una secuencia de saltos pseudo-aleatorios entre los canales frecuenciales mencionados que ofrece un alto grado de robustez frente a interferencias.

La capa de enlace (link layer), se encarga de gestionar características como los requerimientos temporales del estándar, chequeo de mensajes y reenvío de mensajes erróneos recibidos, gestión, filtrado de direcciones etc. Además ofrece la definición de roles (Advertiser, Scanner, Master and Slave) que permiten identificar de forma lógica el rol de cada dispositivo en el proceso de comunicación. El nivel LL es del mismo modo responsable de procesos de control como el cambio de parámetros de la conexión o la encriptación.

HCI es un protocolo estándar que permite que la comunicación entre un host y un controlador se lleve a cabo a través de un interfaz serie. A modo de ejemplo, en la mayoría de smartphones u ordenadores el host y la aplicación corren en la CPU principal mientras que el controlador está situado en hardware específico y separado, conectado mediante UART o USB. El estándar Bluetooth define HCI como el conjunto de comandos y eventos para la interacción de ambas partes (host y controlador).

La capa L2CAP (Logic Link Control and Adaptation Protocol), se responsabiliza de dos tareas fundamentales en un proceso de comunicación. En primer lugar, el proceso de multiplexación, es decir, la capacidad de dar formato a mensajes provenientes de las capas OSI superiores y encapsularlos en paquetes estándar BLE así como el proceso inverso.

Por otro lado, la fragmentación y recombinación. Paquetes que en nivel de aplicación suponen datagramas de gran cantidad de bytes son fragmentados correctamente adecuándose al MTU de BLE (27 bytes de payload máximo).

Para BLE, la capa L2CAP es la encargada de dar acceso y soporte a los dos protocolos fundamentales. Por un lado, ATT (Attribute Protocol), un protocolo basado en atributos presentados por dispositivo, con arquitectura cliente-servidor, que permite el intercambio de información. Por otro lado, SMP (Security Manager Protocol), protocolo que proporciona un framework para generar y distribuir claves de seguridad entre dos dispositivos.

En el nivel más alto de la capa de protocolos, encontraremos de forma paralela las capas GAP y GATT. Esta primera, GAP (Generic Acces Profile), permite que un dispositivo sea visible hacia el resto de dispositivos y además determina como puede interactuar un dispositivo entre otro. Establece distintas normas y conceptos para estandarizar las operaciones de más bajo nivel como:

- Roles de interacción
- Modos de operación y transición entre ellos
- Procedimientos para establecimiento de comunicación
- Modos de seguridad y procedimientos

Al otro lado, GATT (Generic Attribute Profile), que define como dos dispositivos BLE transfieren información. Este proceso tiene lugar cuando dos dispositivos han superado la fase de establecimiento de comunicación (controlada por GAP) y comienza la transferencia de información pudiendo ser de forma bidireccional.

A continuación se recogen alguna de las aplicaciones para las cuales fue concebida esta tecnología y en las cuales las empresas comienzan a apostar por su puesta en marcha, [4]:

- Seguridad y sensores de proximidad
- Dispositivos para el hogar
- Dispositivos para salud y bienestar
- Sector automovilístico
- Contadores Inteligentes en el sector energético

2.3 Ventajas y desventajas BLE, [5]

2.3.1 Ventajas

- Ofrece un consumo de energía muy bajo y, por lo tanto, la vida útil de la batería puede ser muy larga. Esto se debe al hecho de que tanto los dispositivos maestros como los esclavos pueden pasar al modo de suspensión profunda entre transacciones. El maestro informa al esclavo sobre la secuencia de salto y cuándo despertarse.
- Se puede utilizar para transferencias de datos de tamaño pequeño, especialmente en aplicaciones basadas en IoT (Internet de las cosas). Ofrece una capacidad máxima de mensajes de hasta 255 bytes.
- Ofrece seguridad mediante algoritmos AES de 128 bits. Por lo tanto, el cifrado se aplica a los datos que se comunican entre los dispositivos maestro y esclavo.
- Utiliza saltos de frecuencia utilizando una banda de 2,4 GHz, por lo que puede soportar interferencias. Además, los canales publicitarios se eligen para que sean diferentes a las frecuencias de los canales wifi para evitar interferencias entre los dispositivos wifi y bluetooth.
- Los dispositivos BLE son robustos para operar en entornos congestionados debido a la introducción de V5.0
- Ofrece confiabilidad y permite la vida digital.
- Los dispositivos de baliza =>BLE se han vuelto populares debido al aumento de velocidad y alcance en la versión bluetooth 5.0.
- El procedimiento de establecimiento de la conexión y la transferencia de datos es muy rápido (aproximadamente 3 ms).
- Los dispositivos de diferentes fabricantes son compatibles con los demás.
- Hay un sinfín de dispositivos que utilizan BLE y también son económicos.

2.3.2 Desventajas

- No se puede usar para velocidades de datos más altas que las que ofrecen las tecnologías wifi y celular. Admite velocidades de datos de 1 Mbps y 2 Mbps.
- No se puede utilizar para comunicaciones inalámbricas de larga distancia a diferencia de los dispositivos móviles y wifi. Soporta hasta 200 metros en LOS (Line of Sight).
- Está abierto a intercepciones y ataques debido a la transmisión / recepción inalámbrica.

2.4 Tramas BLE y Wireshark, [6]:

Las fases serían las siguientes:

ADV_IND
Un dispositivo periférico solicita conexión a cualquier dispositivo central (es decir, no dirigido a un dispositivo central en particular).
Tanto las coins, el gateway y los assets tienen de este tipo de fase.

CONNECT_REQ
Emitido por el dispositivo maestro. Esta PDU es enviada por la capa de enlace en el estado de inicio y recibida por la capa de enlace en el Advertising State. Una vez que el dispositivo esclavo recibió esta PDU, la conexión se establecerá inmediatamente e ingresará al estado de conexión (el dispositivo maestro y el dispositivo esclavo intercambiarán la PDU efectiva o la PDU vacía).

SCAN_REQ
Solicitud de escaneo, emitida por dispositivo maestro, enviada al dispositivo esclavo, enviada por la Capa de Enlace en el Estado de Escaneo, recibida por una Capa de Enlace en el Advertising State.

SCAN_RSP
Enviado por la capa de enlace en el Advertising State, recibido por una capa de enlace en el estado de escaneo, el dispositivo esclavo podría transferir más datos publicitarios al dispositivo maestro.

Para analizar las tramas BLE se ha tenido que utilizar Wireshark, si pulsamos en una de las tramas salen tres apartados:

Frame: Te salen los datos acerca de la trama elegida, como la encapsulación, número de trama, longitud, protocolo…

Nordic BLE Sniffer: Es el tipo de encapsulación de la trama. Te da datos como la dirección en la que va la trama, si está encriptada, el canal....

Bluetooth Low Energy Link Layer: Te salen datos como el tipo de PDU o el Advertising Address, que sería lo que identifica a las coins, los assets o el Gateway.

3. EQUIPAMIENTOS Y TECNOLOGÍAS USADAS.

3.1 Coins

Forman una red peer to peer de pequeños nodo de sensores de bajo coste, esta red recopila datos de estado y ubicación en interiores y envía esta información a la nube, [7].

Figura 3: Ejemplo de Coins utilizados.

3.2 Gateway

Es asignada a estas Coins, recopila datos y transfiere esta información a la nube a través de Wi-Fi, [8].

Figura 4: Ejemplo de gateway utilizado.

3.3 Assets

Cada Asset tiene una identificación única. Incluso si hay varios usuarios en las instalaciones generales, cada persona puede ser identificada con esta identificación única. Cuando la persona que lleva un asset cambia de habitación o lugar se conecta con el Coin más cercano para poder saber la nueva localización, [2].

3.4 nRF52 DK, [9]

Esta placa será la que conectaremos al ordenador para visualizar las tramas BLE que intercambian los distintos dispositivos, para ello debemos instalar Wireshark que será donde se visualicen las tramas y nRF connect que será donde configuraremos la placa para que den los resultados adecuados.

El nRF52 DK es un kit de desarrollo de placa única versátil para Bluetooth Low Energy, Bluetooth mesh, NFC, ANT y desarrollo propietario de 2,4 GHz en los SoC nRF52805, nRF52810 y nRF52832.

Incluye una antena NFC que permite rápidamente la utilización del periférico de etiqueta NFC-A en el nRF52832. Todos los GPIO están disponibles a través de conectores de borde y encabezados, y 4 botones y 4 Los LED simplifican la salida y la entrada desde y hacia el SoC.

Es compatible con Arduino Uno Revision 3, lo que hace posible montar escudos de terceros con facilidad. Viene con un depurador SEGGER J-Link integrado que permite programar y depurar tanto el SoC integrado como los SoC externos a través del encabezado de depuración. También interactúa directamente con el kit Power Profiler.

Puede ser alimentado por USB, pero también incluye un soporte de batería CR2032, lo que permite probar prototipos en el campo.

Se admiten los IDE y las cadenas de herramientas de SEGGER Embedded Studio, Keil, GCC e IAR; consulte IDE y cadenas de herramientas para obtener más información.

La caja incluye una placa nRF52 DK, una antena NFC y una batería CR2032. El diseño del hardware y los esquemas están disponibles

Figura 5: Placa nRF52 DK utilizada.

