

PROGRAMA ASTURIAS 2016-2017

RESULTADOS DEL PROYECTO:

SISTEMAS AVANZADOS DE POSICIONAMIENTO

REFERENCIA Nº: IDI/2016/000292

Entidad Beneficiaria:



Fecha inicio: 01/01/2016

Fecha fin: 31/03/2017

1. INTRODUCCIÓN

Este proyecto se plantea ante la necesidad de nuevas soluciones para los sistemas de posicionamiento basadas fundamentalmente en la búsqueda de:

- **Combinación de portabilidad y una mayor precisión del posicionamiento.**
- **Posicionamiento en entornos indoor.**

Desde el punto de vista del desarrollo de nuevos sistemas que combinen la portabilidad con un aumento en la precisión del posicionamiento, la mayor dificultad se encuentra la obtención de los datos con sistemas más precisos, lo que hace necesario el desarrollo de nuevos sistemas tanto para la toma de datos previa, como para el post-procesado posterior de los mismos.

Desde Fundación ITMA se ha trabajado en 2 líneas:

1. La captura de los datos del GPS de precisión en formatos L1 y L2 (junto con la posibilidad de incluir otros sensores en función de la aplicación a que vaya destinado el sistema) de forma sincronizada.
2. El desarrollo del post-procesado necesario para que a partir de los datos capturados se obtenga un fichero en un formato amigable, donde todos los datos procedentes de los sensores estén dispuestos de forma sincronizada con los datos de posicionamiento de precisión.

En relación con el desarrollo de nuevos sistemas de posicionamiento *indoor*, se trata de desarrollar nuevas tecnologías que permitan aplicar con seguridad y precisión estas tecnologías en espacios interiores. Se debe tener en cuenta que el desarrollo de sistemas de posicionamiento de precisión y de posicionamiento *indoor* cuenta con innumerables aplicaciones (detección, localización, seguimiento y control de materias primas, productos, vehículos y personas en almacenes, plantas industriales, cadenas de producción y distribución, optimización de actividades de transporte y logística, trazabilidad completa de producto, seguridad, etc.) a un gran número de sectores (logístico, farmacéutico, transporte marítimo y terrestre, industrial, ingeniería, topología y construcción, etc.).

2. OBJETIVOS

El objetivo global del proyecto es el lanzamiento de una nueva línea de investigación en sistemas avanzados de posicionamiento a través de dos vías:

- El desarrollo de sistemas de posicionamiento global portables de precisión.
- El desarrollo de sistemas de posicionamiento en entornos *indoor*.

Para ello, se plantean los siguientes objetivos específicos:

- Desarrollo de un sistema de captura de datos sincronizado para un GPS de precisión.

- Desarrollo de un sistema de post-procesado para convertir los datos brutos obtenidos por el sistema GPS de precisión en datos comprensibles y usables por el usuario.
- Desarrollo de un sistema de post-procesado capaz de proporcionar un único fichero con todos los datos capturados (GPS y sensores) de forma sincronizada incluyendo las soluciones GPS de precisión finales.
- Estudio de tecnologías de posicionamiento *indoor*.
- Desarrollo de sistema demo de posicionamiento *indoor* y obtención de conclusiones prácticas sobre la tecnología.

3. ACTIVIDADES DESARROLLADAS

HITO 1: DISEÑO, DESARROLLO Y VALIDACIÓN SOFTWARE CAPTURA DE DATOS GPS DE PRECISIÓN.

Se ha diseñado y desarrollado un software completo en lenguaje C++, que permite la configuración de un GPS, tanto en modo L1 como en modo L1+L2 (precisión). Además permite modificar diferentes parámetros del propio GPS, como puede ser su velocidad de captura y transmisión de datos.

Del mismo modo, el software, a través de dos hilos diferentes, captura de forma simultánea e independiente el posicionamiento GPS en formato texto y en formato binario, que será el que nos permitirá obtener la mayor precisión posible, a través del post-procesado de los datos.

Además, se han implementado todas las funciones necesarias para realizar tanto el proceso de configuración, como de lectura y grabado de datos a través del protocolo UART (*Universal Asynchronous Receiver-Transmitter*) en el sistema que controla nuestro GPS, que en nuestro caso, para lograr la capacidad de portabilidad se ha optado por una Raspberry Pi, como se muestra en la siguiente figura.



Figura 1. Montaje GPS L1+L2 sobre Raspberry Pi

Todos los protocolos de configuración y comunicación, implementan un sistema de control de errores basado en un algoritmo de petición-respuesta, que garantiza que todos los procesos realizados se han realizado de forma correcta.

Por último, el software incluye la posibilidad de utilizar otros sensores además del propio GPS, como son una IMU o cualquier dispositivo que emita pulsos, de forma simultánea e independiente. Para ello se han implementado además del ya mencionado protocolo UART, los protocolos I2C y SPI, cubriendo así prácticamente todos los protocolos de comunicación que podrían utilizar cualquier GPS, así como cualquier otro dispositivo como el comentado sistema de posicionamiento inercial (IMU).

A continuación se muestra una captura de todos los ficheros que componen el programa final de captura de datos.

Nombre	Fecha de creación	Fecha de modifica...	Tipo	Tamaño
.git	29/03/2016 11:54	29/03/2016 11:54	Carpeta de archivos	
data	27/04/2016 12:57	27/04/2016 12:57	Carpeta de archivos	
info	29/03/2016 11:56	29/03/2016 11:56	Carpeta de archivos	
INTERRUPT	27/04/2016 12:57	27/04/2016 12:57	Carpeta de archivos	
beagle_cape.c	15/01/2016 9:47	22/02/2016 11:46	Archivo C	1 KB
beagle_cape.h	15/01/2016 9:48	22/02/2016 11:46	C/C++ Header	1 KB
binary.c	19/01/2016 13:48	22/02/2016 11:53	Archivo C	3 KB
binary.h	19/01/2016 13:48	22/02/2016 11:46	C/C++ Header	1 KB
BinaryMsg.h	19/01/2016 13:49	22/02/2016 11:46	C/C++ Header	50 KB
led_driver.c	09/02/2016 13:50	22/02/2016 11:46	Archivo C	6 KB
led_driver.h	09/02/2016 13:50	22/02/2016 11:46	C/C++ Header	1 KB
main.c	28/01/2016 13:50	22/02/2016 11:58	Archivo C	2 KB
Makefile	18/02/2016 13:51	22/02/2016 11:46	Archivo	1 KB
navigation.c	22/02/2016 10:51	26/04/2016 16:26	Archivo C	7 KB
navigation.h	26/02/2016 14:52	22/02/2016 11:46	C/C++ Header	1 KB
prueba.c	08/03/2016 9:52	26/04/2016 16:10	Archivo C	1 KB
serial.c	17/03/2016 11:53	22/02/2016 11:53	Archivo C	2 KB
serial.h	17/03/2016 11:53	22/02/2016 11:46	C/C++ Header	1 KB
switch_driver.c	13/04/2016 9:56	22/02/2016 10:46	Archivo C	5 KB
switch_driver.h	13/04/2016 9:57	22/02/2016 10:46	C/C++ Header	1 KB

Figura 2. Firmware Captura Datos

En esta captura podemos observar los ficheros encargados de la captura del GPS en formato texto, navigation.c y navigation.h, que incluye además la posibilidad de incluir otro tipo de sensores como ya se comentó anteriormente y la captura del GPS en formato binario, binary.c y binary.h.

Por otro lado, se encuentran los ficheros de control de transmisión de datos a través del puerto serie, serial.c y serial.h, junto con los ficheros que controlan los leds de indicadores, led_driver.c y led_driver.h, junto con el que controla los botones utilizados para comenzar la captura de datos, la parada y la transmisión de estos a un lápiz usb, switch_driver.c y switch_driver.h.

Durante el desarrollo de todo este software se ha realizado de forma periódica, la validación del software, mediante la realización de pruebas de campo con un GPS de precisión montado en una Raspberri Pi incorporando nuestro software de captura.

Las pruebas llevadas a cabo, han consistido en realizar varias veces el mismo recorrido para obtener los ficheros de captura. Una vez obtenidos dichos ficheros, se ha realizado una fase de post-procesado de los mismos con el objetivo de comprobar que todos los valores capturados y guardados son coherentes tanto en tiempo, como en posicionamiento, así como que los resultados son repetitivos y, por tanto, fiables.

