

MANUAL DE USUARIO

ROBOMAP

ROBOMAP es una plataforma robótica desarrollada por el Instituto Universitario de Ciencias y Tecnologías Cibernéticas (IUCTC) en la línea de investigación de Sistemas de Posicionamiento en Interiores o **IPS** (del inglés *Indoors Positioning System*).



Figura 1. Plataforma robótica ROBOMAP

La metodología actual que utiliza el IUCTC en esta línea de investigación es la que se conoce como Análisis de escena o **Fingerprinting**. Se basa en la captura y estudio de la Intensidad de Señal Recibida o **RSS** (del inglés *Received Signal Strength*) de diferentes emisores de señal en dos situaciones diferentes, la fase de calibrado y la fase de posicionamiento. La primera, también conocida como fase *offline*, consiste en un estudio previo del entorno en el que se creará una base de datos con la que contrastar la información obtenida durante la segunda, también llamada fase *online*.

El proceso de captura de datos durante el análisis del entorno puede llegar a consumir bastante tiempo y, además, se deben tener en cuenta diversos factores que pueden afectar a la emisión y recepción de las señales, como son la temperatura y humedad del ambiente, la posición y orientación del dispositivo receptor, entre otros. Con la idea de **automatizar** este proceso nace ROBOMAP.

Este proyecto se ha ido trabajando por módulos a medida que diferentes miembros del IUCTC junto con estudiantes, del grado de Ingeniería Informática en la Universidad de Las Palmas de Gran Canaria (ULPGC), han hecho sus aportaciones como parte de sus **Prácticas Externas** y/o **Trabajos de Fin de Carrera**. De esta manera, surge la necesidad de crear este documento, una guía en la que se describe brevemente su diseño y cómo realizar una campaña de recogida de datos con éxito.

1. DISEÑO

ROBOMAP está estructurado en diferentes módulos que dotan a la plataforma de diferentes funcionalidades. Todos ellos, se comunican entre sí utilizando la **arquitectura cliente-servidor** con el **protocolo UDP** (del inglés *User Datagram Protocol*). De manera resumida, se describen a continuación:

- **Módulo ROBOMAP (MR):** Los compañeros Alberto Ramos Sánchez y José Carlos Rodríguez dotaron a la plataforma de movimiento omnidireccional y, con el uso de balizas ultrasónicas para determinar su posición, indicarle una ruta a seguir dentro del espacio designado.
- **Módulo Meteorológico (MM):** David Suárez Suárez desarrolló, utilizando una placa Arduino y una serie de sensores, un sistema de toma de datos de la luminosidad, temperatura y humedad del ambiente en la zona de trabajo. Estos valores son importantes ya que pueden influir en la emisión y recepción de las señales Bluetooth.
- **Módulo de Adquisición de Datos BLE (MAD-BLE):** Los compañeros Yousuf Boutahar El Maachi e Iru Nervey Navarro Alejo incorporaron en la plataforma un receptor de señales provenientes de diferentes emisores o *beacons* BLE (del inglés *Bluetooth Low Energy*).
- **Módulo Orientador del Receptor BLE (MOR-BLE):** También desarrollado por David Suárez Suárez, permite poner el dispositivo receptor BLE en un soporte capaz de rotar, con la finalidad de registrar las señales recibidas desde diferentes ángulos.
- **Módulo Controlador (MC):** Yousuf e Iru, diseñaron e implementaron lo que se puede denominar el director de la plataforma. Este, durante una campaña de recogida de datos, es el encargado de comunicarse con las demás partes, indicándoles cuándo activarse, cómo operar y dónde guardar los datos almacenados a lo largo de todo el proceso.
- **Módulo de alertas (MA):** El compañero Fabián A. Beirutti Pérez, implementó una nueva funcionalidad a la plataforma, un sistema de alertas que consiste en unos mensajes que muestran al usuario el resultado de diferentes controles de errores y el estado funcional durante una campaña. Además, se diseñó una simple GUI para la inspección de las alertas, llamada ROBOMAP Inspector.

2. PREPARACIÓN PREVIA

Actualmente, se necesitan una serie de equipos y herramientas software para iniciar el proceso de captura. Antes de realizar una campaña de recogida de datos, debemos comprobar que disponemos de todo el material y que esté debidamente cargado y configurado.