3.5 Postman

En nuestro proyecto se ha usado para realizar las consultas sobre la API que utilizamos, de esta forma podemos hacer distintas comprobaciones de cambios de posición de los asset, es decir, los geofences activados en la página web, y nos muestran distintos tipos de datos de esa consulta.
Un ejemplo sería la figura siguiente:

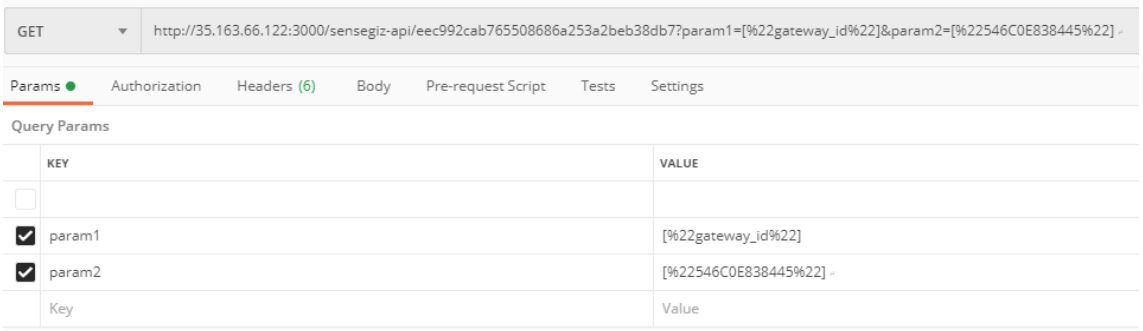

Figura 6: Ejemplo de consulta en Postman.

3.5.1 Características de Postman, [10]

Es una herramienta que nos permite crear peticiones sobre APIs de forma que podamos probarlas.

- **Crear Peticiones**, te permite crear y enviar peticiones http a servicios REST mediante un interface gráfico. Estas peticiones pueden ser guardadas y reproducidas a posteriori.
- **Definir Colecciones**, mediante Postman podemos agrupar las APIs en colecciones. En estas colecciones podemos definir el modelo de autentificación de las APIs para que se añada en cada petición. De igual manera podemos ejecutar un conjunto de test, así como definir variables para la colección.
- **Gestionar la Documentación**, genera documentación basada en las API y colecciones que hemos creado en la herramienta. Además esta documentación podemos hacerla pública.
- **Entorno Colaborativo**, permite compartir las API para un equipo entre varias personas. Para ello se apoya en una herramienta de colaborativa en Cloud.
- **Genera código de invocación**, dado un API es capaz de generar el código de invocación para diferentes lenguajes de programación: C, cURL, C#, Go, Java, JavaScript, NodeJS, Objective-C, PHP, Python, Ruby, Shell, Swift,…
- **Establecer variables**, con Postman podemos crear variables locales y globales que posteriormente utilicemos dentro de nuestras invocaciones o pruebas.
- **Soporta Ciclo Vida API management**, desde Postman podemos gestionar el ciclo de vida del API Management, desde la conceptualización del API, la definición del API, el desarrollo del API y la monitorización y mantenimiento del API.
- **Crear mockups**, mediante Postman podemos crear un servidor de mockups o sandbox para que se puedan testear nuestras API antes de que estas estén desarrolladas.

3.6 WIRESHARK

Wireshark, es un analizador de protocolos utilizado para realizar análisis y solucionar problemas en redes de comunicaciones, para análisis de datos y protocolos, y como una herramienta didáctica. Cuenta con todas las características estándar de un analizador de protocolos de forma únicamente hueca.

La funcionalidad que provee es similar a la de tcpdump, pero añade una interfaz gráfica y muchas opciones de organización y filtrado de información. Así, permite ver todo el tráfico que pasa a través de una red (usualmente una red Ethernet, aunque es compatible con algunas otras) estableciendo la configuración en modo promiscuo.

Permite examinar datos o de un archivo de captura salvado en disco. Se puede analizar la información capturada, a través de los detalles y sumarios por cada paquete. Wireshark incluye un completo lenguaje para filtrar lo que queremos ver y la habilidad de mostrar el flujo reconstruido de una sesión de TCP.

Wireshark es software libre, y se ejecuta sobre la mayoría de sistemas operativos Unix y compatibles, incluyendo Linux, Solaris, FreeBSD, NetBSD, OpenBSD, Android, y macOS, así como en Microsoft Windows, [11].

Para este proyecto se ha hecho uso de Wireshark para poder observar las tramas BLE que generaban los dispositivos utilizados al activarlos o cambiar de posición.

3.7 nRF Connect, [12]

nRF Connect Programmer es una aplicación disponible en nRF Connect for Desktop que puede utilizar para programar firmware en dispositivos nórdicos. La aplicación le permite ver el diseño de la memoria de los dispositivos USB J-Link y Nordic. También le permite mostrar el contenido de los archivos HEX y escribirlo en los dispositivos.

Dispositivos soportados:

- Nordic Thingy:91
- nRF91 Series DKs
- nRF53 Series DKs
- nRF52 Series DKs and Dongle
- nRF51 Series DKs and Dongle

3.7.1 Highlights BLE

- Aplicación multiplataforma fácil de usar para pruebas de conectividad Bluetooth LE.
- Admite la detección automática del kit nórdico conectado y la carga automática de FW.
- Admite LE Security introducido en Bluetooth 4.2.
- Hasta 8 conexiones Bluetooth LE simultáneas.
- Máximo 8 conexiones centrales simultáneas.
- Max 1 conexión periférica.
- Busca dispositivos Bluetooth LE.
- Analiza datos publicitarios.
- Muestra el valor RSSI.
- Se conecta a cualquier dispositivo Bluetooth LE conectable.
- Descubre y analiza servicios y características.
- Asistente de introducción para instalar las herramientas de desarrollo y nRF Connect SDK para SoC nRF5340 y SoC de la serie nRF52.
- Administrador de cadena de herramientas de un solo paso para instalar herramientas de desarrollo y nRF Connect SDK para SoC nRF5340 y SoC de la serie nRF52.

4. CONFIGURACIONES

4.1 nrF Sniffer BLE, [13]

El software nRF Sniffer para Bluetooth LE consta de firmware que se programa en una placa de desarrollo o un complemento de captura para Wireshark que registra y analiza los datos detectados.
Antes de comenzar a configurar nRF Sniffer, hay que tener los siguientes requisitos instalados:
• Wireshark v2.4.6 o posterior (se recomienda v3.0.7 o posterior en Windows).
• Python v3.6 o posterior.

Hay que descargar nRF Sniffer para Bluetooth LE v3.xo posterior y extraer el archivo en una carpeta a nuestra elección. Esta carpeta se llama Sniffer_Software.
Luego hay que programar el firmware en la placa, instalar la herramienta de captura nRF Sniffer y agregar un perfil de Wireshark para el Sniffer.

4.1.1 Progamación del nRF Sniffer firmware, [13]

Hay que conectar una placa de desarrollo que ejecute el firmware nRF Sniffer al ordenador para poder utilizar el nRF Sniffer para Bluetooth LE.

Hay varias formas de programar el firmware nRF Sniffer. La utilizada utiliza nRF Connect Programmer.

Para programar la placa , hay que seguir los siguientes pasos:

1) Instalar nRF Connect Programmer.
2) En macOS y Linux, hay que instalar el software SEGGER J-Link.
   Como he usado Windows este paso he podido saltarlo ya que está incluido en nRF Connect for Desktop.
3) Hay que localizar el archivo de firmware HEX de su placa.
   Todos los archivos de firmware HEX se encuentran en Sniffer_Software / hex /.
4) Para programar el archivo HEX es necesario programar un kit de desarrollo o el Dongle nRF51.

4.1.2 nRF Connect Programmer, [14]

La aplicación Programmer se instala como una aplicación para nRF Connect for Desktop.

Antes de instalar la aplicación Programmer, se descarga e instala nRF Connect for Desktop (versión 3.2.0 o posterior).

Para instalar la aplicación Programmer:

- Hay que abrir nRF Connect para escritorio.
- Buscar la aplicación Programmer en la lista de aplicaciones y haga clic en Instalar.

Cuando la aplicación está instalada, se inicia pulsando en Open.

Si hay una nueva versión de la aplicación, se muestra un botón Update junto al botón Open.
Se hace clic en este botón para instalar la última versión. Para desinstalar la aplicación, se hace clic en el botón de flecha hacia abajo y se selecciona Uninstall.

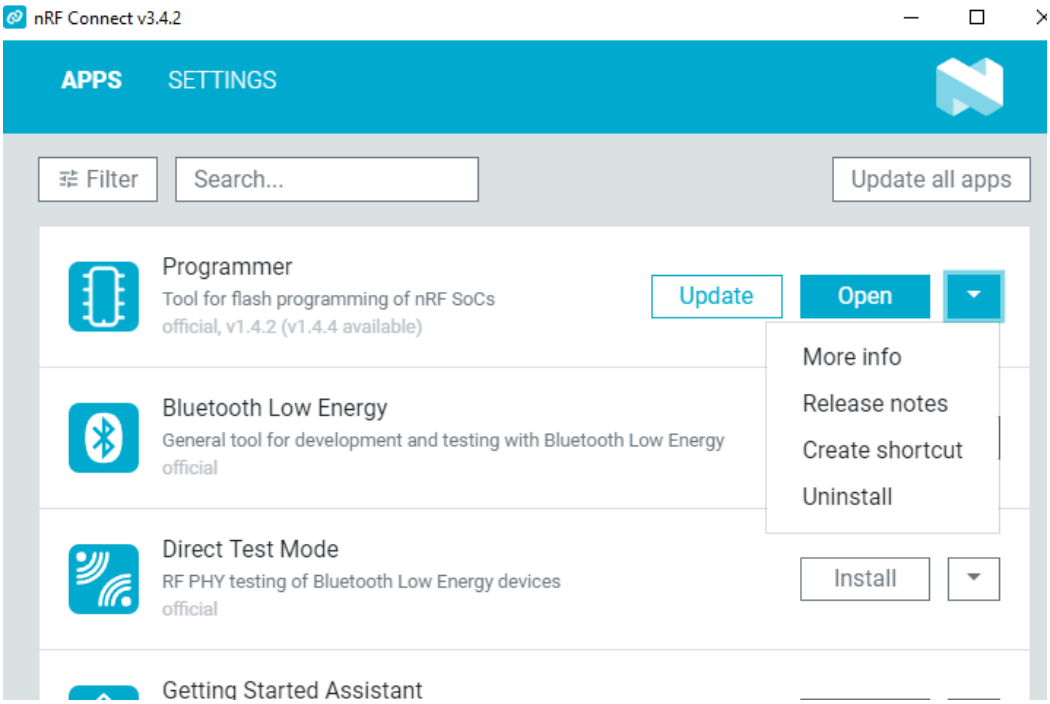

Figura 7: Menú nRF Connect.