Este proceso consiste, en primer lugar, en comprobar que todos los mensajes poseen una marca temporal que coincide con el momento en el que se realizaron las capturas, así como que todos los

mensajes han sido escritos de forma correcta, sin ningún error de lectura por parte de nuestro software. Estos errores, que sucedieron durante la depuración del mismo, consistían en la captura de mensajes incompletos, la lectura de caracteres erróneos o la inclusión de saltos de línea en medio de un mensaje de GPS.

La segunda parte del proceso de validación, se encargaría de comprobar que las posiciones GPS coinciden con las posiciones reales donde se realizaron las pruebas. Este proceso fue tedioso, puesto que debimos comprobar mediante alguno de los softwares webs existentes de localización de posiciones GPS, que cada una de las posiciones GPS capturadas correspondían con posiciones coherentes en el mapa global.

A continuación mostramos una imagen para ejemplificar el proceso que se hizo con cada uno de los puntos GPS capturados por nuestro sistema.

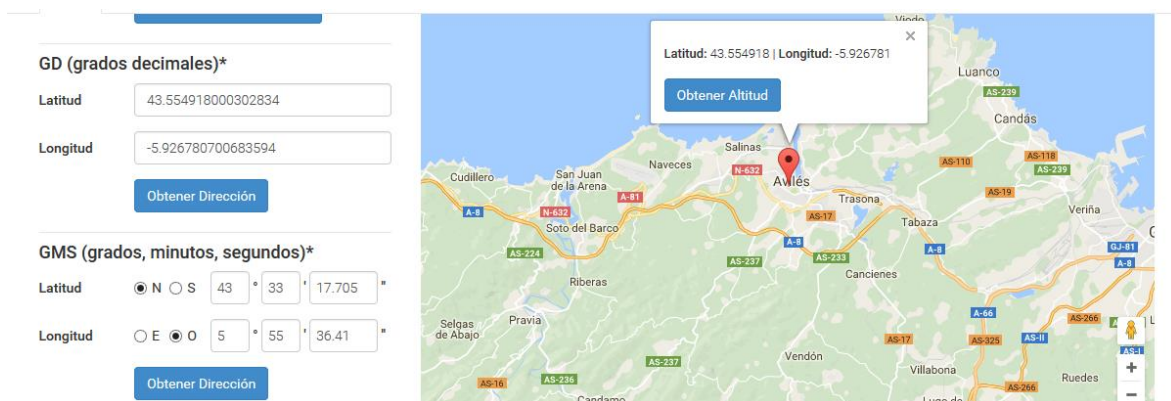


Figura 3. Validación captura GPS vs posicionamiento real

Seguidamente mostramos un ejemplo de parte de un fichero con la captura del GPS en formato texto, junto con los valores de una IMU que se añadió al sistema.

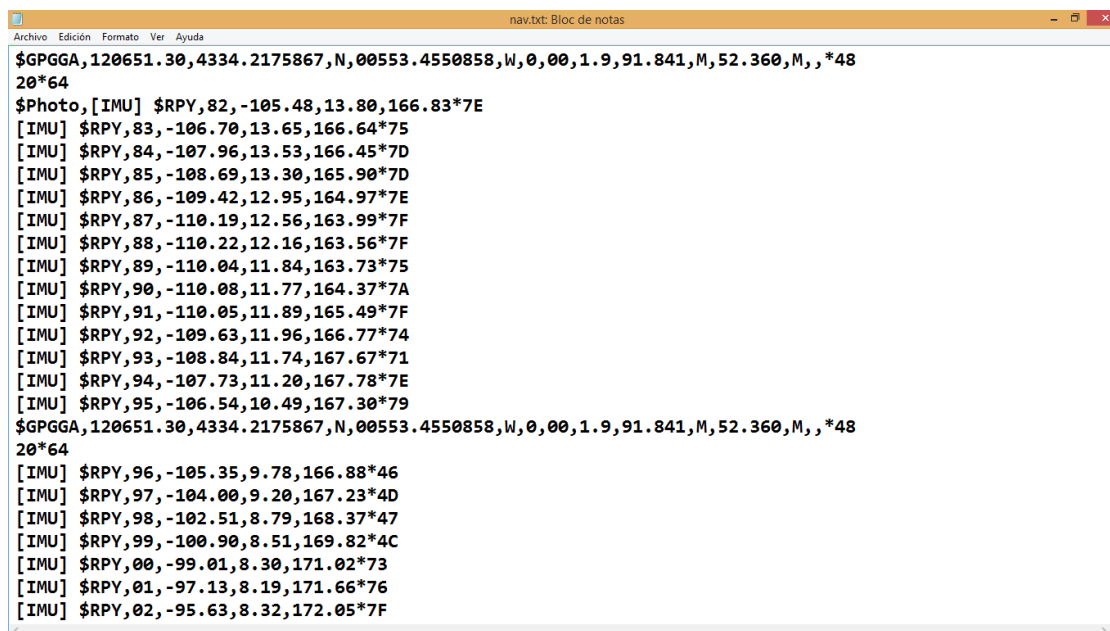


Figura 4. Ejemplo captura GPS formato texto

El fichero en formato binario, no se muestra puesto que aparecerían textos ininteligibles, que no aportan ningún valor gráfico, pese a que en el post-procesado de los datos, nos permitirán conseguir el aumento de precisión que buscábamos.

HITO 2: DISEÑO, DESARROLLO Y VALIDACIÓN SOFTWARE CONVERSIÓN DATOS BINARIOS A FORMATO LEGIBLE.

Se ha desarrollado un software en lenguaje Python, con interfaz gráfica, que utilizando los ficheros de texto y binario que se obtienen con nuestro sistema de captura de datos, es capaz de convertir de forma automática los datos binarios, que contienen magnitudes basadas en las medidas de una serie de constelaciones de satélites, y que no son transparentes ni legibles directamente por el usuario a un formato legible, manteniendo la precisión de los mismos.

El formato elegido para tal efecto, como ya se comentó, es RINEX, por tratarse de un formato estandarizado que permite la gestión y almacenamiento de las medidas generadas por nuestro receptor GPS, así como su post-procesado off-line.

A continuación se muestra el aspecto de la pantalla de nuestro software, a partir de la cual se introducen los datos de partida en el GPS:

Elevation Mask (deg) [0-90] Cinematic Time (sec) [0-500]

TXT FILE
 ...

BIN FILE
 ...

RINEX DIRECTORY
 ...

RINEX NAME

Frequency
 Hz.

0%

Figura 5. Software Conversión datos binarios a formato legible

Podemos observar, que en nuestra aplicación existen diferentes opciones que son totalmente configurables, que van desde la frecuencia de muestreo del GPS, pasando por el ángulo de elevación que queremos tener en cuenta para considerar que satélites son aceptables a la hora de capturar posiciones o el tiempo que consideramos para pasar a un estado de captura de datos GPS en modo cinemático y no estático.

Ahora se muestra la carpeta que contiene todos los ficheros con el código fuente de nuestro software de conversión de datos binarios a formato legible.

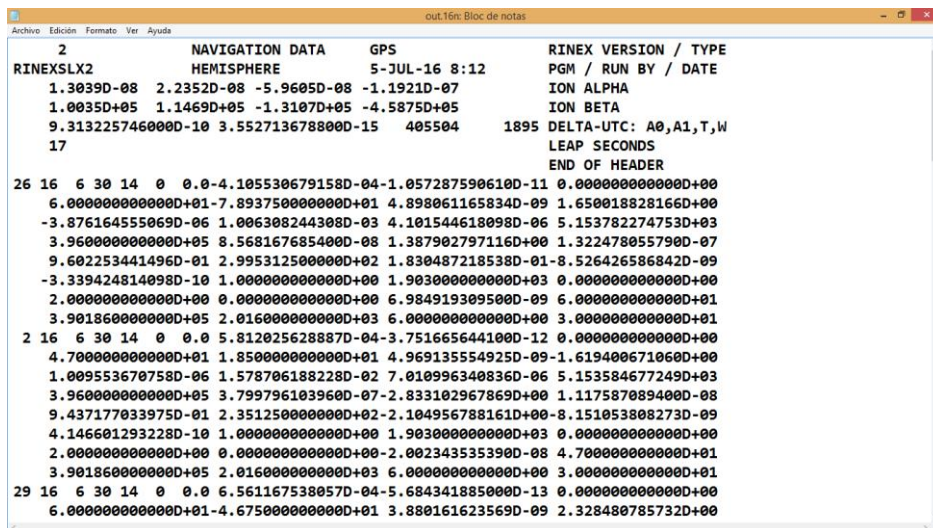
Nombre	Fecha de creación	Fecha de modifica...	Tipo	Tamaño
DLLs	18/05/2016 14:46	18/05/2016 14:46	Carpeta de archivos	
cinematic.ui	09/03/2016 10:24	18/04/2016 17:55	Archivo UI	10 KB
cinematic_interface.py	17/03/2016 12:28	05/05/2016 12:46	Archivo PY	20 KB
config.py	13/04/2016 14:45	18/04/2016 16:26	Archivo PY	1 KB
setup.py	02/05/2016 11:45	16/08/2016 11:00	Archivo PY	1 KB
RinexSLX2.exe	07/06/2016 10:46	18/07/2014 15:15	Aplicación	163 KB