2.1 Ordenador principal

El ordenador principal es el centro de operaciones de ROBOMAP, esto es, lugar donde se inicializarán y controlarán todos los procesos en una campaña de captura de datos. A este equipo se conectará el Hub USB para permitir la comunicación con el resto de los dispositivos de ROBOMAP.

Durante la redacción de esta guía, es un portátil cedido por el IUCTC que contiene toda la información y código relativo a cada uno de los módulos en “C:/IPS”. Entonces, deberemos:

- Disponer de los archivos de programa y/o *scripts* correspondientes a todos los módulos (MR es en C/C++ y los demás en Python).
- Comprobar que los archivos estén ubicados dentro de las rutas especificadas de los ficheros de configuración de cada módulo (modificar las rutas si corresponde).

2.2 Ordenador auxiliar

El equipo auxiliar, ordenador portátil o no, será en el que nos apoyemos para realizar una campaña. Aquí, iniciaremos los programas de las balizas ultrasónicas y la de creación y comunicación de rutas para la plataforma. Entonces, deberemos:

- Tener instalado el programa de creación de rutas (herramienta de diseño Blender y el *add-on* desarrollado por Alberto).
- Tener instalado el software de control de las balizas ultrasónicas de Marvelmind.

2.3 Resto del material

Además de los ordenadores, principal y auxiliar, debemos comprobar que todos los módulos que conforman ROBOMAP disponen de todos sus componentes y, según corresponda, que tengan batería, estén conectados y/o configurados correctamente:



Figura 2. Baliza de ultrasonido de Marvelmind



Figura 3. Baliza BLE de Kontakt.io



Figura 4. Router CISCO

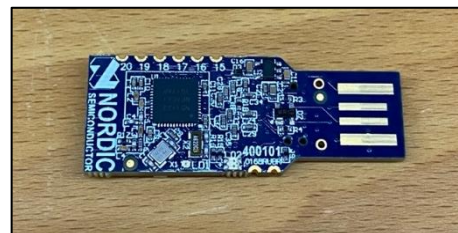


Figura 5. Dongle Bluetooth

El router de la Figura 4, configurado previamente por un compañero del IUCTC, crea una red local llamada “lynksis”, a la que conectaremos los ordenadores para establecer la comunicación entre ellos. Este no hubo necesidad de modificarlo a pesar de que a veces tardada en iniciar la red.

Hay que destacar que estas fueron las herramientas utilizadas durante la redacción de este documento. Esto no implica que sea la única opción, existen alternativas para alguno de los elementos, por ejemplo, se disponía de balizas BLE de otras dos compañías diferentes.

IMPORTANTE: Desactivar el modo ahorro de batería y modo de suspensión de pantalla para evitar posibles problemas con la ejecución de código.

3. REALIZAR UNA CAMPAÑA

De manera resumida, los pasos a seguir para realizar una campaña de captura de datos son:

1. Comprobar que se detectan las balizas de ultrasonido y se reciben señales de las BLE.
2. Encender el router CISCO y conectar los equipos, principal y auxiliar, a su red.
3. Configurar y distribuir las balizas BLE y de ultrasonido.
4. Crear o cargar la escena y la ruta a seguir.
5. Iniciar servidores MAD-BLE, MOR-BLE, MM y MA.
6. Iniciar servidor ROBOMAP en equipo principal y cliente ROBOMAP en equipo auxiliar.
7. Si se quiere seguir la campaña, iniciar ROBOMAP Inspector.
8. Iniciar MC.

3.1 Balizas ultrasonido

La herramienta software de [Marvelmind](#), también conocida como Dashboard, nos permitirá monitorizar y configurar las **balizas de ultrasonido**. Toda la información necesaria para la instalación del programa la encontramos en el apartado de descargas de la web oficial. En principio, con descargar la última versión nos bastará.



Figura 6. Baliza de ultrasonido y receptor de Marvelmind

Exceptuando dos balizas que se posicionan en la parte superior de ROBOMAP, al resto, nos referiremos como estáticas, ya que las distribuiremos dentro del espacio de prueba designado y no las moveremos hasta finalizar.

El receptor va conectado por cable al ordenador auxiliar y, las dos balizas estáticas, una vez posicionadas en la plataforma, se conectan por cable al Hub USB.

Toda la información necesaria sobre el Dashboard, así como, la imagen mostrada a continuación, donde se describen las partes que conforman la interfaz del programa, lo podemos encontrar en el [manual digital](#).