4.1.3 Programming a Development Kit or the nRF51 Dongle, [14]

1. Se abre nRF Connect para escritorio y se inicia nRF Connect Programmer.
2. Conectamos un kit de desarrollo al ordenador con un cable USB.

   En la barra de navegación, No hay dispositivos disponibles cambiamos a Seleccionar dispositivo.

3. Se hace clic en Seleccionar dispositivo y seleccionamos el dispositivo de la lista desplegable.

   El texto del botón cambia al SEGGER ID del dispositivo seleccionado y la sección Device Memory Layout indica que el dispositivo está conectado.

4. Para ver visualmente el diseño de la memoria antes de programar, hago clic en Read en el menú.
5. Hago clic en Add HEX file en el panel File para agregar los archivos que quiero programar, usando una de las siguientes opciones:
   - Seleccionando archivos recientemente usados.
   - Seleccionando los archivos navegando hasta su destino de archivo.
6. Si no hay archivos usados recientemente, se hace clic en Browse en la lista desplegable que aparece como resultado del paso anterior.
7. Selecciono el archivo de imagen del firmware (con la extensión .hex) en el explorador de archivos que se abre.
8. Hago clic en Erase & Write en el panel Device para programar el dispositivo.

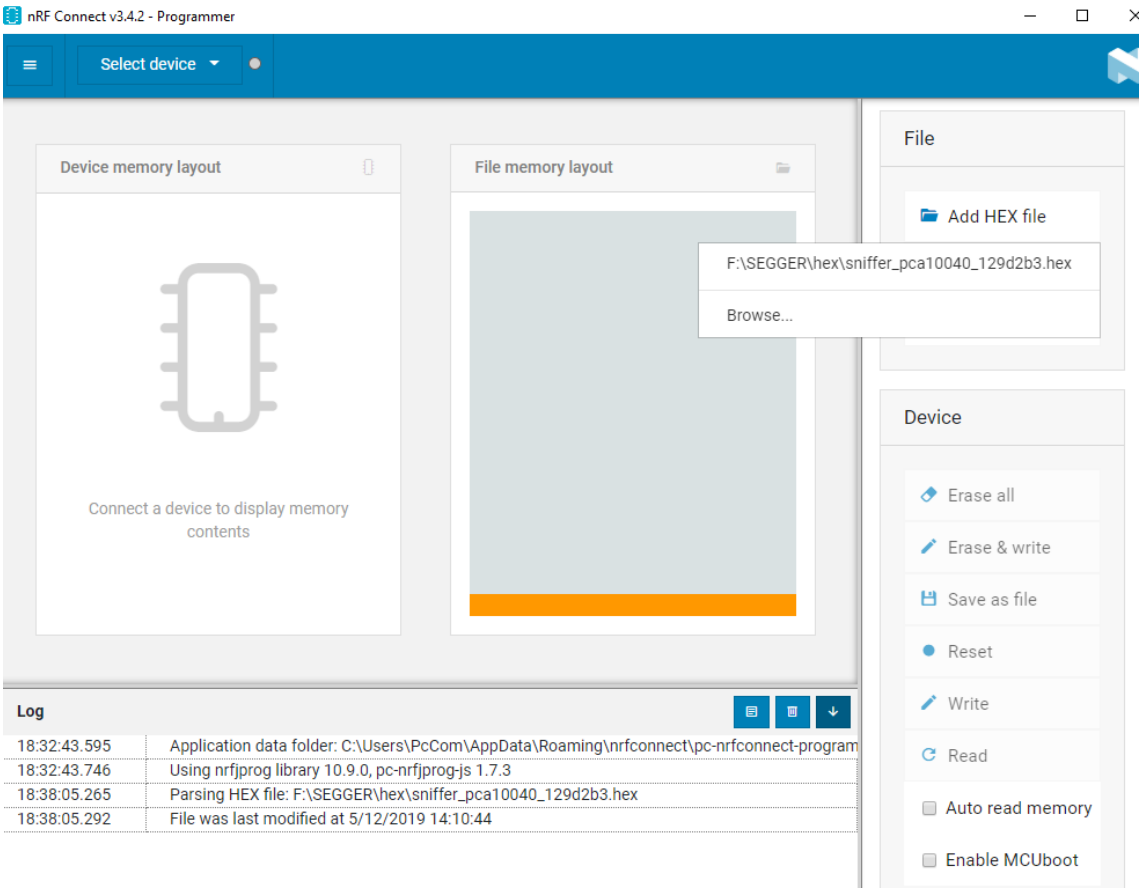

Figura 8: Añadir HEX file en nRF Connect.

4.1.4 Instalación de la herramienta de captura nRF Sniffer, [13]

El software nRF Sniffer para Bluetooth LE se instala como un complemento de captura externo en Wireshark.

Para instalar la herramienta de captura nRF Sniffer, hay que seguir los siguientes pasos:

1. Instalar los requisitos de Python:

a) Abro una ventana de comandos en la carpeta Sniffer_Software / extcap /.
b) Escribo pip3 install -r requirements.txt para instalar los requisitos.
c) Cierro la ventana de comandos.

2. Copio la herramienta de captura Sniffer en la carpeta de Wireshark para los complementos de captura externos:

a) Abro Wireshark.
b) Go to Help > About Wireshark

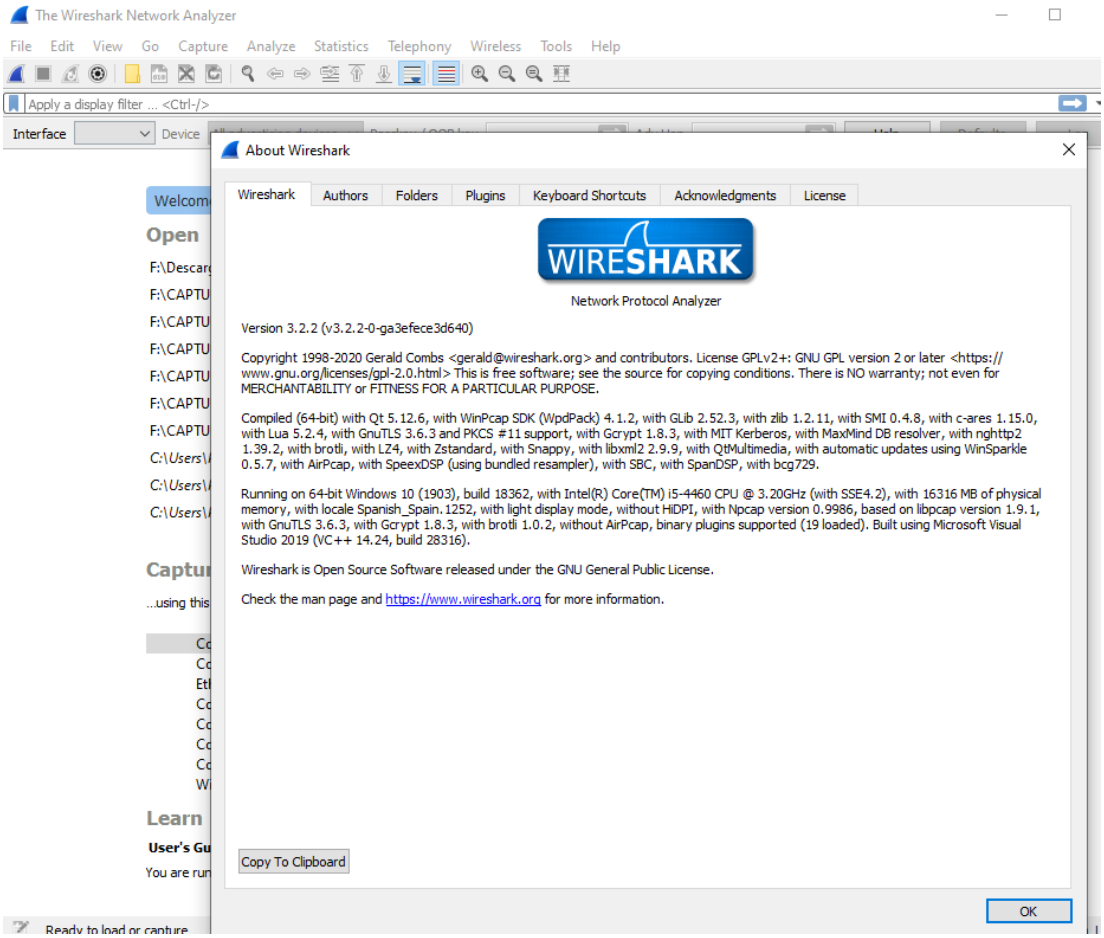

Figura 9: About Wireshark.

c) Selecciono la pestaña Folders.
d) Hago doble clic en la ubicación de la ruta Extcap para abrir esta carpeta.