Figura 6. Fichero Formato Legible

El fichero “cinematic.ui” es el fichero gráfico donde se diseñó la parte estética de la aplicación. El fichero “cinematic_interface.py” es el programa principal de Python, donde se encuentran todas las funciones que se encarga de realizar la conversión de datos a formato legible, apoyándose en el ejecutable RinexSLX2.exe. Además, en este fichero se relacionan cada una de las funciones con cada uno de los complementos de la interfaz gráfica que el usuario puede modificar.

Del mismo modo que en el caso anterior, se ha realizado un proceso de validación de este software.

Utilizando los ficheros obtenidos con el software de captura, tanto el fichero en formato texto como el fichero en formato binario, estos se adjuntan a nuestro software de post-procesado obteniéndose un fichero legible en formato Rinex, que se muestra a continuación.



```

2          NAVIGATION DATA      GPS          RINEX VERSION / TYPE
RINEXSLX2  HEMISPHERE          5-JUL-16 8:12  PGM / RUN BY / DATE
1.3039D-08 2.2352D-08 -5.9605D-08 -1.1921D-07  ION ALPHA
1.0035D+05 1.1469D+05 -1.3107D+05 -4.5875D+05  ION BETA
9.313225746000D-10 3.552713678800D-15 405504 1895 DELTA-UTC: A0,A1,T,W
17 LEAP SECONDS
END OF HEADER

26 16 6 30 14 0 0.0 -4.105530679158D-04 -1.057287590610D-11 0.000000000000D+00
6.000000000000D+01 -7.893750000000D+01 4.898061165834D-09 1.650018828166D+00
-3.876164555069D-06 1.006308244308D-03 4.101544618098D-06 5.153782274753D+03
3.960000000000D+05 8.568167685400D-08 1.387902797116D+00 1.322478055790D-07
9.602253441496D-01 2.995312500000D+02 1.830487218538D-01 -8.526426586842D-09
-3.339424814098D-10 1.000000000000D+00 1.903000000000D+03 0.000000000000D+00
2.000000000000D+00 0.000000000000D+00 6.984919309500D-09 6.000000000000D+01
3.901860000000D+05 2.016000000000D+03 6.000000000000D+00 3.000000000000D+01
2 16 6 30 14 0 0.0 5.812025628887D-04 -3.751665644100D-12 0.000000000000D+00
4.700000000000D+01 1.850000000000D+01 4.969135554925D-09 -1.619400671060D+00
1.009553670758D-06 1.578706188228D-02 7.010996340836D-06 5.153584677249D+03
3.960000000000D+05 3.799796103960D-07 -2.833102967869D+00 1.117587089400D-08
9.437177033975D-01 2.351250000000D+02 -2.104956788161D+00 -8.151053808273D-09
4.146601293228D-10 1.000000000000D+00 1.903000000000D+03 0.000000000000D+00
2.000000000000D+00 0.000000000000D+00 -2.002343535390D-08 4.700000000000D+01
3.901860000000D+05 2.016000000000D+03 6.000000000000D+00 3.000000000000D+01
29 16 6 30 14 0 0.0 6.561167538057D-04 -5.684341885000D-13 0.000000000000D+00
6.000000000000D+01 -4.675000000000D+01 3.880161623569D-09 2.328480785732D+00
  
```

Figura 7. Fichero Formato Legible

En este fichero, ya se obtiene una solución de precisión, pero que no es fácil de manejar, puesto que obtenemos datos precisos para cada uno de los satélites que en ese momento de la captura han sido detectados.

El proceso de validación, consistió en comparar que las marcas temporales de nuestro fichero de salida, coincidía con las marcas temporales de nuestro fichero de texto a la entrada del software y que no existían incoherencias entre ambos ficheros, como se procedió en el caso anterior.

Para ello se compararon los datos obtenidos al realizar el mismo recorrido al menos dos veces, de forma que se observara que los resultados eran repetitivos y coherentes con la configuración de los parámetros pertinentes de la propia aplicación.

HITO 3: DISEÑO, DESARROLLO Y VALIDACIÓN SOFTWARE SINCRONIZACIÓN.

Se ha diseñado e implementado un software de sincronización, utilizando nuevamente lenguaje Python y con una interfaz gráfica, que utilizando los ficheros obtenidos en el software anterior, nos proporciona como resultado final un fichero ordenado y coherente, en el que aparezcan de forma sincronizada cada uno de los datos capturados, GPS de precisión y sensores.

Como se planteaba se ha desarrollado un algoritmo propio para que los tiempos y posiciones que ofrecen los sensores incluidos al sistema, excepto en el caso del GPS; sean reales y fiables. Además de diseñar un algoritmo que “predice” posiciones de precisión en puntos temporales en los que no se dispone de valores reales de GPS.

A continuación, mostramos una imagen de la interfaz de nuestro software:

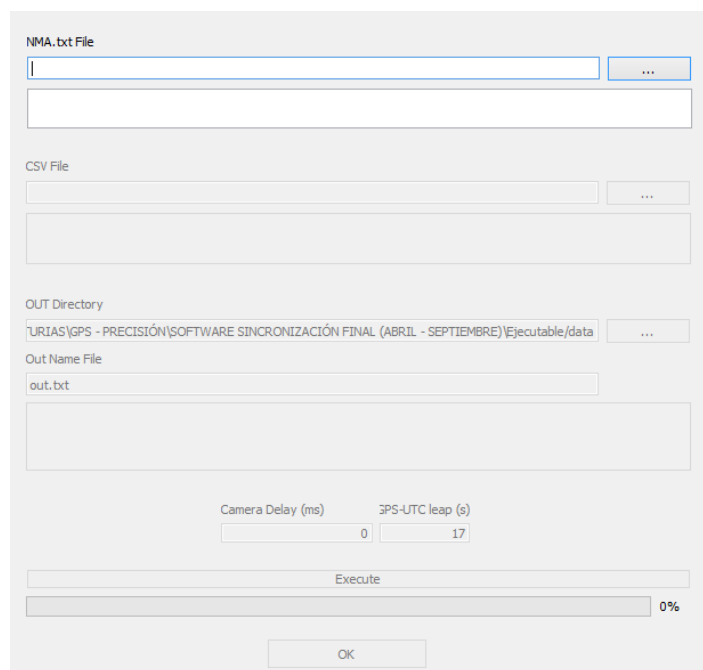


Figura 8. Software Sincronización

Ahora, mostramos todos los ficheros que conforman el código fuente de nuestra aplicación:

Nombre	Fecha de creación	Fecha de modifica...	Tipo	Tamaño
DLLs	16/06/2016 11:01	16/06/2016 11:02	Carpeta de archivos	
crc.py	10/05/2016 13:00	29/04/2016 9:20	Archivo PY	3 KB
deo.ui	22/04/2016 10:59	07/06/2016 12:39	Archivo UI	8 KB
deo_interface.py	20/05/2016 11:00	16/08/2016 12:27	Archivo PY	46 KB
setup.py	08/06/2016 9:01	12/07/2016 14:23	Archivo PY	1 KB

