



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

Desarrollo de una aplicación para la localización de robots al cuidado de personas en sus hogares

Trabajo Fin de Grado

Grado en Ingeniería Informática

Autor: Alexis Landa Benítez

Cotutores: Enrique Jorge Bernabeu Soler

Ángel Valera Fernández

2017-2018

Desarrollo de una aplicación para la localización de robots al cuidado de personas en sus hogares

Resumen

En el presente trabajo se explicará la tecnología RFID y sus posibilidades. Además, se mostrará la aplicación desarrollada que permite la detección, la lectura y la escritura de las etiquetas, así como la configuración del lector y las antenas utilizadas. Funcionalidades importantes para el futuro desarrollo de un sistema de localización de robots en el hogar, sistema que utiliza las etiquetas como fuente de información del entorno.

Con este sistema se evitarían técnicas de reconocimiento de formas para identificar objetos y se crearía un sistema de ubicación preciso y de bajo coste en un entorno cerrado. De esta manera, se podrá restar carga computacional al propio robot, mejorando el rendimiento en cuanto al tiempo de respuesta y consumo de energía.

Palabras clave: RFID, lectura de etiquetas RFID, escritura de etiquetas RFID, robot, sistema de localización.

Abstract

This paper will explain RFID technology and its possibilities. In addition, it will show the application developed that allows the detection, reading and writing of tags, as well as the configuration of the reader and the antennas used. Important functionalities for the future development of a system for locating robots in the home, a system that uses labels as a source of information about the environment.

Such a system would avoid shape-recognition techniques for identifying objects and create an accurate and low-cost location system in a closed environment. In this way, computational load can be subtracted from the robot itself, improving performance in terms of response time and energy consumption.

Keywords: RFID, RFID tag reading, RFID tag writing, robot, location system.

Índice de contenido

1	Introducción	7
1.1	Agradecimientos	7
1.2	Motivación y justificación del contenido	7
1.3	Objetivos	8
1.4	Estructura de la memoria.....	8
2.	Estado del arte.....	11
2.1	Introducción.....	11
2.2	Arquitectura.....	11
2.3	Tipos de etiqueta	13
2.4	Organización de la memoria.....	14
2.5	Aplicaciones.....	16
2.6	Propuesta.....	17
3.	Análisis del problema	19
3.1	Identificación del problema.....	19
3.2	Solución propuesta	22
4.	Diseño de la solución	23
4.1	Arquitectura del sistema	23
4.2	Diseño detallado	23
4.3	Tecnología utilizada	25
5.	Desarrollo de la solución propuesta.....	27
5.1	Menu principal.....	27
5.2	Búsqueda de etiquetas	28
5.3	Lectura de etiquetas	29
5.4	Escritura de etiquetas	31
5.5	Test de distancia.....	37
5.6	Configuración del lector	39
6.	Pruebas.....	43
6.1	Lectura de etiquetas	43
6.2	Escritura de etiquetas	47
6.3	Detección de etiquetas	49
7.	Conclusiones	51
7.1	Trabajo realizado	51
7.2	Problemas.....	52
7.3	Posibles mejoras.....	52
	Referencias.....	53