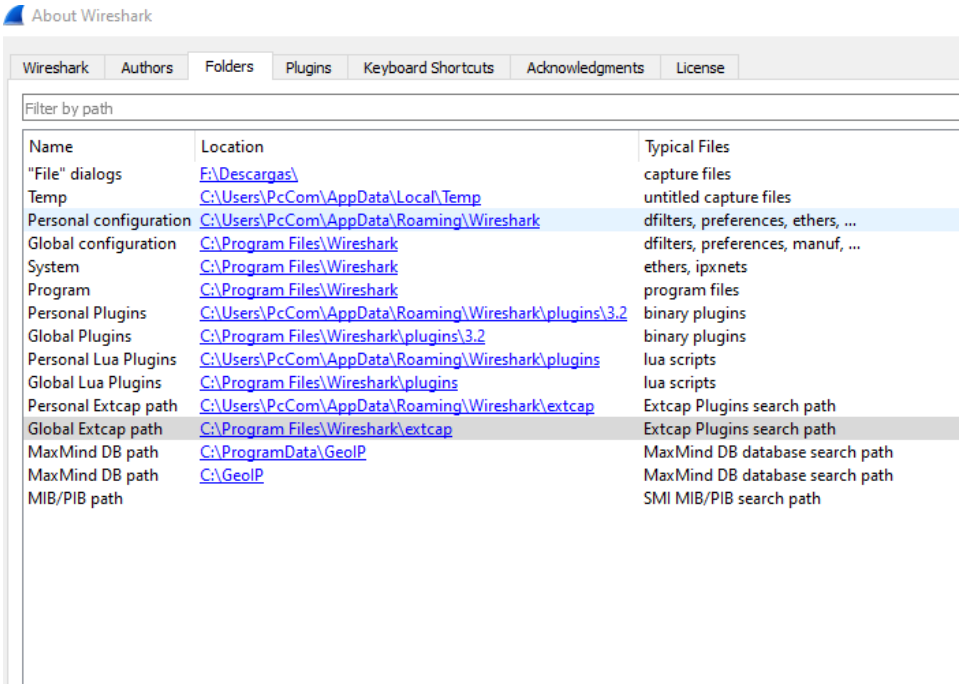

Figura 10: About Wireshark => Folders.

e) Copio el contenido de la carpeta Sniffer_Software / extcap / en esta carpeta.

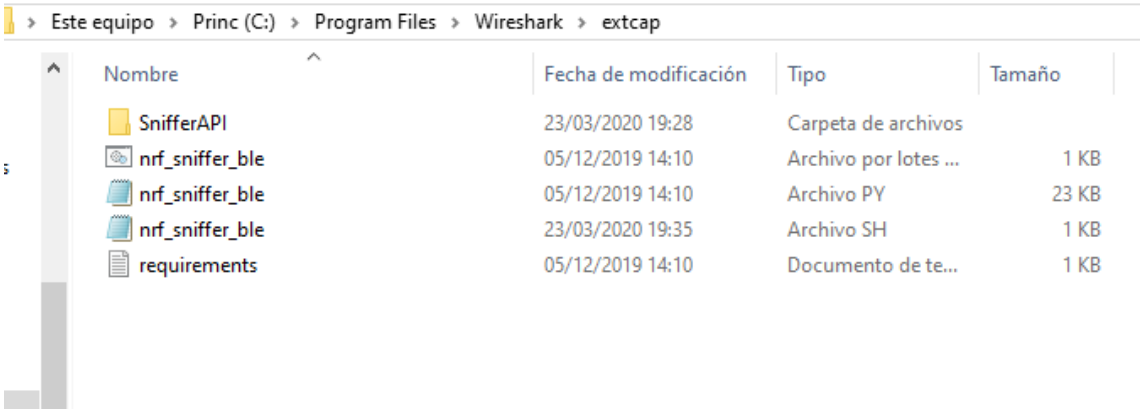

Figura 11: Extcap.

3. Nos aseguramos que los archivos nRF Sniffer se pueden ejecutar correctamente:

a) Abro una ventana de comando en la carpeta de Wireshark para complementos de captura externos.
b) Ejecuto la herramienta Sniffer para enumerar las interfaces disponibles.

En Windows,se escribe nrf_sniffer_ble.bat --extcap-interfaces.

Se debe ver algo similar a lo que se muestra en la siguiente captura de pantalla.

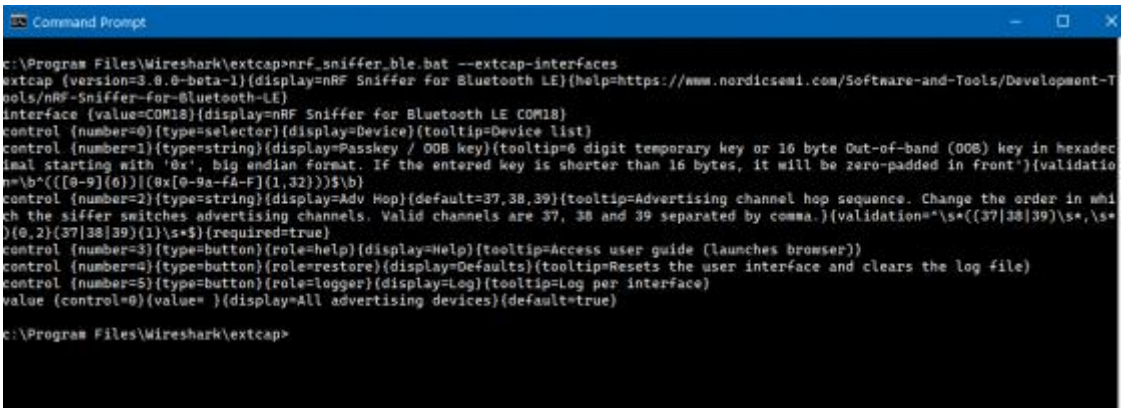

Figura 12: Respuesta consola.

c) Si el paso anterior da error, verifico si Python 3 es accesible.

En Windows, se usa el comando python --version.

4. Habilito la herramienta de captura nRF Sniffer en Wireshark:

a) Actualizo las interfaces en Wireshark seleccionando Capture> Refresh Interfaces o presionando F5.

Se debe ver que nRF Sniffer se muestra como una de las interfaces en la página de inicio.

b) Selecciono View > Interface Toolbars > nRF Sniffer for Bluetooth LE para habilitar la interfaz Sniffer.

4.1.5 Agregar un perfil de Wireshark para nRF Sniffer, [13]

Se puede agregar un perfil en Wireshark para mostrar los datos registrados por el nRF Sniffer para Bluetooth LE.

Para agregar el perfil Sniffer nRF en Wireshark, hay que seguir los siguientes pasos:

1. Go to Help > About Wireshark
2. Selecciono Folders.
3. Doble clic en la ubicación de Personal configurationl para abrir esta carpeta.

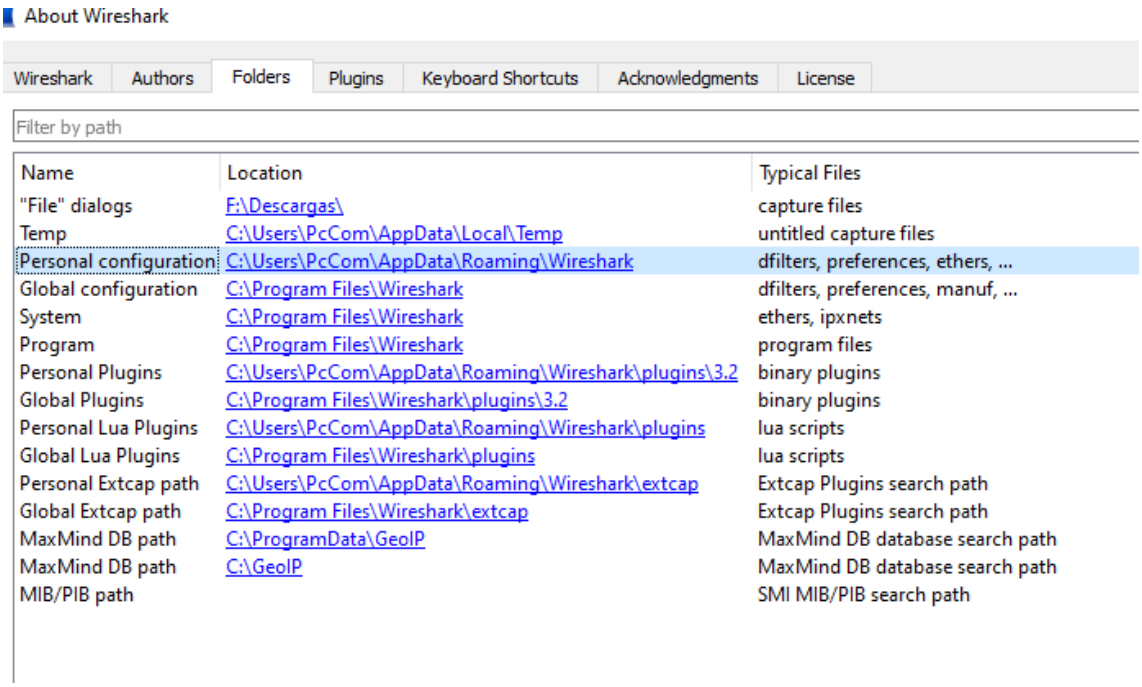

Figura 13: About Wireshark => Folders => Personal configuration.

4. Copio la carpeta de perfil Sniffer_Software / Profile_nRF_Sniffer_Bluetooth_LE en la subcarpeta de perfiles de esta carpeta.

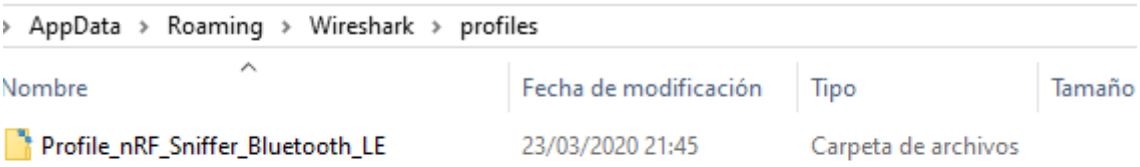

Figura 14: Profiles.