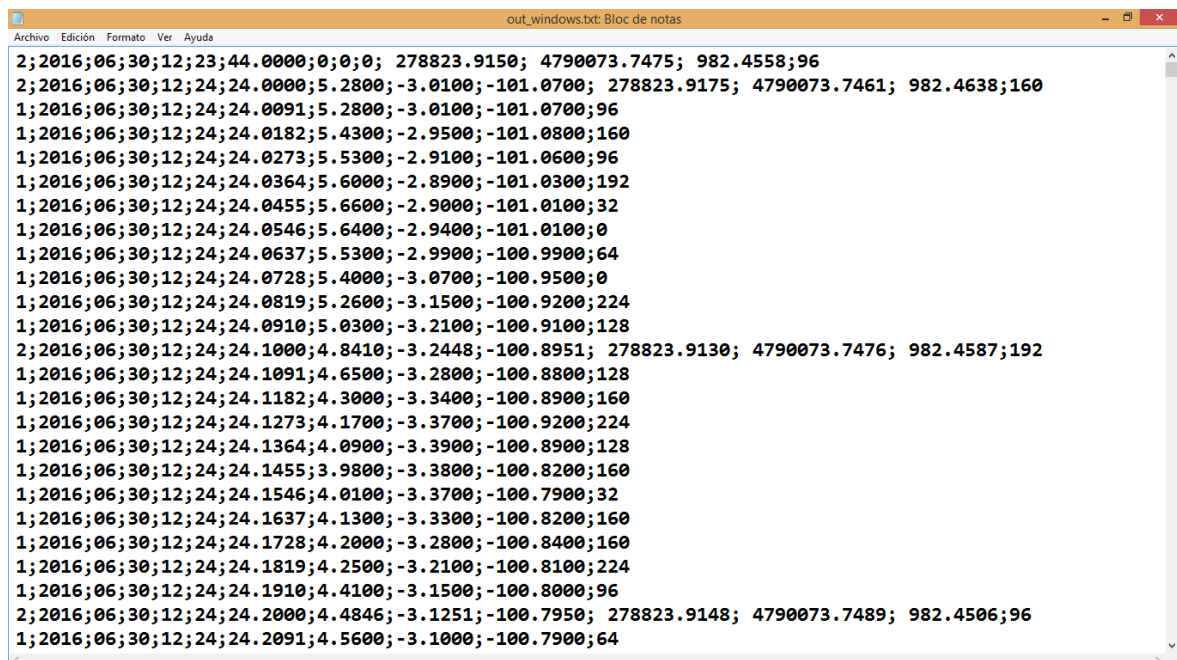
Figura 9. Software Sincronización

El fichero “deo.ui” es el fichero gráfico donde se desarrolló la parte estética de la aplicación. El fichero “deo_interface.py” es el programa principal de Python, donde está implementado todo el algoritmo necesario para realizar la sincronización entre todos los datos obtenidos, tanto del GPS de precisión, como del resto de dispositivos que pueden incluirse. Este proceso se encarga de sincronizar temporalmente todos los datos capturados, mostrándolos de forma ordenada y sencilla.

El fichero “crc.py”, es un fichero que se encarga de generar un código de comprobación de errores para cada uno de los mensajes obtenidos, de forma que podamos detectar cuando ocurre alguna incidencia en el proceso.

Del mismo modo que el resto de sistemas, ha sido necesario realizar un proceso de validación. Utilizando los datos que se obtuvieron a lo largo de los procesos anteriores, se aplicaba sobre estos el algoritmo que ha sido implementado en nuestra aplicación.

A continuación mostramos un ejemplo del fichero final que se obtiene tras aplicar nuestro software de sincronización:



```

out_windows.txt: Bloc de notas
Archivo Edición Formato Ver Ayuda
2;2016;06;30;12;23;44.0000;0;0;0; 278823.9150; 4790073.7475; 982.4558;96
2;2016;06;30;12;24;24.0000;5.2800;-3.0100;-101.0700; 278823.9175; 4790073.7461; 982.4638;160
1;2016;06;30;12;24;24.0091;5.2800;-3.0100;-101.0700;96
1;2016;06;30;12;24;24.0182;5.4300;-2.9500;-101.0800;160
1;2016;06;30;12;24;24.0273;5.5300;-2.9100;-101.0600;96
1;2016;06;30;12;24;24.0364;5.6000;-2.8900;-101.0300;192
1;2016;06;30;12;24;24.0455;5.6600;-2.9000;-101.0100;32
1;2016;06;30;12;24;24.0546;5.6400;-2.9400;-101.0100;0
1;2016;06;30;12;24;24.0637;5.5300;-2.9900;-100.9900;64
1;2016;06;30;12;24;24.0728;5.4000;-3.0700;-100.9500;0
1;2016;06;30;12;24;24.0819;5.2600;-3.1500;-100.9200;224
1;2016;06;30;12;24;24.0910;5.0300;-3.2100;-100.9100;128
2;2016;06;30;12;24;24.1000;4.8410;-3.2448;-100.8951; 278823.9130; 4790073.7476; 982.4587;192
1;2016;06;30;12;24;24.1091;4.6500;-3.2800;-100.8800;128
1;2016;06;30;12;24;24.1182;4.3000;-3.3400;-100.8900;160
1;2016;06;30;12;24;24.1273;4.1700;-3.3700;-100.9200;224
1;2016;06;30;12;24;24.1364;4.0900;-3.3900;-100.8900;128
1;2016;06;30;12;24;24.1455;3.9800;-3.3800;-100.8200;160
1;2016;06;30;12;24;24.1546;4.0100;-3.3700;-100.7900;32
1;2016;06;30;12;24;24.1637;4.1300;-3.3300;-100.8200;160
1;2016;06;30;12;24;24.1728;4.2000;-3.2800;-100.8400;160
1;2016;06;30;12;24;24.1819;4.2500;-3.2100;-100.8100;224
1;2016;06;30;12;24;24.1910;4.4100;-3.1500;-100.8000;96
2;2016;06;30;12;24;24.2000;4.4846;-3.1251;-100.7950; 278823.9148; 4790073.7489; 982.4506;96
1;2016;06;30;12;24;24.2091;4.5600;-3.1000;-100.7900;64

```

Figura 10. Ejemplo documento salida software sincronización

En el fichero, obtenemos las diferentes líneas de datos obtenidas. En este caso, se ha utilizado un GPS (2) junto con una IMU (1). Cada una de las líneas corresponde a una captura de uno de ellos, siendo el primer dato el identificador que nos indica a que dispositivo pertenece.

En este caso los valores de GPS, incluyen posicionamiento y tiempo, siendo los valores de IMU para ese punto interpolados; por el contrario, la IMU únicamente posee valores de ángulos reales, siendo los valores de tiempo y GPS predichos, en función de los GPS existentes.

El proceso de validación consistió por tanto en comprobar que toda la información que se generaba en este fichero, tanto posiciones reales, posiciones interpoladas, tiempos reales y tiempos interpolados, se encontraban ordenados temporalmente, que existía una conexión real y correcta entre las posiciones GPS y su marca temporal y que los datos interpolados eran coherentes y consistentes, en comparación con los valores reales no interpolados.

HITO 4: GPS INDOOR

Durante la anualidad 2016 se llevó a cabo un estudio de las tecnologías existentes en el mercado en el ámbito de la localización y posicionamiento en interiores. Se analizaron las características de distintas soluciones, tanto tecnologías inalámbricas (WiFi, BLE, RFID, NFC, UWB...) como tecnologías que no utilizan radiofrecuencia (sistemas inerciales, magnéticos, basados en marcadores visuales, etc.).

De entre todas estas posibilidades se optó por una solución basada en técnicas de radiofrecuencia de banda ultra ancha o *Ultrawideband* (UWB). Dicha tecnología se usa habitualmente para la transferencia de información empleando un ancho de banda muy grande (superior a los 500 MHz), sin interferir en las transmisiones convencionales de banda estrecha en su mismo rango de frecuencias. Permite, de esta forma, enviar paquetes de datos muy grandes a altas velocidades en un corto alcance.

La implantación de UWB en sistemas de localización de tiempo real o *Real-Time Location Systems* (RTLS) es cada vez mayor debido a su elevada precisión (menor que 30 cm) en rangos grandes y su relativo bajo consumo. Al tratarse de sistemas de baja potencia, se consideran particularmente apropiados en entornos sensibles a las ondas de radiofrecuencia, como los hospitales.

Al contrario que el resto de sistemas de radiofrecuencia, basados en tecnologías de onda portadora, UWB presenta una gran inmunidad frente al llamado *multipath fading* o desvanecimiento por trayectorias múltiples. Cada uno de sus pulsos ocupa todo el ancho de banda dedicado a la transmisión, lo que significa que son pulsos estrechos en el dominio temporal, por lo que la mayoría de las reflexiones de la señal enviada no se solapa con el pulso original. Esta característica lo hace especialmente atractivo en sistemas de posicionamiento en interiores, donde los dispositivos han de mantener una alta fiabilidad en presencia de múltiples obstáculos.