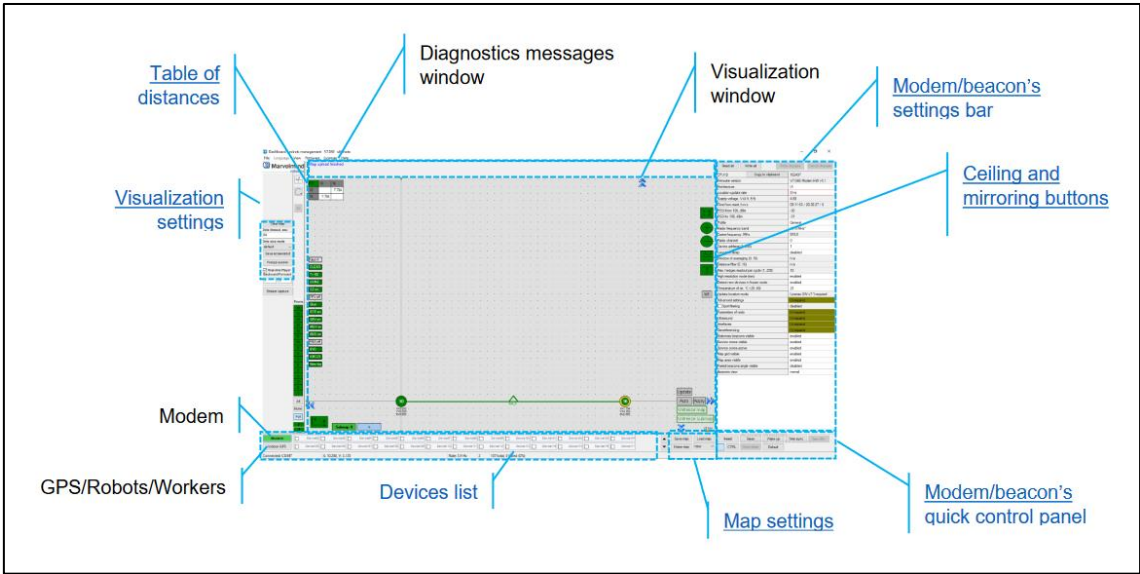


Figura 7. Vista general del Dashboard de Marvelmind

Comprobar la carga de las balizas es simple, debemos conectarla por cable al ordenador donde tengamos el software instalado, abrir el programa y verificar que el voltaje sea un valor dentro del rango indicado. Cuanto más cerca al mínimo, menor será el porcentaje de la batería y viceversa.

Supply voltage, V (3.50..4.30)	4.18
--------------------------------	------

Figura 8. Ejemplo de lectura del voltaje de una baliza

Para activar como poner en modo reposo las balizas, deberemos ir a la lista que aparece en la parte inferior de la pantalla, allí, al hacer doble clic en una de ellas cambiará su estado y se pondrá de color verde para las estáticas y azul para las móviles (en inglés *hedge*).

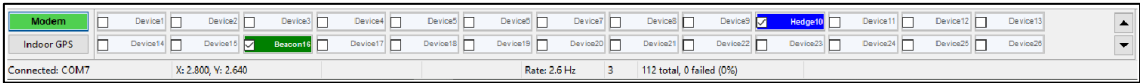


Figura 9. Lista de balizas en el Dashboard de Marvelmind

3.2 Kio Setup Manager

Las **balizas BLE** de [Kontakt.io](https://kontakt.io) se pueden monitorizar y configurar a través de una aplicación móvil denominada **Kio Setup Manager**. Sin embargo, para hacer uso de la aplicación necesitaremos iniciar sesión en Kio Cloud (pedir las credenciales a los tutores Alexis y/o Gabriele).