5. En Wireshark, selecciono Edit > Configuration Profiles.
6. Selecciono Profile_nRF_Sniffer_Bluetooth_LE y pulso OK.

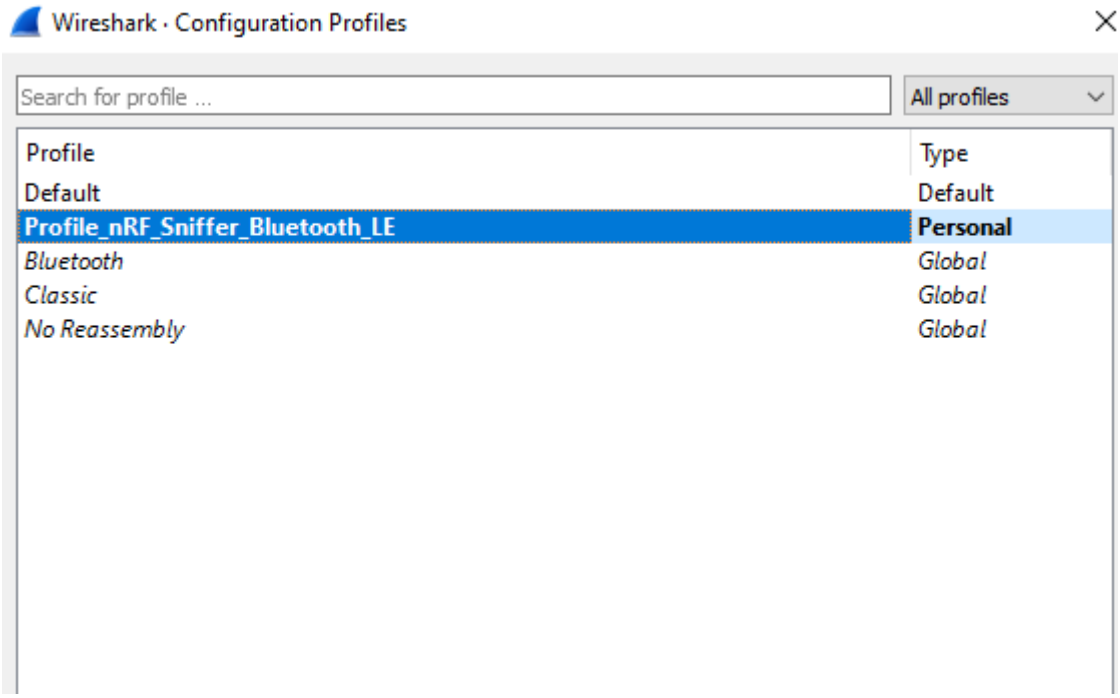

Figura 15: Profile_nRF_Sniffer_Bluetooth_LE.

4.2 API for asset tracking

Lo primero que hacemos para poder comprobar la API es descargar e instalar el programa Postman, que nos permitirá realizar las consultas.[15]

Una vez instalado debemos configurar Postman para poder obtener los datos mediante Postman, para ello seguimos los siguientes pasos:

1) Pulsamos la pestaña de Launchpad y le damos a Create a request

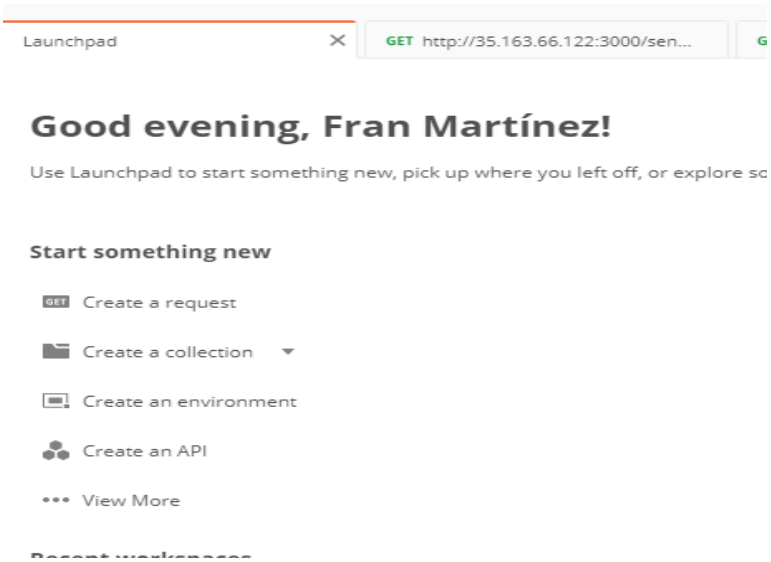

Figura 16: Launchpad.

2) Se abrirá una pestaña donde aparecerá que selecciones un método, que escribas una URL, y tres columnas en query params llamadas key, value y description.

Figura 17: Pestaña consulta.

3) El siguiente paso es rellenar con los datos de nuestra API.
   - En método seleccionamos GET
   - En URL usamos la URL de nuestra API
   - API Key: Esta clave la proporciona el equipo de sensegiz, la clave debe mencionarse con la URL para cada solicitud.
   - Parámetros opcionales: Hay 2 parámetros opcionales que dependen entre sí. El primer parámetro opcional contiene el nombre y el segundo parámetro opcional contiene el valor asociado del nombre dado en el primer parámetro opcional.

     Los primeros parámetros opcionales son:
     ➢ gateway_id => Especifica el ID de la puerta de enlace
     ➢ device_id => Especifica coin id
     ➢ safr_id => Especifica find id
     ➢ user_name => Especifica el nombre dado a la búsqueda en dashboard
     ➢ ubicación => Especifica el nombre dado a la coin en dashboard
     ➢ from => Especifica la fecha de inicio
     ➢ to => Especifica la fecha de finalización

Un ejemplo de esta configuración es la siguiente consulta:

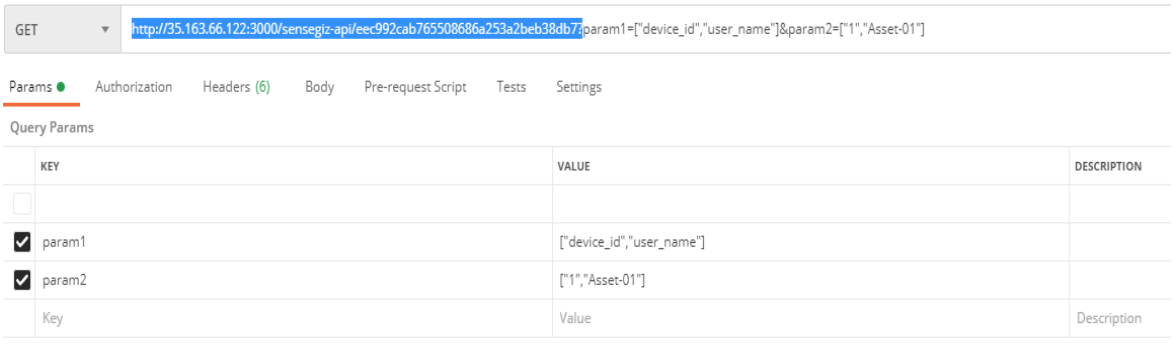

Figura 18: Ejemplo consulta.

4.3 Sensegiz Coin app

Esta aplicación es utilizada para configurar la gateway en el lugar donde vas a trabajar para que empiecen a funcionar los diferentes dispositivos, los pasos para configurarla son los siguientes:

1) Entro con mi usuario y añado la gateway

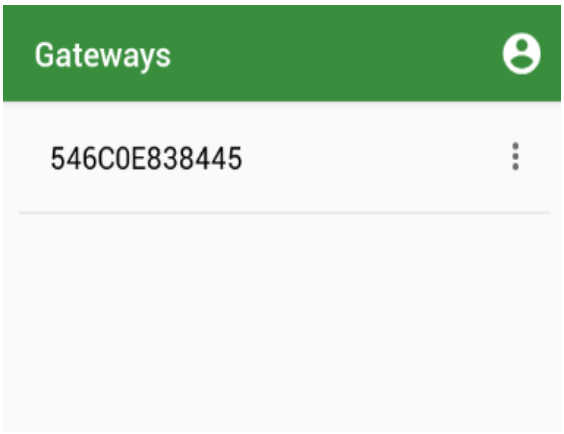

Figura 19: Añadir gateway.

2) El siguiente paso es darle a Configuration Settings.

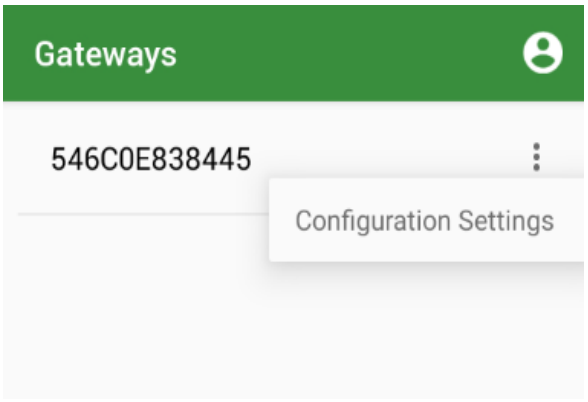

Figura 20: Configuration Settings.

3) Buscamos nuestra red Wifi y conectamos la gateway a ella.

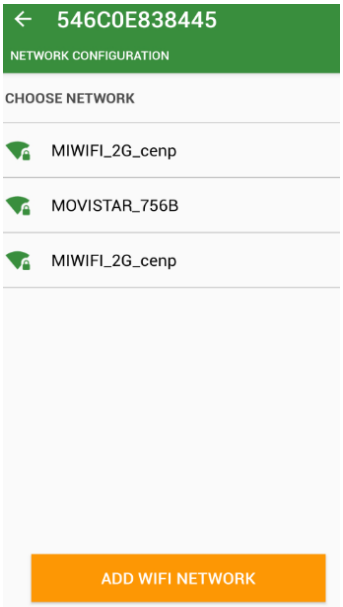

Figura 21: Agregar Wifi

4) Después de agregar la red Wifi, pasamos al último paso que consiste en escribir la URL del server y el número de puerto.

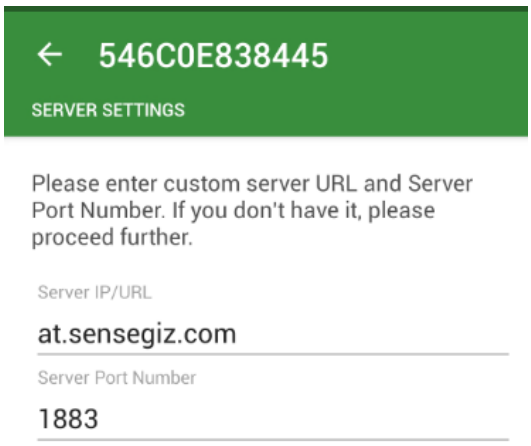

Figura 22: URL y puerto.