El sistema seleccionado se basa en la técnica de tiempo de vuelo o *Time of Flight, Time of Arrival* (ToA). Como su propio nombre indica, consiste en determinar la distancia entre transmisor y receptor a partir del tiempo que tarda la señal en propagarse de un punto al otro. Cuando la sincronización entre las dos unidades de radiotransmisión no es posible, se aplica un método denominado telemetría bidireccional, de dos vías o *Two-Way Ranging* (TWR). En este caso, se considera la medida del viaje de ida y de vuelta al completo, compensando adecuadamente las diferencias de fase entre cada reloj.

Una vez seleccionada la tecnología *Ultrawideband* como la más prometedora de todas las que existen hoy en día en el mercado, durante la anualidad 2017 se ha procedido a realizar una prueba de concepto con la que llevar a cabo una evaluación básica del sistema y de su rendimiento dentro de un recinto cerrado.

El objetivo de estas primeras pruebas fue verificar las características de la tecnología y comprobar su alcance práctico para distintas aplicaciones dentro de un entorno real y controlado. El kit de posicionamiento adquirido para la realización de estas pruebas posee un software propietario que se puede configurar para su empleo en distintas aplicaciones. Como el objetivo era la evaluación básica de la tecnología, se partió de este software para testar las distintas configuraciones y comprobar su versatilidad y alcance práctico.

Las pruebas fueron realizadas en un entorno controlado, es decir en una misma sala de unos 5,5 x 6 metros y 3 metros de altura aproximadamente, que se tomó como localización tipo. Como se ha comentado, se ha empleado el kit de posicionamiento adquirido en la anualidad 2016. Aunque también estaba previsto la adquisición de un kit adicional para ampliar el número de etiquetas a monitorizar, finalmente se ha considerado suficiente con el ya adquirido.

Se han evaluado tres topologías diferentes del sistema, siendo cada una representativa de diversos casos de uso:

1. Seguimiento o localización

Con esta topología se pretende realizar el seguimiento de un elemento dentro de una zona delimitada. Para ello es necesario utilizar 3 dispositivos UWB (anclas), situados a la misma altura y preferiblemente más elevados que el dispositivo que se quiere localizar. Cada una de las anclas calcula su distancia hasta la etiqueta UWB asociada al elemento bajo seguimiento. Mediante un método de triangulación a partir de esas distancias, se obtiene de forma precisa la posición de dicha etiqueta.

Se han realizado distintas pruebas montando las 3 anclas en trípodes a unos 1,6 metros de altura y formando un triángulo rectángulo isósceles, tal y como se puede ver en la Figura 11. La distancia fijada entre el ancla central (A0) y cada una de las anclas secundarias (A1 y A2) fue de 4 metros. El software se ejecuta en un ordenador conectado al ancla central.

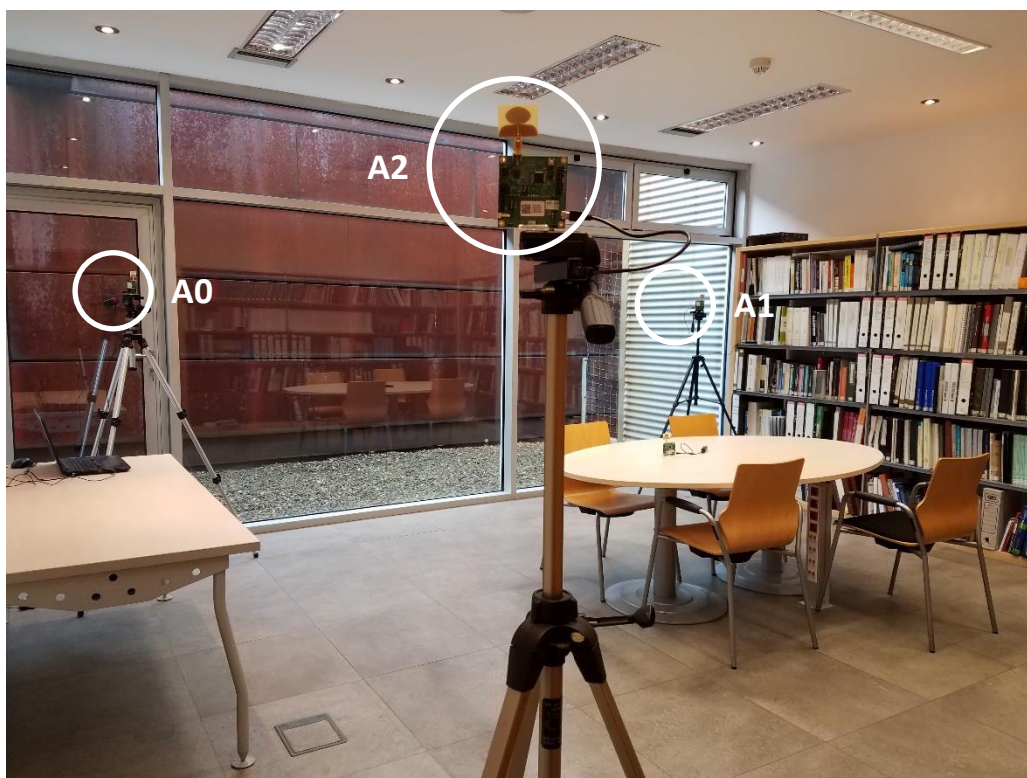


Figura 11. Montaje de las tres anclas (A0, A1 y A2) en el Escenario 1 (Seguimiento o localización)

Tal y como se puede comprobar a modo de ejemplo en la Figura 12 y en la Figura 14, la etiqueta se situó en diversos puntos dentro de la sala, siendo localizada a través del software (Figura 13 y Figura 15 respectivamente).

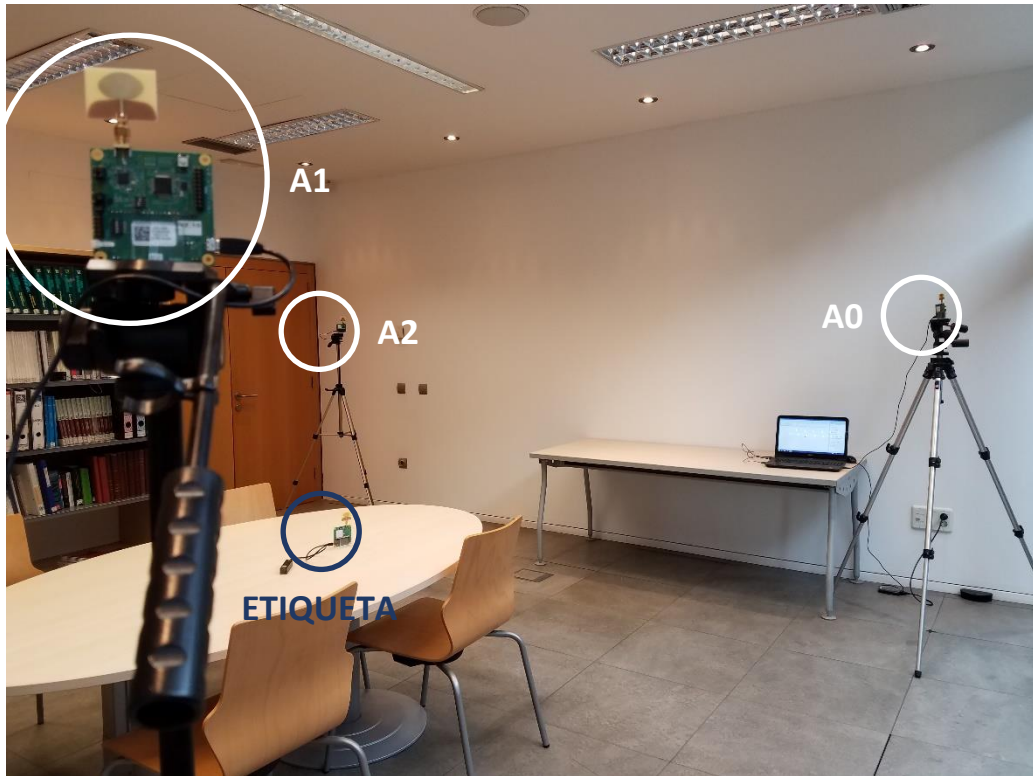


Figura 12. Localización 1 de la etiqueta en el escenario 1 (Seguimiento o localización)