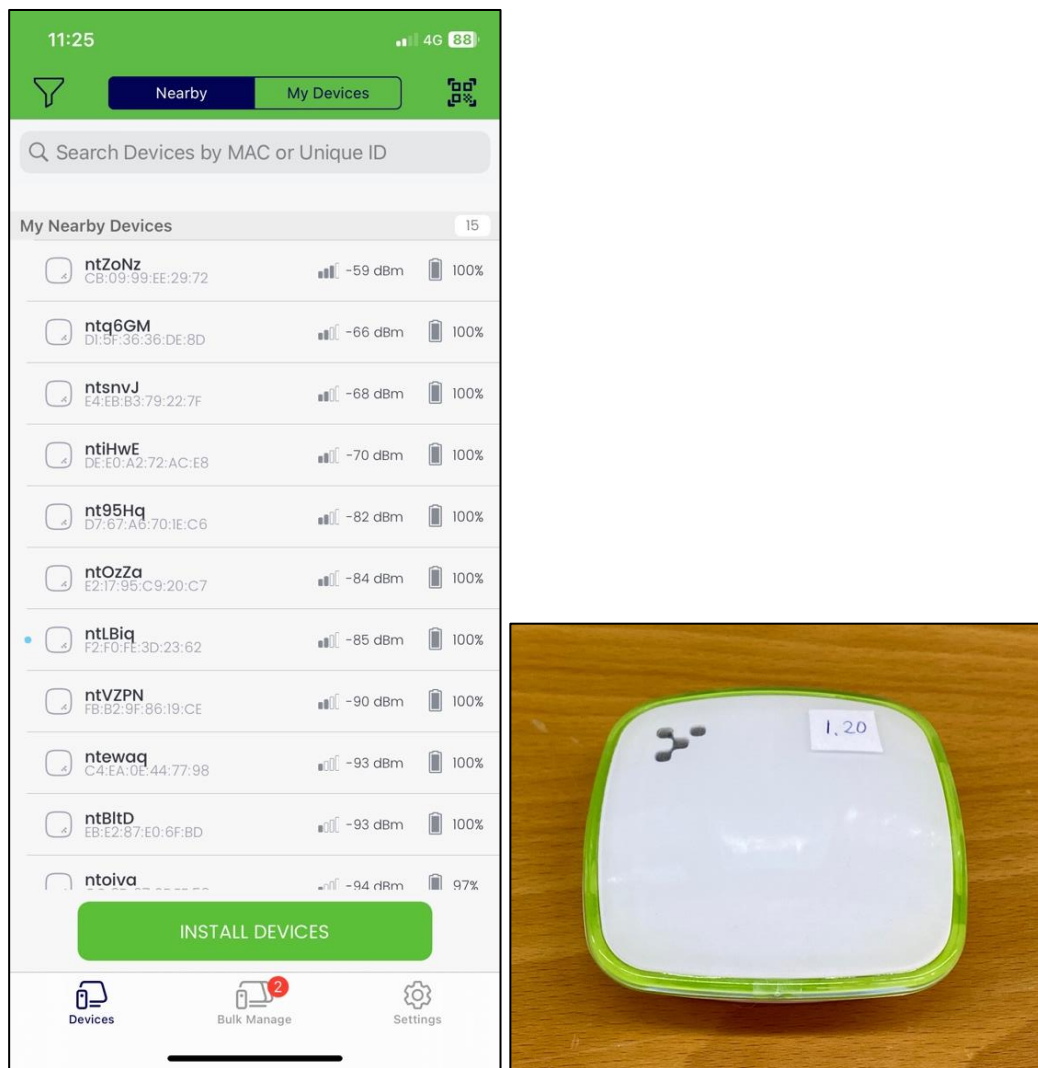


Figura 10. Aplicación móvil Kio Setup Manager y baliza BLE de Kontakt.io

Una vez tengamos la sesión iniciada y el Bluetooth del móvil esté activado, podremos ver los dispositivos vinculados a nuestra cuenta y aquellos de los que recibimos señales. Por último, pinchando en cada uno de ellos, accederemos a un menú más detallado donde podremos configurarlos a nuestro gusto.

3.3 Blender

La herramienta software [Blender](#), nos permite diseñar y modelar objetos en 3D. Utilizando una versión superior a la 2.8 y con la extensión “Robotcontrol editor toolbox” desarrollada por el compañero Alberto Ramos, podremos diseñar escenarios, rutas y controlar de forma remota a ROBOMAP.

Encontraremos información relevante sobre la instalación de la extensión, así como, pequeñas explicaciones sobre el manejo de la herramienta en su memoria de TFG y en el repositorio:

<https://github.com/bertoramos/blender-editor>

Una vez tengamos todo instalado correctamente podremos diseñar o importar una escena, que no es más que una representación digital del espacio físico donde realizaremos la campaña. Se hace básicamente para ubicar a la plataforma dentro del espacio designado y así detectar posibles colisiones con antelación. Para importar una escena, es tan simple como abrir el archivo .blend que lo contenga.