4.4 Agregar un mapa nuevo

1) Hay que hacer clic en Add new location.
2) Luego escribir el nombre y descripción del mapa en el lugar correspondiente.
3) Seleccionamos la gateway que estamos utilizando
4) En floor plan seleccionamos la imagen de nuestro mapa.

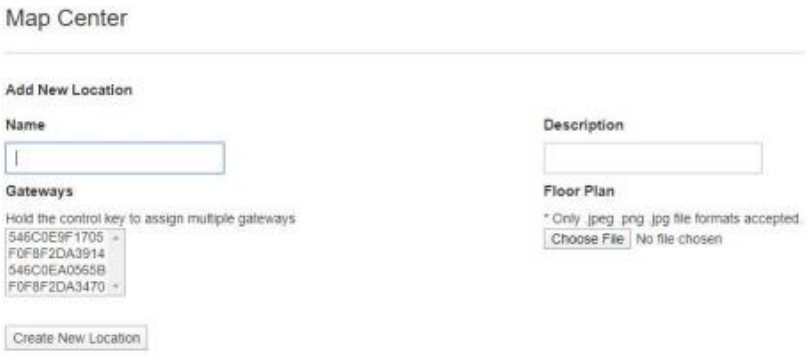

Figura 23: Agregar un mapa.

Además de agregar un mapa hay más opciones como ver la localización de los assets en vivo, o añadir/editar o borrar una coin.

Sl. No	Location Name	Location Description	Location Image	Monitor Location	Search Assets	Add/Edit Coins	Delete Location
1	Casa	piso 3	casa.jpg	Live Monitor	Search/Find Asset	Add/Edit	

Figura 24: Otras opciones mapa.

El mapa utilizado para las pruebas es el siguiente:

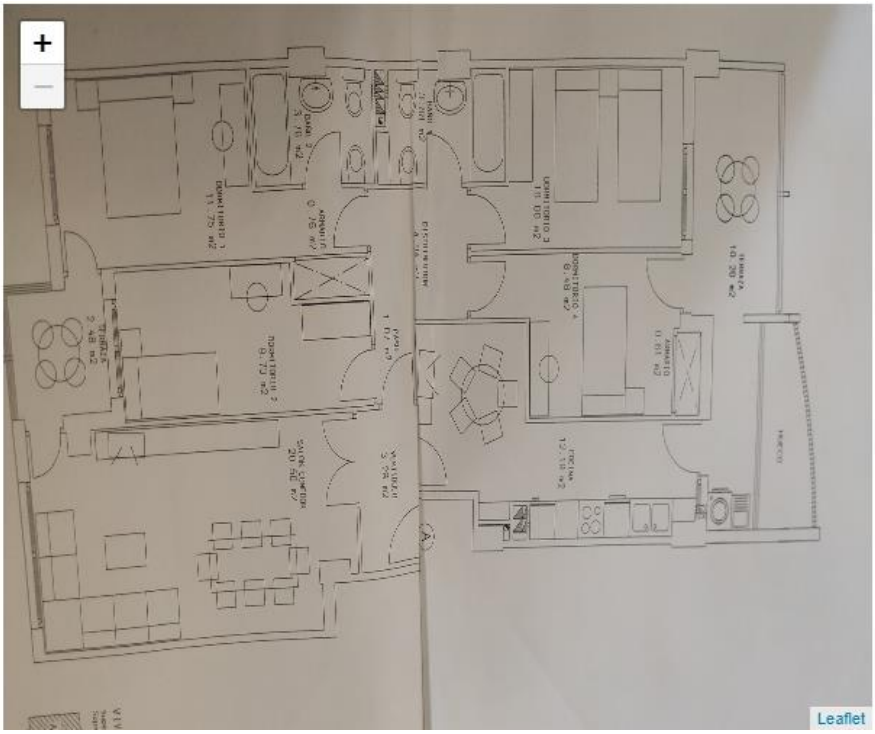

Figura 25: Mapa 2 sin dispositivos.

4.5 Inventario dispositivos y MAC

Aquí podemos observar un inventario donde se muestran los dispositivos y su respectiva MAC, esto se ha usado para facilitar el reconocimiento de cada dispositivo en el desarrollo del proyecto.

Tabla 1: Inventario.

Dispositivo	**MAC**
Gateway	54-6C-0E-83-84-45
Coin 1	C4-64-E3-8D-2A-5E
Coin 2	C4-64-E3-8D-26-B4
Coin 3	C4-64-E3-8D-36-7B
Coin 4	C4-64-E3-8D-3F-4F
Coin 5	C4-64-E3-8D-2A-51
Asset 1	7C:EC:79:DC:BC:89
Asset 2	7C:EC:79:DD:D2:22
Asset 3	7C:EC:79:C0:60:3E

5. PRUEBAS Y RESULTADOS

Antes de empezar debemos conectar la gateway al ordenador, una vez salga online, encendemos las coins, cuando aparezcan en la página web que están online seguimos con el siguiente paso que es encender los assets. Es importante seguir este orden para el correcto funcionamiento de los dispositivos.

5.1 Pruebas de funcionamiento (Mapa 1)

En este apartado se muestran las pruebas recogidas en el primer mapa utilizado, que era el que estuve usando hasta que empezó el confinamiento, donde tuve que utilizar el mapa de mi hogar.

El mapa era el siguiente:

Figura 26: Mapa 1 con dispositivos.

Tabla 2: Datos Mapa 1.

Asset	Origen	Destino	Tiempo actualización
1	2	1	7 minutos
2	1	5	20 minutos
2	3	1	7:55 minutos
3	4	1	1:30 minutos
3	1	2	1:25 minutos

- Los resultados recogidos en este primer mapa y como indica la tabla, nos dicen que el tiempo que tarda en actualizar el Asset 1 cuando lo movemos de Coin 2 a Coin 1 es de una media de 7 minutos.
- Para mover Asset 2 de Coin 3 a Coin 1 tarda unos 7:55 minutos y en el recorrido completo de Coin 1 a Coin 5 ha tardado alrededor de 20 minutos.
- Y finalmente para mover Asset 3 de Coin 4 a Coin 1 el tiempo es aproximadamente 1:30 minutos y de Coin 1 a Coin 2 tarda cerca de 1:25 minutos

5.2 Estudio de cobertura (Mapa 1)

Haciendo pruebas con respecto a la distancia a la que se pierde la cobertura nos damos cuenta que si logramos una distancia de separación entre dos coins de aproximadamente 95-100 metros se pierde la cobertura por lo que esto nos indica que esta distancia sería su límite de cobertura para funcionar correctamente, para lograr esta distancia se han apagado todos los coins excepto 2, para poder lograr esa distancia sin problemas.. A una distancia menor, podemos observar que sigue funcionando correctamente y solo varía el tiempo en que se mueve el Asset de una Coin a otra.

5.3 Pruebas de funcionamiento (Mapa 2)

Debido al confinamiento, se ha tenido que hacer un cambio en el mapa, debido a que tuve que realizar las pruebas desde mi casa, por lo que tuve que actualizar el mapa en la página web y hacer el mismo tipo de pruebas de funcionamiento que en los apartados anteriores. El nuevo mapa es el perteneciente a mi vivienda como se muestra en la “Figura 27”.

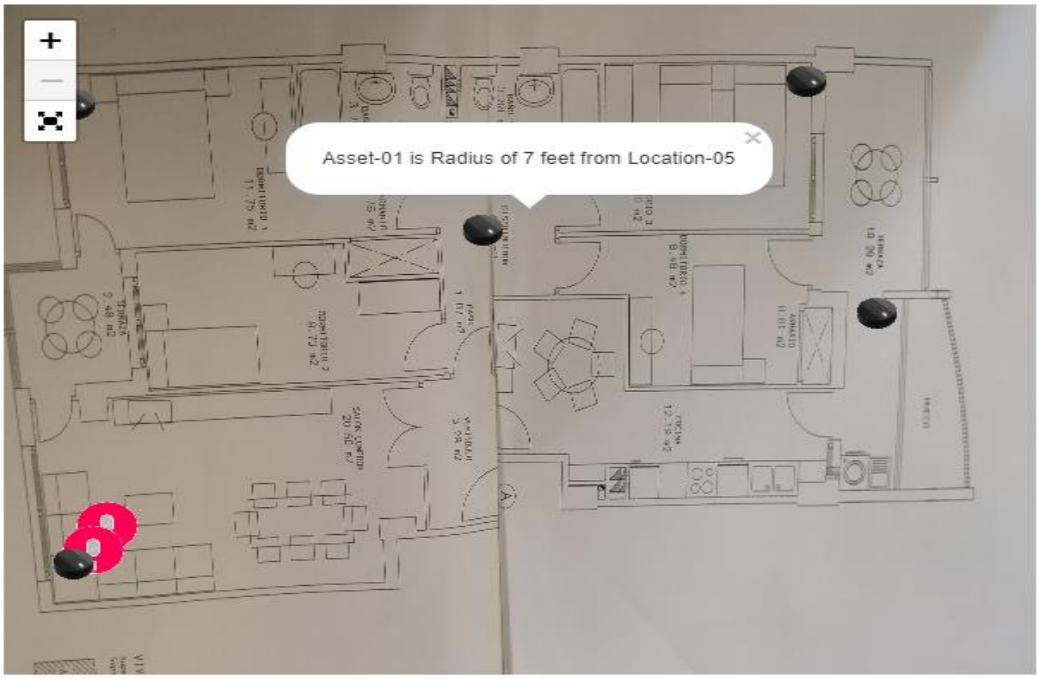

Figura 27: Mapa 2 con dispositivos.

Las pruebas realizadas para probar la precisión son las siguientes:

Tabla 3: Datos Mapa 2.