Anchor ID	X (m)	Y (m)	Z (m)
<input checked="" type="checkbox"/> 0	0.00	0.00	1.60
<input checked="" type="checkbox"/> 1	0.00	4.00	1.60
<input checked="" type="checkbox"/> 2	-4.00	0.00	1.60
<input type="checkbox"/> 3	5.00	5.00	3.00

Tag ID/Label	X (m)	Y (m)	Z (m)	R95 (m)	Anc 0 range (m)	Anc 1 range (m)	Anc 2 range (m)	Anc 3 range (m)
<input type="checkbox"/> Tag 0	1.645	2.370	0.631	<input type="checkbox"/> 0.051	3.090	2.543	3.465	

ETIQUETA

Settings: Tracking/Navigation Mode, Geo-Fencing Mode, Zone 1 (m) 2.00, Zone 2 (m) 4.00, Alarm Outside, Alarm Inside, Use Auto-Positioning, Filtering options: None, Logging disabled, Start, Show Tag History 20, Show Tag Table, Show Anchor Table, Show Anchor-Tag Correction Table

Figura 13. Posición relativa de la etiqueta - Localización 1 en el Escenario 1 (Seguimiento o localización)

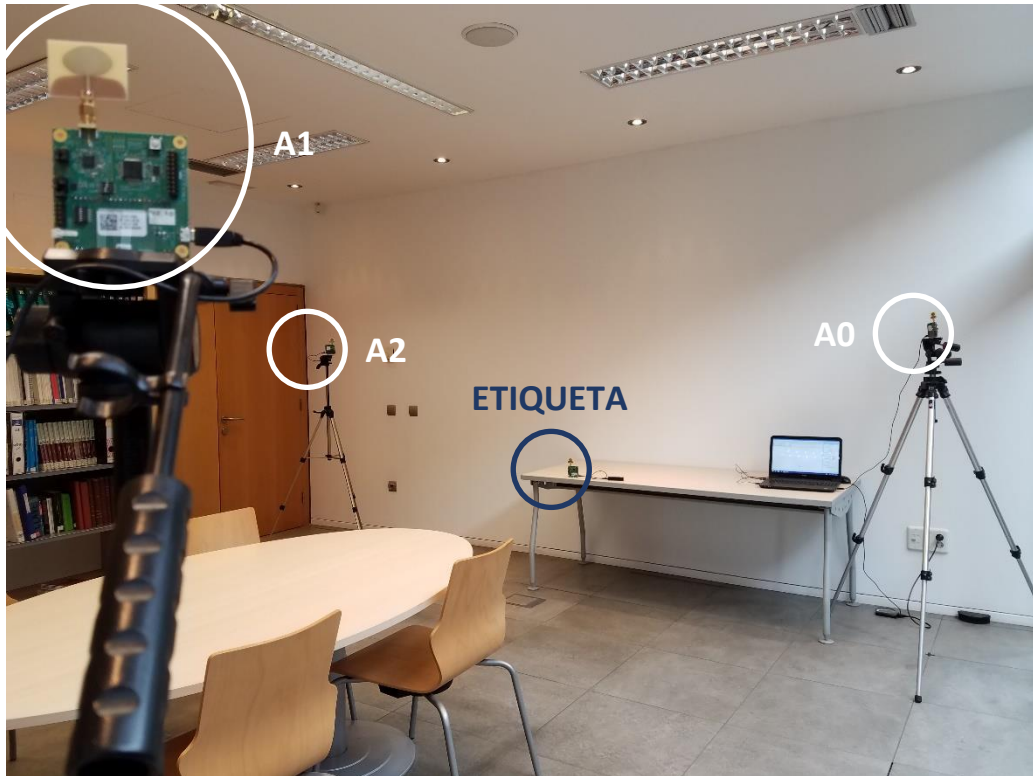


Figura 14. Localización 2 de la etiqueta en el escenario 1 (Seguimiento o localización)

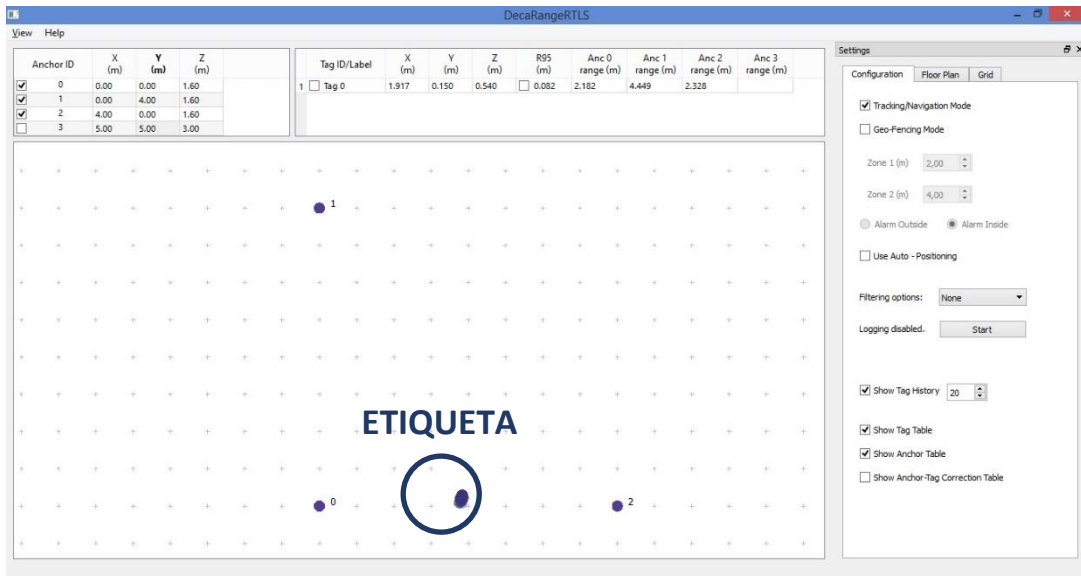


Figura 15. Posición relativa de la etiqueta - Localización 2 en el Escenario 1 (Seguimiento o localización)

Se trata de una localización en tiempo real, pudiendo realizarse un seguimiento de la última trayectoria efectuada por la etiqueta móvil, como se puede ver en el ejemplo indicado en la Figura 16.

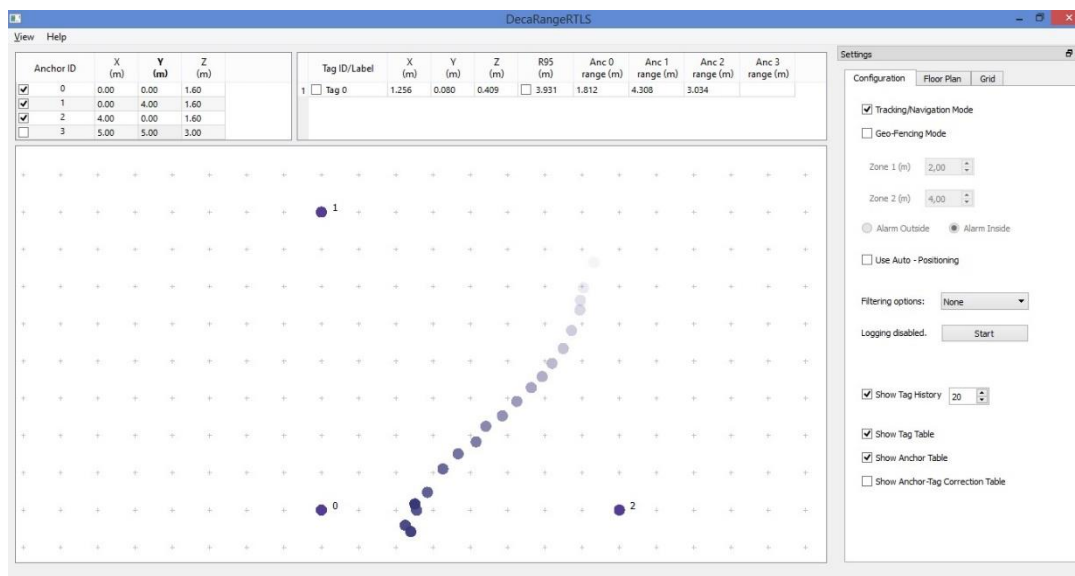


Figura 16. Trayectoria ejemplo de la etiqueta en el Escenario 1

Esta topología está pensada para desarrollar sistemas de rastreo de activos, logística y automatización. Aplicando la configuración descrita sería posible, por ejemplo, llevar a cabo la detección, localización, seguimiento y control de materias primas, productos, vehículos o personas dentro de almacenes, plantas industriales, cadenas de producción o distribución. También se podría optimizar notablemente las actividades de transporte y logística, así como la trazabilidad completa de un producto en cualquier recinto cerrado.