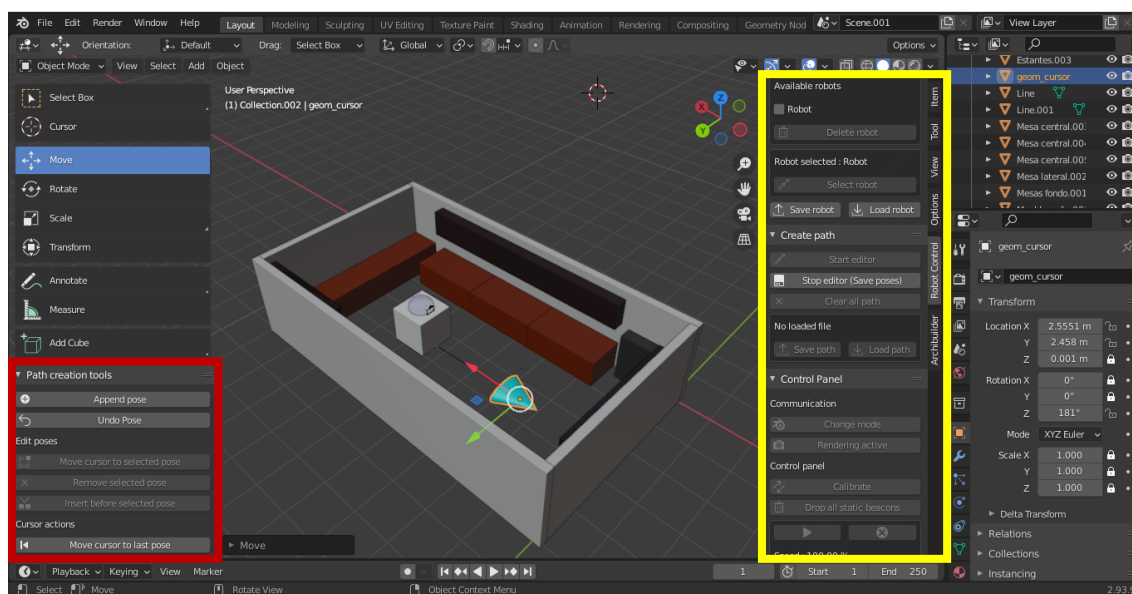


Figura 11. Menús y funcionalidades de la herramienta Robotcontrol en Blender

Al presionar la tecla ‘N’ de nuestro teclado deberíamos tener un menú desplegable igual al que vemos en el recuadro amarillo de la imagen anterior, que corresponde con una captura del software Blender con la escena que representa a la biblioteca del IUCTC. Y, si navegamos hasta la pestaña “Robot Control”, accederemos a dicha herramienta y podremos crear un robot.

Podremos modificar un poco los campos salvo las direcciones IP y los puertos, que sí deben ser comunes a los indicados en los archivos de configuración del equipo principal en ROBOMAP. A continuación, deberemos seleccionar el robot que queramos de los que nos salgan listados y podremos entonces utilizar las funciones deseadas, como diseñar una ruta o usar el control remoto.

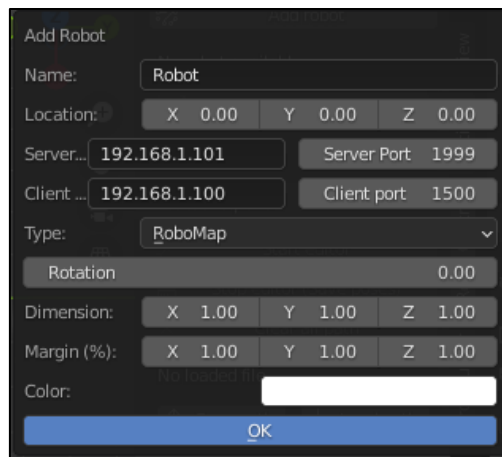


Figura 12. Ventana de creación de robot de la herramienta Robotcontrol en Blender

Para acceder al modo de edición y/o creación de rutas tendremos que seleccionar un robot y darle al botón “Start editor”. Nos debería salir el menú desplegable del recuadro rojo de la Figura 11 y ser visible el triángulo de color cian de la Figura 13. Una vez hayamos desplazado y/o cambiado la orientación del cursor, si queremos guardar esa posición utilizaremos la opción “Append pose” del recuadro rojo.