Asset	**Origen**	**Destino**	**Tiempo actualización**
1	2	1	4/5 minutos
2	2	1	4/5 minutos
1	1	2	4:05 minutos
2	1	2	25 segundos
1	2	3	4:45 minutos
2	2	3	4:20 minutos
1	4	5	12 minutos
2	4	5	1 minuto
1	5	2	3:15 minutos
2	5	2	20 minutos

- Se puede observar que para ir de Coin 2 a Coin 1 con Assets 1 y 2 se tarda entre 4 y 5 minutos en actualizar la posición. Esta es la mayor distancia entre dos Coins que hay en este mapa.

- Esta imagen corresponde al cambio de ubicación desde el Coin 1 al Coin 2 en el cual el Asset 2 tarda en actualizarse 25 seg y el asset 1 tarda 4:05 minutos. Como vemos en este caso hay bastante diferencia en la precisión de los dos Asset.

- La siguiente prueba es del Coin 2 al Coin 3, en este paso el Asset 1 tarda en actualizar 4:45 minutos y el asset 2 tarda 4:20 minutos, en este caso tienen una precisión similar.

- En la prueba del cambio de Coin 4 a Coin 5 tarda en actualizarse en el mapa 1 minuto el Asset 2 mientras que el Asset 1 tarda 12 minutos, aquí vuelve a observarse una gran diferencia en cuanto a la precisión.

- En el siguiente cambio que es mover los Assets desde Coin 5 a Coin 2 el Asset 1 ha tardado 3:15 minutos en actualizar su posición en el mapa mientras que el Asset 2 ha tardado 20 minutos.

5.4 Estudio de cobertura (Mapa 2)

En cuanto a la cobertura, al ser este mapa más pequeño, no hay forma de lograr una distancia de mínimo 95-100 metros entre dos Coins por lo que no habría ningún problema de cobertura en una vivienda que sea un piso, ya que los Coins abarcan toda la casa y no hay ningún punto que no. Por lo que en este caso no se ha podido comprobar la pérdida de cobertura.

5.5 Geofences

Se usarán las Geofences para crear alertas por SMS / correo electrónico según las preferencias del usuario, ya sea que el usuario necesite una alerta cuando un FIND se encuentre dentro de la geofence o fuera de ella, sirve para saber los movimientos que ha seguido el asset a lo largo del tiempo y además que te avise cuando cambia de localización o lleva un tiempo inactivo. Esto es bueno para saber si el anciano está bien, y si ves que lleva tiempo inactivo puedes contactar con él para ver si se encuentra bien.

Figura 28: Configuración alarmas.

5.6 Api for Asset Tracking

Para este apartado Sense4Location nos ha proporcionado una clave de acceso a la API, en la cual podemos realizar consultas mediante el uso del método GET para filtrar los cambios de posición de los Assets.

Los parámetros de La API son los siguientes:

URL

35.163.66.122:3000/Sense4Location-api

MÉTODO

GET

PARÁMETROS URL

1. **API KEY**

   eec992cab765508686a253a2beb38db7

   La URL final queda como:

   35.163.66.122:3000/Sense4Location-api/eec992cab765508686a253a2beb38db7

2. **PARÁMETROS OPCIONALES**

   - gateway_id: id de la gateway
   - device_id: id de las coins
   - safr_id: id de los finds
   - user_name: nombre dado a find
   - location: nombre dado a la coin
   - from: fecha inicio
   - to: fecha final

Las consultas realizadas han sido las siguientes:

Consulta 1

http://35.163.66.122:3000/Sense4Location-api/eec992cab765508686a253a2beb38db7?param1=[%22gateway_id%22]&param2=[%22546C0E838445%22]

En el programa Postman quedaría la configuración de esta forma:

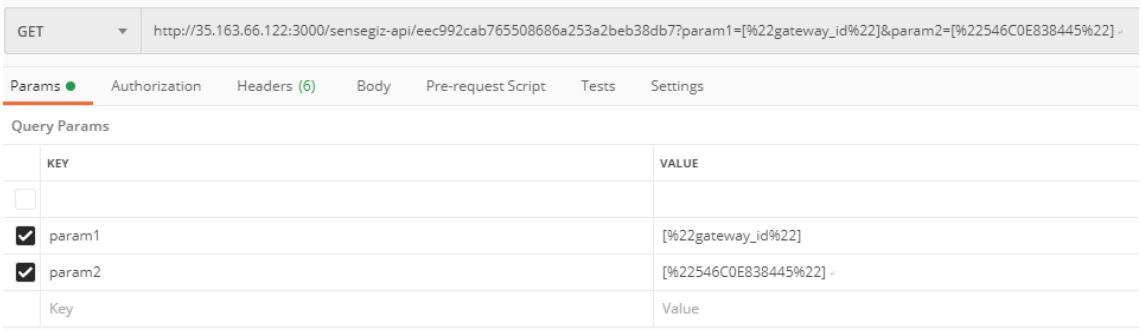

Figura 29: Ejemplo consulta 1.

Al poner esta consulta en el navegador nos da la respuesta, la respuesta sería la siguiente:


```
{"status":"success","data":
[{"id":139324,"user_id":78,"gateway_id":"5
46C0E838445","device_id":1,"safr_id":"1","
user_name":"Asset-
01","location":"Location-
01","type":"Geofence not
Configured","updated_on":"2020-02-
12T13:03:46.000Z","distance":"2","out_time
":"2020-02-
13T09:47:13.000Z","user_type":"1","total_t
ime":"00:0:0:20:4","status":0},
{"id":139328,"user_id":78,"gateway_id":"54
6C0E838445","device_id":3,"safr_id":"3","u
ser_name":"Asset-03","location":"Location-
03","type":"Geofence not
Configured","updated_on":"2020-02-
12T13:09:00.000Z","distance":"1","out_time
":"2020-02-
12T13:25:26.000Z","user_type":"1","total_t
ime":"00:0:0:00:1","status":0},
{"id":139330,"user_id":78,"gateway_id":"54
6C0E838445","device_id":4,"safr_id":"2","u
ser_name":"Asset-02","location":"Location-
04","type":"Geofence not
Configured","updated_on":"2020-02-
12T13:14:48.000Z","distance":"1","out_time
":"2020-02-
12T13:32:40.000Z","user_type":"1","total_t
ime":"00:0:0:00:1","status":0},
{"id":139335,"user_id":78,"gateway_id":"54
6C0E838445","device_id":1,"safr_id":"3","u
ser_name":"Asset-03","location":"Location-
01","type":"Geofence not
Configured","updated_on":"2020-02-
12T13:28:44.000Z","distance":"1","out_time
```


Figura 30: Respuesta navegador.

En este caso nos devuelve success y la información de id, gateway_id… además de todos los cambios de Geofence que ha habido, ya que solo filtra por la gateway y no por los distintos assets.

Consulta 2

35.163.66.122:3000/Sense4Location-api/eec992cab765508686a253a2beb38db7?param1=[%22device_id%22,%22user_name%22]&param2=[%221%22,%22Asset-01%22]

En el programa Postman la consulta quedaría de la siguiente forma:

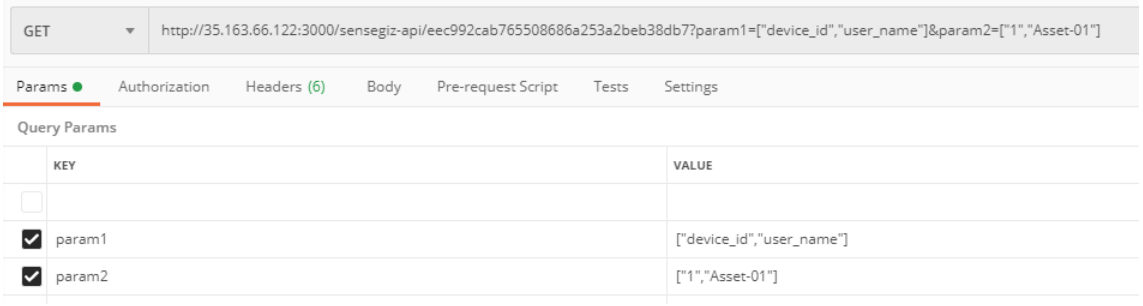

Figura 31: Consulta 2.

En este caso hemos filtrado cogiendo de parámetros el coin 1 y el Asset 01 por lo que esto provoca que nos devuelva los distintos geofences que ha sufrido el Asset 01, en nuestro caso tenemos configurado que nos envíe una alerta al correo cuando un Asset entra o sale de una localización, esta alarma es lo que se refleja en el Postman.

Ejemplo de la alerta del Asset 01 entrando a location 1:

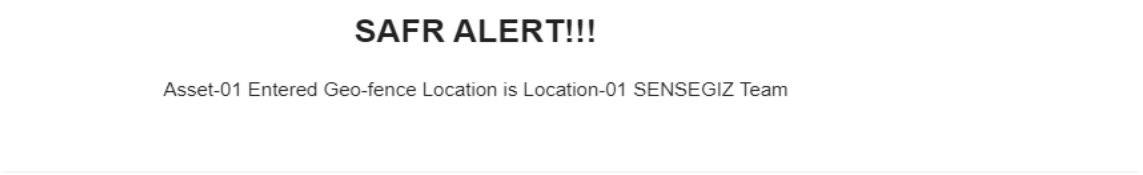

Figura 32: Alarma entrada.

Ejemplo de la alerta del Asset 01 saliendo de location 1:

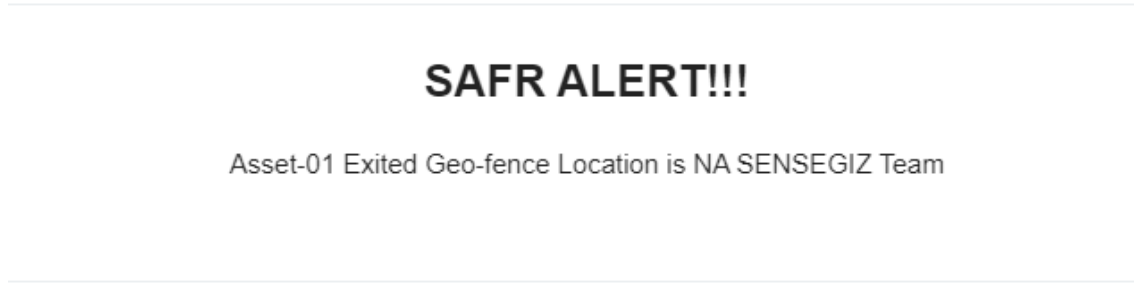

Figura 33: Alarma salida.