2. Cercado virtual o geo-fencing

En este caso se trata de establecer una serie de perímetros predefinidos alrededor de un dispositivo UWB (ancla), con el fin de definir la posición relativa de los elementos de su entorno. El ancla puede calcular su distancia hasta cada una de las etiquetas situadas en sus proximidades, a partir de la cual se puede determinar si dichas etiquetas se encuentran dentro o fuera de las zonas delimitadas.

Se han realizado varias configuraciones en las que se ha fijado el ancla en una posición central de la sala, a unos 1,7 metros de altura (Figura 17). Se establecieron dos perímetros concéntricos con ella, de 1,5 y 3 metros respectivamente. Existe la posibilidad de definir un aviso, tanto si se detecta una etiqueta fuera del cercado exterior como si se detecta dentro del cercado interior. En este caso, se definió la alarma para posiciones externas.



Figura 17. Montaje del ancla en el Escenario 2 (Cercado virtual)

Con el fin de analizar todas las situaciones posibles, se situó una etiqueta en cada una de las 3 zonas delimitadas por esos perímetros (Figura 18), siendo correctamente detectadas por el software, como se puede comprobar en la Figura 19.

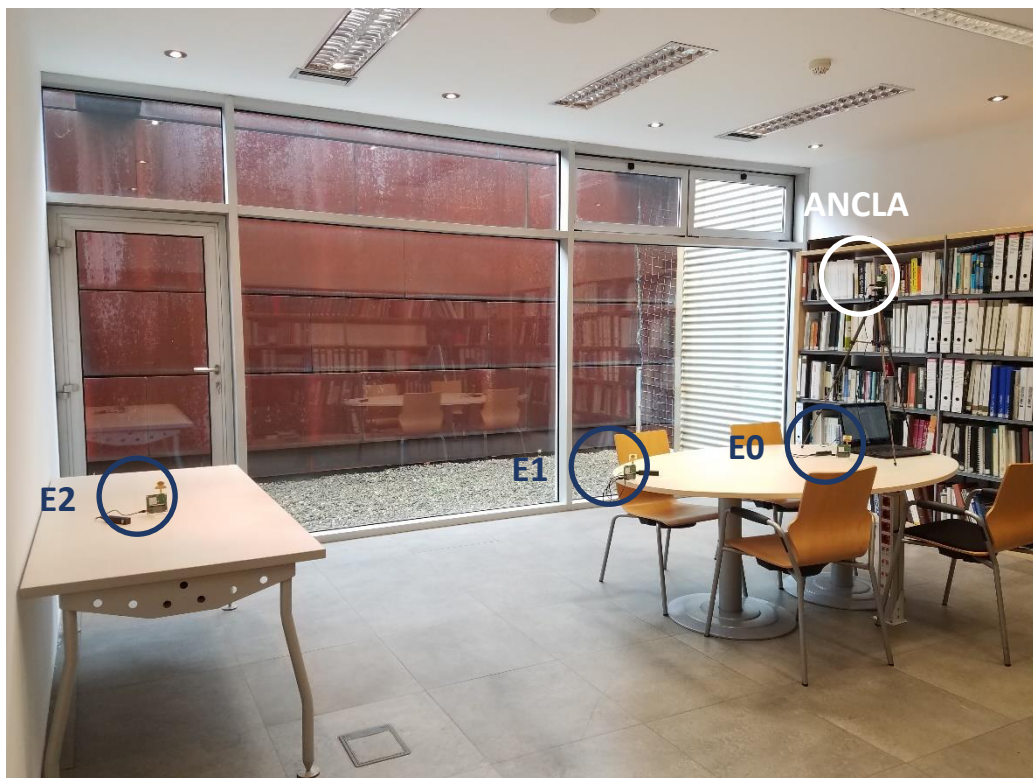


Figura 18. Localización de las etiquetas (E0, E1 y E2) en el escenario 2 (Cercado virtual)

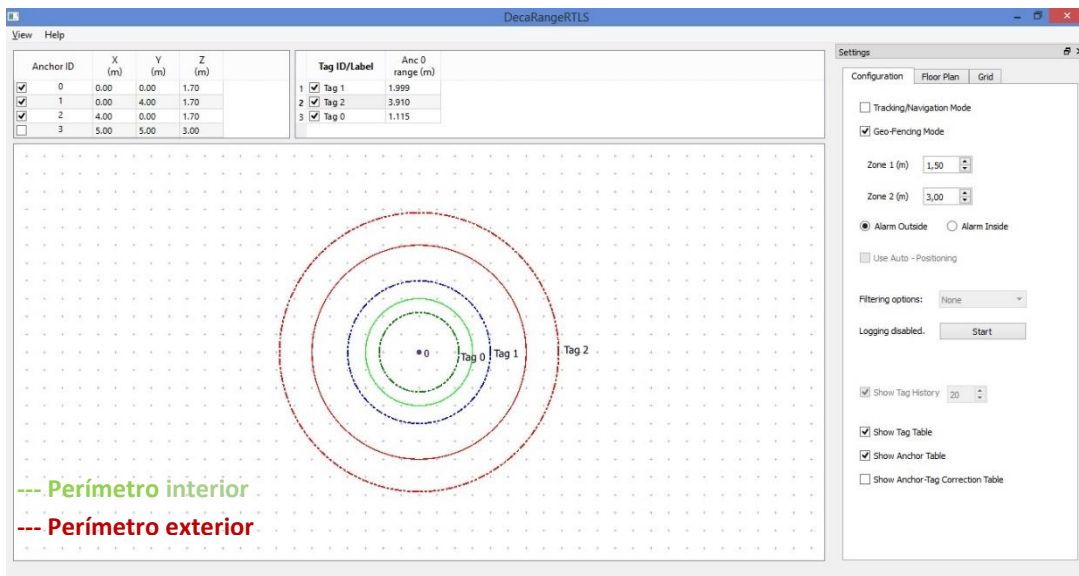


Figura 19. Distancia relativa a la que se encuentra la etiqueta 0 (verde oscuro), la etiqueta 1 (azul) y la etiqueta 2 (rojo-alarma) - Localización 1 en el Escenario 2 (Cercado virtual)

Posteriormente se modificó la posición de las etiquetas entrando y saliendo de las áreas de detección. Como se puede ver en la Figura 20, estas variaciones son monitorizadas en tiempo real.

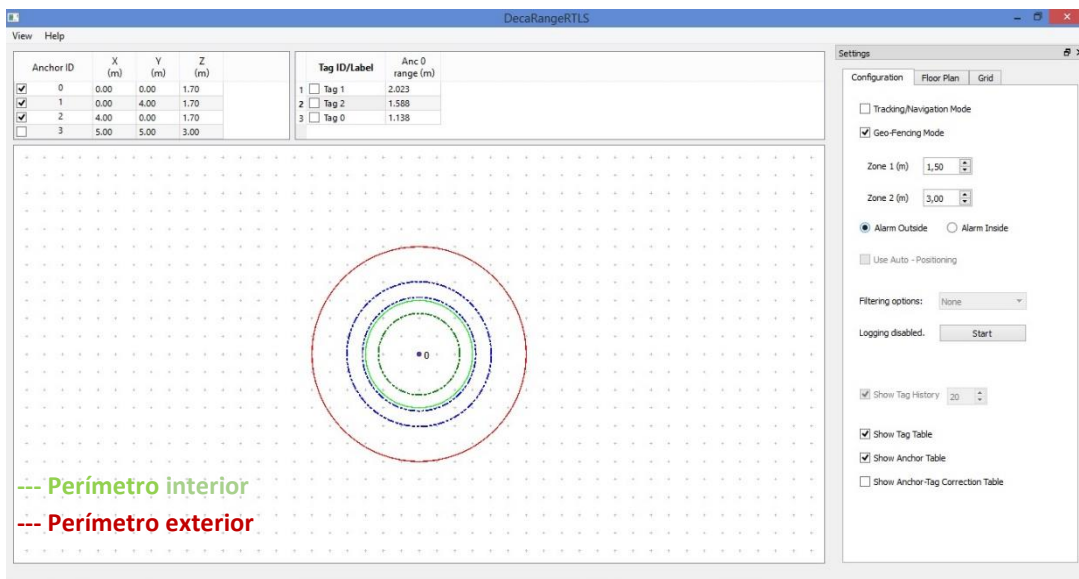


Figura 20. Distancia relativa a la que se encuentra la etiqueta 0 (verde oscuro) y la etiqueta 1 y 2 (azul) - Localización 2 en el Escenario 2 (Cercado virtual)

Este tipo de escenario se puede implementar en sistemas de seguridad y protección, geo-cercas y domótica. Empleando esta topología es posible determinar de forma precisa perímetros de seguridad entorno a maquinaria, prevenir colisiones de elementos móviles, permitir o denegar el paso a estancias que exijan dispositivos de seguridad adicionales o monitorizar la posición de menores, entre otras muchas aplicaciones.