Tabla de figuras

2.1	Esquema de funcionamiento de la tecnología RFID	12
2.2	Bandas de frecuencia en tecnología RFID	13
2.3	Bancos y bloques de memoria del Class 1 Gen 2.....	14
3.1	Robot donde irá incorporado las antenas y el lector	19
3.2	Lector RFID SkyeModule Nova.....	20
3.3	Antenas del lector	20
3.4	Etiquetas RFID flexibles de tipo pegatina	21
3.5	Etiquetas RFID rígidas de tipo pegatina	21
3.6	Etiquetas RFID rígidas para atornillar	22
4.1	Configuración de los componentes.....	23
4.2	Arquitectura software.....	24
4.3	Logo de Skyetek.....	25
4.4	Logo de C#	25
4.5	Logo de Microsoft Visual Studio.....	25
5.1	Menú principal	27
5.2	Opción ‘ <i>Search Tags</i> ’	28
5.3	Búsqueda finalizada	28
5.4	Selección de etiqueta	29
5.5	Resultado de la lectura.....	29
5.6	Aviso al leer el banco de memoria	30
5.7	Menú de escritura	31
5.8	Opción ‘ <i>Write new EPC</i> ’	31
5.9	Inserción del nuevo EPC.....	32
5.10	EPC actualizado	32
5.11	Editar un bloque de memoria	33
5.12	Inserción de un nuevo bloque	33
5.13	Banco de memoria EPC actualizado	34
5.14	Escritura en la memoria del usuario.....	34
5.15	Memoria del usuario actualizada	35
5.16	Elección de bloque en memoria de usuario.....	36
5.17	Cambios generados en memoria de usuario	36
5.18	Cambio de contraseña de acceso.....	36
5.19	Cambio de contraseña de inhabilitación	37
5.20	Test de distancia	37
5.21	Detección de la etiqueta	38
5.22	Configuración del lector (‘ <i>Show Reader Configuration</i> ’).....	39
5.23	Menú configuración del lector (‘ <i>Modify Reader Configuration</i> ’).....	40



5.24	Menú 'Host Interface Type'	40
5.25	Menú 'RegulatoryMode'	40
5.26	Menú ' <i>PowerLevel</i> '	41
5.27	Estrategias Frequency Hopping Sequence	41
5.28	Menú 'ModulationDepth'	42
5.29	Menú ' <i>RetryCount</i> '	42
6.1	Etiqueta flexible de tipo pegatina	43
6.2	Lectura de etiqueta flexible	43
6.3	Etiqueta rígida de tipo pegatina	44
6.4	Lectura de etiqueta rígida de tipo pegatina	45
6.5	Etiqueta rígida.....	46
6.6	Lectura de tarjeta rígida	46
6.7	Configuración de la etiqueta antes de los cambios	47
6.8	Configuración después de los cambios	48
6.9	Resultados de la distancia a la que pueden ser detectadas las etiquetas.....	49

1 Introducción

1.1 Agradecimientos

Me gustaría dar mi agradecimiento a mis tutores, Enrique Jorge Bernabeu Soler y Ángel Valera Fernández, por todo el apoyo que me han otorgado.

Desde el primer momento me han guiado y resuelto todas las dudas que me han surgido, además de darme la flexibilidad para la realización de este trabajo.

1.2 Motivación

El proyecto consiste en el desarrollo, en primer lugar, de una aplicación capaz de escribir y leer un código en una tarjeta RFID. Poniendo adecuadamente las tarjetas en paredes y techo de un hogar, se puede localizar aproximadamente la posición del robot. El robot es el que tiene el lector, por lo tanto, según donde se ubique el robot, este es capaz de leer el código de sólo algunas tarjetas. Según las tarjetas que pueda ser capaz de leer, se puede estimar la posición aproximada del robot.

Las tarjetas también se pueden adherir a objetos (mesa, silla, televisión, ...) con lo que la lectura de un código podría informar al robot de información sobre la forma y uso del objeto.

Las tarjetas RFID proporcionan reducción de coste computacional para el desplazamiento y seguimiento de una persona en el entorno de su hogar. Esto no significa que se tengan que eliminar sensores del robot como: cámara (realiza tarea de vigilancia como detección de posibles caídas), infrarrojos (para la detección de obstáculos cercanos al robot), envío de mensaje de ayuda (en caso de detección de alguna situación anómala de la persona), etc.

Por lo que se ha desarrollado en este TFG una API y una aplicación software que permite leer y escribir código en una serie de tarjetas RFID, realizar pruebas de distancia y configurar el lector. Esta aplicación va a permitir ser utilizada en un robot en el hogar de una persona para realizar tareas de vigilancia.