Para establecer una conexión con la plataforma desde Blender, debemos ejecutar el código en C/C++ del módulo ROBOMAP en el ordenador principal y, a continuación, clicar el botón del recuadro azul que aparece en la siguiente figura.

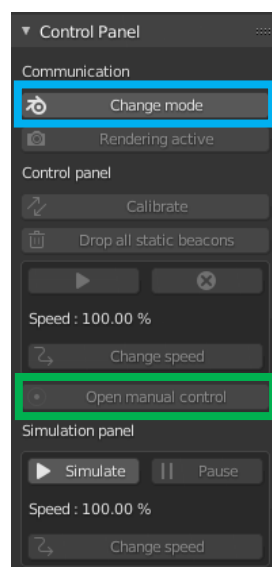


Figura 13. Panel de control de la herramienta Robotcontrol en Blender

Para manejar manualmente de manera remota a ROBOMAP, hay que establecer primero la conexión descrita en el párrafo anterior y, al clicar en el botón del recuadro verde, “Open manual control”, controlaremos el movimiento de la plataforma mediante el teclado.

Otra aspecto importante para tener en cuenta, es que tras establecer conexión con el MR, si pulsamos en el botón “Calibrate”, que se encuentra en el menú “Control Panel”, podremos ajustar la posición de las balizas de ultrasonido estáticas en la escena virtual.

3.4 R+ Manager

Con la herramienta software R+ Manager podremos configurar las características de cada uno de los motores DYNAMIXEL que conforman a la plataforma. [Manual en formato digital](#).

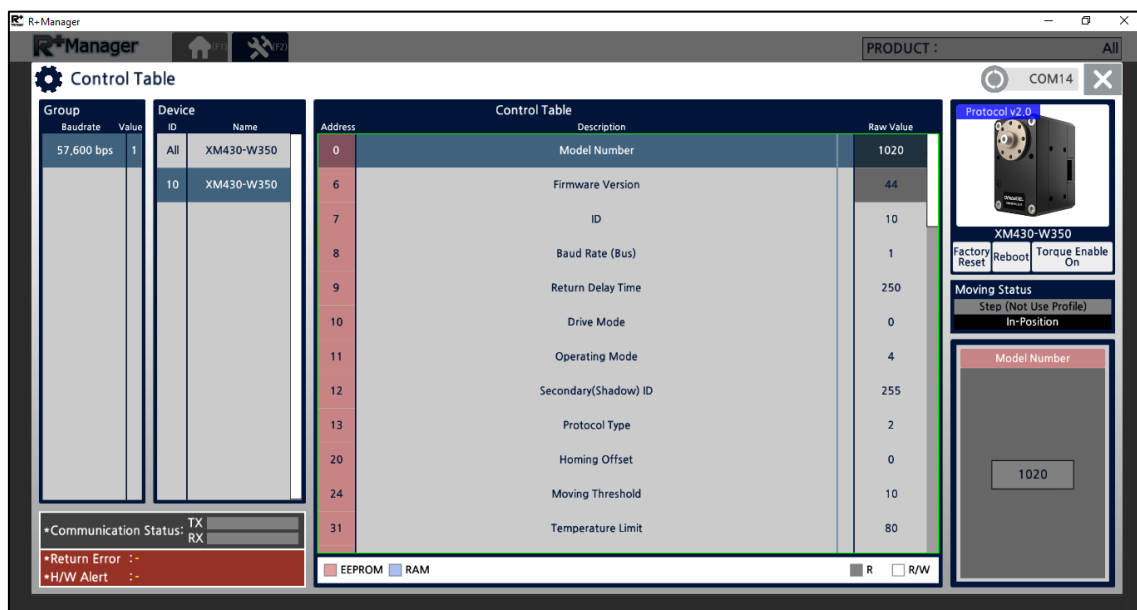


Figura 14. Ventana de configuración de los motores DYNAMIXEL en R+Manager

3.5 ROBOMAP Inspector

Es una interfaz gráfica desarrolla con PySimpleGUI que permite ver las alertas generadas durante una campaña, además, si establecemos la conexión con la plataforma una vez está ya allá comenzado, se puede solicitar el historial y ver las alertas previamente generadas en otro campo para no mezclar, las recibidas en directo y las antiguas.