3. Navegación

Esta topología permite la navegación de un elemento dentro de una zona delimitada. Al igual que en el primer escenario, es necesario posicionar estratégicamente al menos 3 dispositivos UWB (anclas) dentro del área de navegación. Es de nuevo el cálculo de las distancias hasta cada una de las anclas lo que posibilita al elemento conocer su posición relativa mediante triangulación.

La configuración práctica es la misma que en la topología de localización. La única diferencia es que, para este caso, el ordenador se conectó a la etiqueta móvil en lugar de conectarlo a una de las anclas, como se puede comprobar en la Figura 21.

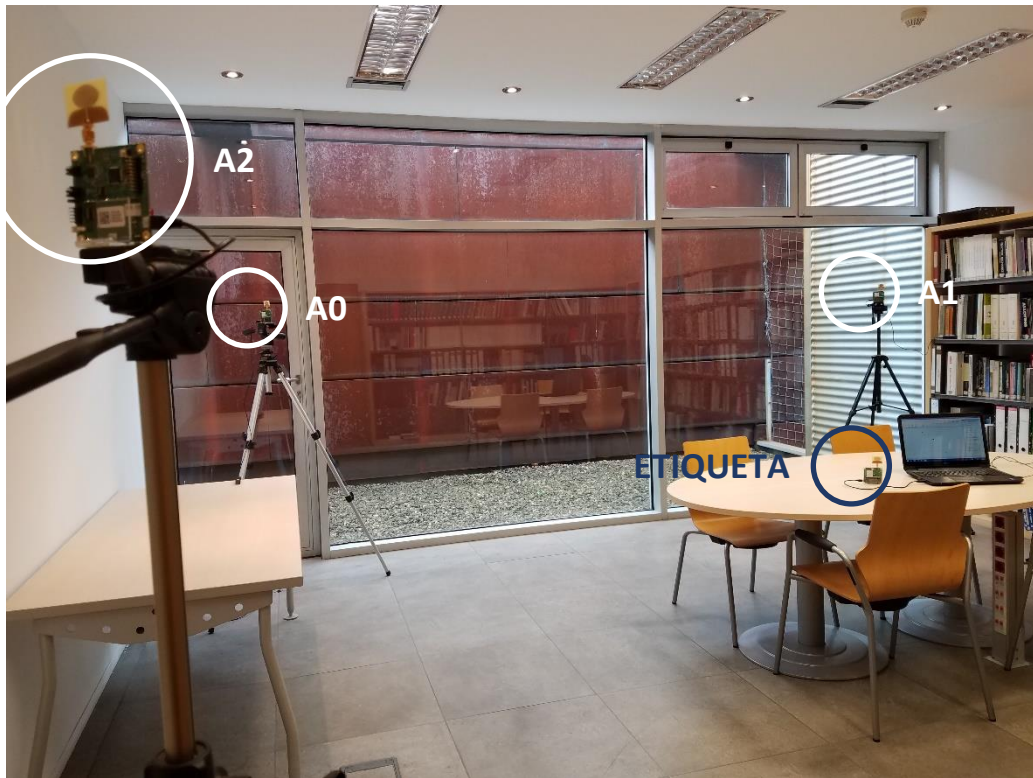


Figura 21. Montaje de las tres anclas (A0, A1 y A2) y de la etiqueta en el Escenario 3 (Navegación)

Con esta configuración se pudo determinar con precisión la posición en tiempo real de la etiqueta dentro de la sala (Figura 22).

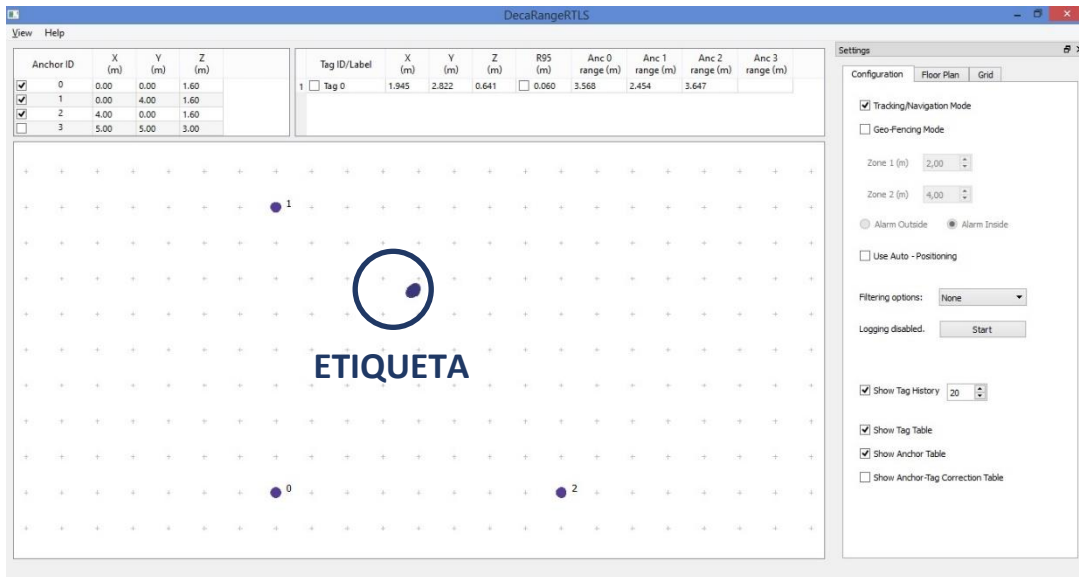


Figura 22. Posición relativa de la etiqueta en el Escenario 3 (Navegación)

Esta última topología tiene múltiples aplicaciones dentro del ámbito de la navegación en interiores y la robótica. Una localización adecuada de los dispositivos UWB posibilitaría la navegación dentro de almacenes, supermercados o centros comerciales, guiando a los clientes hasta los productos o locales demandados. También permitiría a distintos dispositivos o máquinas industriales conocer su posición dentro de las naves, facilitando la automatización de los procesos.

4. CONCLUSIONES

El proyecto ha cumplido sus dos objetivos principales. Por un lado, se ha llevado a cabo el **desarrollo de sistemas de posicionamiento global portables de precisión**, que han implicado el desarrollo de un sistema de captura de datos, de un sistema de post-procesado para convertir los datos brutos obtenidos por el sistema de GPS de precisión en datos comprensible y usables y por último, el desarrollo de un sistema de post-procesado capaz de proporcionar un único fichero con todos los datos capturados de forma sincronizada.

Los sistemas descritos fueron validados para comprobar que los resultados de posicionamiento obtenidos eran repetitivos y coherentes y que toda la información que se generaba tanto de posiciones reales, posiciones interpoladas, tiempos reales y tiempos interpolados, se encontraban ordenados temporalmente, que existía una conexión real y correcta entre las posiciones GPS y su marca temporal y que los datos interpolados eran coherentes y consistentes, en comparación con los valores reales no interpolados.

En cuanto al desarrollo de un **sistema de posicionamiento en entornos indoor**, se ha realizado un estudio de las tecnologías disponibles, seleccionado *Ultrawideband* (UWB) para la realización de una prueba de concepto a fin de evaluar la usabilidad de la tecnología en tres topologías: seguimiento, *geo-fencing* y navegación.

Los resultados obtenidos tienen aplicabilidad en futuros desarrollos a llevar a cabo en el campo del seguimiento, logística y seguridad en empresas de diversos sectores. Los sistemas desarrollados en base a tres premisas que son precisión, portabilidad y solución global (incluido post-procesado), lo

que les permite adaptarse y personalizarse de forma sencilla a distintos ámbitos y lugares de aplicación.