Hemos realizado diversas pruebas cambiando los parámetros device_id y user_name alternando con locations 1, 2, 3, 4, 5 y Assets 01, 02, 03.
En todos se ha podido observar cómo seguía enviándose una alerta en cada entrada o salida de un Asset en una location, por lo que se puede afirmar que el funcionamiento es el correcto.

En el postman se ha podido observar que no muestra todos los tipos de alertas de geofence porque el periodo de inactividad no lo muestra, mientras que el correo si lo envía cuando un Asset pasa mucho tiempo sin cambiarse de localización:

Ejemplo de la alerta del Asset 01 cuando hay un período de inactividad:

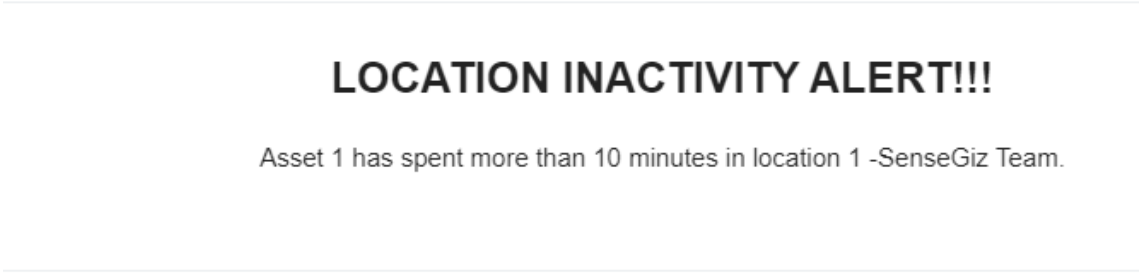

Figura 34: Alarma inactividad.

Otros ejemplos de consultas:

Localización 2 y Asset 02

35.163.66.122:3000/Sense4Location-api/eec992cab765508686a253a2beb38db7?param1=[%22device_id%22,%22user_name%22]&param2=[%222%22,%22Asset-02%22]

Localización 3 y Asset 03

35.163.66.122:3000/Sense4Location-api/eec992cab765508686a253a2beb38db7?param1=[%22device_id%22,%22user_name%22]&param2=[%223%22,%22Asset-03%22]

Localización 1 y Asset 02

35.163.66.122:3000/Sense4Location-api/eec992cab765508686a253a2beb38db7?param1=[%22device_id%22,%22user_name%22]&param2=[%221%22,%22Asset-02%22]

Localización 1 y Asset 03

35.163.66.122:3000/Sense4Location-api/eec992cab765508686a253a2beb38db7?param1=[%22device_id%22,%22user_name%22]&param2=[%221%22,%22Asset-03%22]

Localización 2 y Asset 01

35.163.66.122:3000/Sense4Location-api/eec992cab765508686a253a2beb38db7?param1=[%22device_id%22,%22user_name%22]&param2=[%222%22,%22Asset-01%22]

5.7 Resultados

En cuanto a los resultados, se ha podido observar que la precisión de los dispositivos no es la mejor, ya que hay veces que tardan poco en actualizarse pero otras veces tardan bastante tiempo en actualizar para ser un localizador a tiempo real, por lo que queda demostrado que es necesario un cambio en los tiempos de actualización para poder servir de más ayuda a la hora de vigilar el estado de los mayores.

En cuanto a geofence, se ha elegido distintas geofences para comprobar el funcionamiento del sistema, y como se ha podido comprobar, en el postman se pueden observar cuando ocurren las distintas geofence a excepción de la del período de inactividad, pero al programar alarmas en las geofence se puede observar que mandan un email con cada tipo de geofence distinta, por lo que se reconoce cada una.

Además se puede ver la utilidad de Postman de cara a ver el historial de un determinado asset a lo largo del tiempo, ya que te muestra la geofence, la localización a donde entra o de la que sale, con los datos respectivos a la fecha, hora y lugar donde ocurre la geofence.

6. PROBLEMAS Y RESOLUCIÓN

Los problemas que han surgido han sido cuando se ha gastado la batería de algún dispositivo que se ha solucionado sustituyendo la batería, otro problema ha sido el orden en el que hay que encender los dispositivos, primero el gateway, después las coins y por último los assets, sino se encendía en este orden podrían dar problemas para conectarse. Y para finalizar, el último problema de la primera parte del proyecto es en los momentos en que la página del desarrollador era modificada, los distintos dispositivos no funcionaban correctamente, pero se solucionaba cuando terminaban de modificar la página.

En la segunda parte del proyecto ha surgido algún problema más, uno de los problemas ha sido que la interfaz de configuración de los sms y alertas no iba bien, ya que no enviaba los mensajes a la opción seleccionada, pero se logró solucionar al ver que había que marcar lo contrario.

Otro de los problemas que han surgido ha sido que en los Assets no afectaba el cambio de baterías para el porcentaje de batería marcado por la página web, siempre marcaba lo mismo aunque pusieras nuevas pilas.

Finalmente, debido a la corta distancia entre location 4 y location 2, la aplicación marcaba que los assets estaban en la localización 4 en lugar de la 2, por lo que se solucionó cambiando la transmisión de potencia al mínimo o también desactivando el location 4.

Sl. No	COIN ID	COIN LOCATION	CONNECTED GATEWAY	RSSI IDENTIFIER	TRANSMISSION POWER	MULTIPLIER (in seconds)	FIND STATUS PERIOD (in seconds)	EDIT	DELETE
1	1	Location-01	546C0E838445	SET	30 feet SET	SET	5 SET		
2	2	Location-02	546C0E838445	SET	30 feet SET	SET	5 SET		
3	3	Location-03	546C0E838445	SET	30 feet SET	SET	5 SET		
4	4	Location-04	546C0E838445	SET	30 feet SET	SET	5 SET		
5	5	Location-05	546C0E838445	SET	30 feet SET	SET	5 SET		

Figura 35:Configuración potencia.

7. CONCLUSIONES

La conclusión que podemos sacar de la tareas desarrolladas es que al abandonar el apartado teórico surgen distintos problemas que afectan al sistema en tiempo real, como por ejemplo ha ocurrido a veces al tener grandes tiempos de actualización al mover el Asset entre dos coins distintas, o también las veces que un coin tenía una potencia más alta y al no tener una distancia suficiente con el Coin donde se encontraba el Asset, aparecía el asset en una posición incorrecta. Pero también ha sido de utilidad abandonar el apartado teórico para encontrar la distancia máxima real, que he podido averiguar aumentando la distancia entre dos coins hasta que se produjo la pérdida de cobertura.

En cuanto a la parte de los geofences, he podido observar cómo funcionan las alertas, que esto es bastante útil para cuando esté usando el dispositivo alguna persona mayor, puesto que avisa en caso de algún movimiento extraño o inactividad, dando la respectiva alerta a los familiares que les serviría para informarse acerca del estado del mayor.

Además he podido trabajar con los distintos filtros para las diferentes alertas e información del equipo dependiendo de las consultas que he realizado, que es algo útil por si quieres repasar bien todo el movimiento de los Assets a lo largo del tiempo.

Por otro lado, los dispositivos utilizados al tener períodos de actualización tan altos a veces no serían tan precisos para un sistema a tiempo real, por eso sería interesante probar en el futuro estos mismos dispositivos pero con una precisión más adecuada al objetivo de seguimiento de las personas mayores en el hogar, para tener unos resultados más realistas.

8. BIBLIOGRAFÍA Y REFERENCIAS

[1] Cámaras de videovigilancia: http://www.alzfae.org/fundacion/447

[2] Safety Solutions & People Tracking: https://sensegiz.com/safety-solutions-and-people-tracking/

[3] BLE (Bluetooth Low Energy):

https://www.elt.es/ble-bluetooth-low-energy#:~:text=HCI%20es%20un%20protocolo%20est%C3%A1ndar,trav%C3%A9s%20de%20un%20interfaz%20serie.&text=Para%20BLE%2C%20la%20capa%20L2CAP,a%20los%20dos%20protocolos%20fundamentales.

[4] Bluetooth Low Energy – Introducción a la tecnología:

https://smart-lighting.es/bluetooth-low-energy-introduccion-la-tecnologia/

[5] Advantages of BLE and disadvantages of BLE: https://www.rfwireless-world.com/Terminology/Advantages-and-Disadvantages-of-BLE-Bluetooth-Low-Energy.html

[6] PDU types: https://www.novelbits.io/bluetooth-low-energy-sniffer-tutorial-advertisements/

[7] Coin: https://sensegiz.com/coin/

[8] Gateway: https://sensegiz.com/gateway/

[9] Placa: https://www.nordicsemi.com/Software-and-Tools/Development-Kits/nRF52-DK

[10] ¿Qué es Postman?: http://www.arquitectoit.com/postman/que-es-postman/

[11] Wireshark: https://es.wikipedia.org/wiki/Wireshark

[12] nRF Connect for Desktop: https://www.nordicsemi.com/Software-and-tools/Development-Tools/nRF-Connect-for-desktop

[13] Guía de instalación de nRF_Sniffer_BLE: https://infocenter.nordicsemi.com/pdf/nRF_Sniffer_BLE_UG_v3.1.pdf

[14] Installing the Programmer app: https://infocenter.nordicsemi.com/index.jsp?topic=%2Fug_nc_programmer%2FUG%2Fcommon%2Fnrf_connect_app_installing.html

[15] Postman: https://www.postman.com/downloads/