Además, gracias a la librería pyinstaller, se consigue tener ejecutables como un programa más de ordenador, evitando así tener Python instalado o ejecutar los archivos desde la terminal. Por el momento solo se tiene el programa para Windows (.exe) pero es tan sencillo obtenerlo para Mac como ejecutar el siguiente comando:

```
“pyinstaller --noconsole --onfile nombre_del_archivo.py”
```

Se definió una alerta como un mensaje estructurado en dos partes: el nivel o importancia y la descripción. La primera indica la relevancia de la segunda. Además, cada nivel tiene asociado un color para aumentar la visibilidad de las alertas.

Color	Nivel	Significado
Verde	Ok	Todo correcto.
Amarillo	Warning	Algo puede estar mal, comprobar el cuerpo de la alerta.
Rojo	Error	Algo no salió como se esperaba.
Azul	Ack	Reconocimiento, por ejemplo, la conexión con el servidor.

Luego, en la carpeta “C:\IPS\alertsModule\” está el código y el ejecutable previamente generado de la interfaz y, en el archivo Servidor_MA se describen las funcionalidades de almacenamiento de las alertas y el código para las peticiones de conexión y envío del historial.

Por otro lado, en “C:\IPS\controladorModule\Service_Alerts” están los diferentes tests ejemplo elaborados, donde, cada uno de ellos genera una alerta según el resultado de las pruebas, solo cambia el cuerpo de la alerta en cada función.

Por último, la invocación de los tests o generación de alertas se hace desde el MC, tal y como si fuera un servicio más.

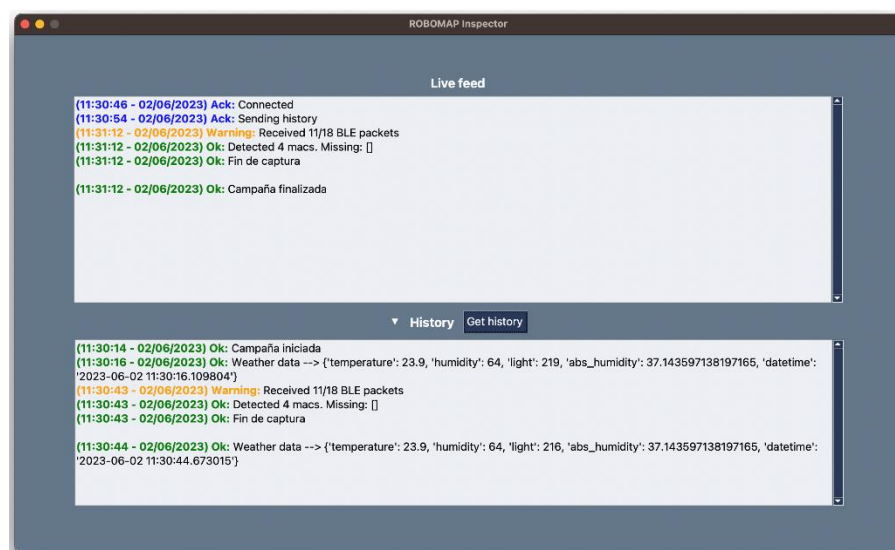


Figura 15. Interfaz gráfica ROBOMAP Inspector.

3.6 Ejecución de código

Los programas que describen el comportamiento de ROBOMAP deberemos ejecutarlos desde terminales diferentes.

- Servidores MAD-BLE, MOR-BLE y MM

“C:\IPS\controladorModule\desplegarServidores\despliegueServidores.py”

- Servidor MR en equipo principal

“C:\IPS\robomapModule\Release\robotips5_msvc2019.exe”

- Módulo Controlador

“C:\IPS\controladorModule\controlador.py”

4. POST-PROCESADO DE DATOS

Una vez finalizada la campaña para los puntos de referencia y los aleatorios, habrá que tratar la información almacenada en la base de datos (BBDD), lo que se conoce como post-procesado de datos, para ser utilizada más adelante con los algoritmos que estimarán la posición del objeto en cuestión.

Al momento de redactar este documento existen una serie de programas, en Python y Java, que obtendrán la información de la herramienta SQLite y la volcarán en archivos con formato “.csv” y “.xlsx”.

Con Python, extraeremos la información de la BBDD y por cada punto de referencia, crearemos diferentes archivos .csv que corresponden a la captura de señales BLE en cada una de las orientaciones indicadas al principio de la campaña. Se hará el mismo proceso con los puntos aleatorios que, aunque conocemos su ubicación al trasladar a ROBOMAP a dicha posición, será desconocido para el algoritmo de posicionamiento.

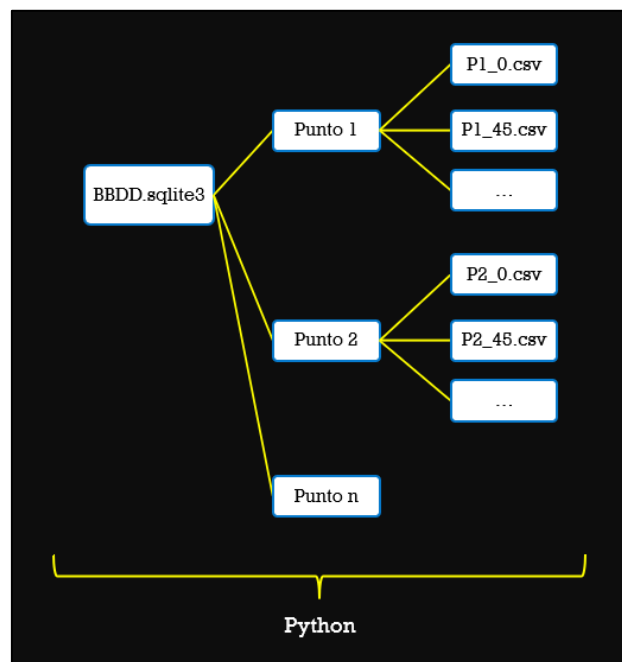


Figura 16. Estructura del procesamiento de datos con el programa en Python

Luego, con el programa descrito en Java, ubicaremos toda la información obtenida en un punto, en un único archivo .xlsx, el cual, estará dividido en diferentes hojas para separar las capturas de señales BLE según canal, protocolo y orientación.

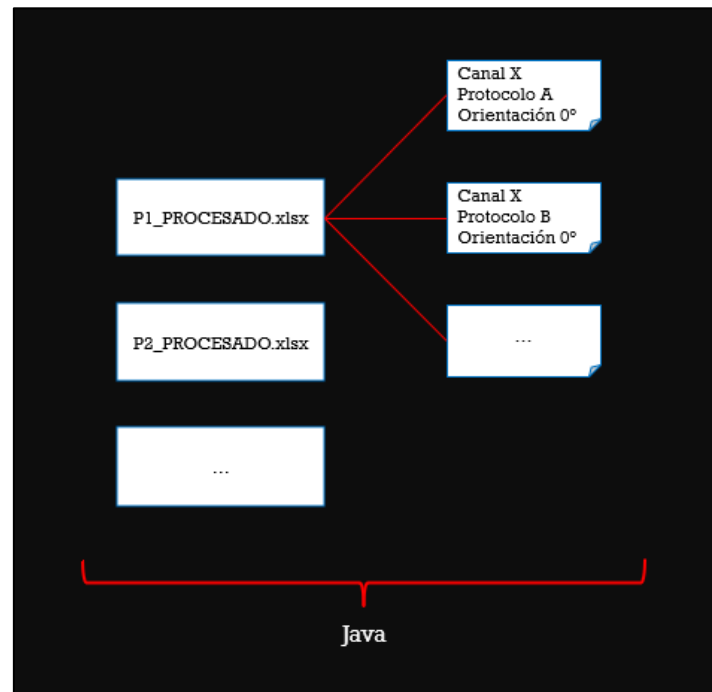


Figura 17. Estructura del procesamiento de datos con el programa en Java

4.1 Pasos

La carpeta “ProcesadoDatos” aloja todos los archivos del proyecto Java que realiza parte del post-procesado. En el mismo directorio, hay una carpeta llamada “Ejecutar_Programa” en la que encontraremos los ficheros necesarios para el resto de los pasos.

1. En la misma ubicación del archivo “Export_Files.py”, guardaremos las correspondientes bases de datos en formato .sqlite3.
2. En un fichero llamado “macs.txt” añadiremos las direcciones físicas de los beacons utilizados en la campaña.
3. En los archivos llamados “Config_Ref_Points.conf” y “Config_Ale_Points.conf” pondremos los parámetros de configuración usados durante la campaña.
4. Desde una terminal ejecutar el fichero “Export_Files.py”.