1.3 Objetivos

Los objetivos que pretende conseguir este trabajo con la creación de una API y una aplicación software son:

API

- Permitir la detección de etiquetas en el entorno.
- Posibilitar la lectura y escritura de datos de las etiquetas RFID.
- Permitir hacer cambios de la configuración del lector.

Aplicación software

- Crear una interfaz para facilitar:
 - La visualización de los bancos de memoria.
 - La lectura y escritura en etiquetas.
 - Las pruebas de distancia a la que el lector puede detectar una etiqueta.
 - La visualización y modificación de la configuración del lector RFID.

1.4 Estructura de la memoria

La memoria se ha dividido en 7 capítulos.

En el primer capítulo se explica de donde nace la idea de desarrollar un sistema de localización con el uso de la tecnología RFID. También se detallan los objetivos a seguir para la realización de este trabajo.

En el segundo capítulo se hace una breve introducción de la tecnología RFID. Se explica su arquitectura y funcionamiento, además de hacer un análisis de los tipos de etiquetas que existen. Por último, se describen algunas aplicaciones potenciales que puede otorgar esta tecnología.

En el tercer capítulo se analiza el problema y se propone una solución a partir de los requisitos planteados.

En el cuarto capítulo se explicará el diseño de la solución propuesta. Todos los componentes que actúan en la aplicación y tecnologías que se han utilizado para realizar la aplicación.

En el quinto capítulo se mostrará la herramienta que ha sido desarrollada. Se hará una breve guía de la aplicación a través de la vistas para comprender su funcionamiento.

En el sexto capítulo se presentará las pruebas realizadas que se han hecho al finalizar la aplicación y el correcto funcionamiento de la herramienta.

En el séptimo capítulo se expondrá las conclusiones que se han podido extraer de trabajar con la tecnología RFID. Además, se hará una valoración de la experiencia obtenida durante la realización de este trabajo.

2. Estado del arte

En este capítulo se hará una breve introducción de la tecnología RFID, tanto su arquitectura como los tipos de etiquetas y sus características. Además, se comentará posibles aplicaciones donde se puede usar este sistema y la propuesta para abordar el presente proyecto.

2.1 Introducción

RFID o identificación por radiofrecuencia (Radio Frequency Identification) es una tecnología basada en el almacenamiento y la recuperación de datos de un dispositivo mediante ondas de radio.

Estos dispositivos son comúnmente llamados etiquetas, y van fijadas o adheridas a objetos, animales o personas. El objetivo de las etiquetas es identificarse y ofrecer información propia ante la señal de un lector sin tener contacto entre sí y a una larga distancia.

Este sistema podría sustituir algún día al sistema por código de barras, ya que presenta muchas ventajas. Una de las más significativas es que el lector no necesita tener contacto directo con la entidad a identificar, no como los códigos de barras que necesitan detectar mediante infrarojos el producto. Además, el código RFID es lo suficientemente largo como para tener un identificador único que represente un artículo en particular, en cambio los códigos de barras se limitan a representar con un código todos los artículos de un tipo de producto.

De momento, se considera como una alternativa ya que los costes aún no son lo suficientemente bajos.

2.2 Arquitectura

Un sistema RFID está formado por tres componentes:

- Etiqueta RFID: formada por un chip, una antena y un transductor radio. El chip tiene un número de serie único que lo identifica y la antena se ocupa de enviar la información que identifica la tarjeta al lector.
- Lector de RFID: está formado por un transceptor, un decodificador y una antena. Su función es detectar y recibir información de las etiquetas del entorno para poder transferirlo al subsistema de procesamiento de datos. Algunos lectores permiten la escritura de datos en las tarjetas, siempre y cuando estas etiquetas tengan la función de escritura habilitada.

- Subsistema de procesamiento de datos o *middleware*: es el software encargado de procesar los datos recibidos por el lector para extraer información útil.

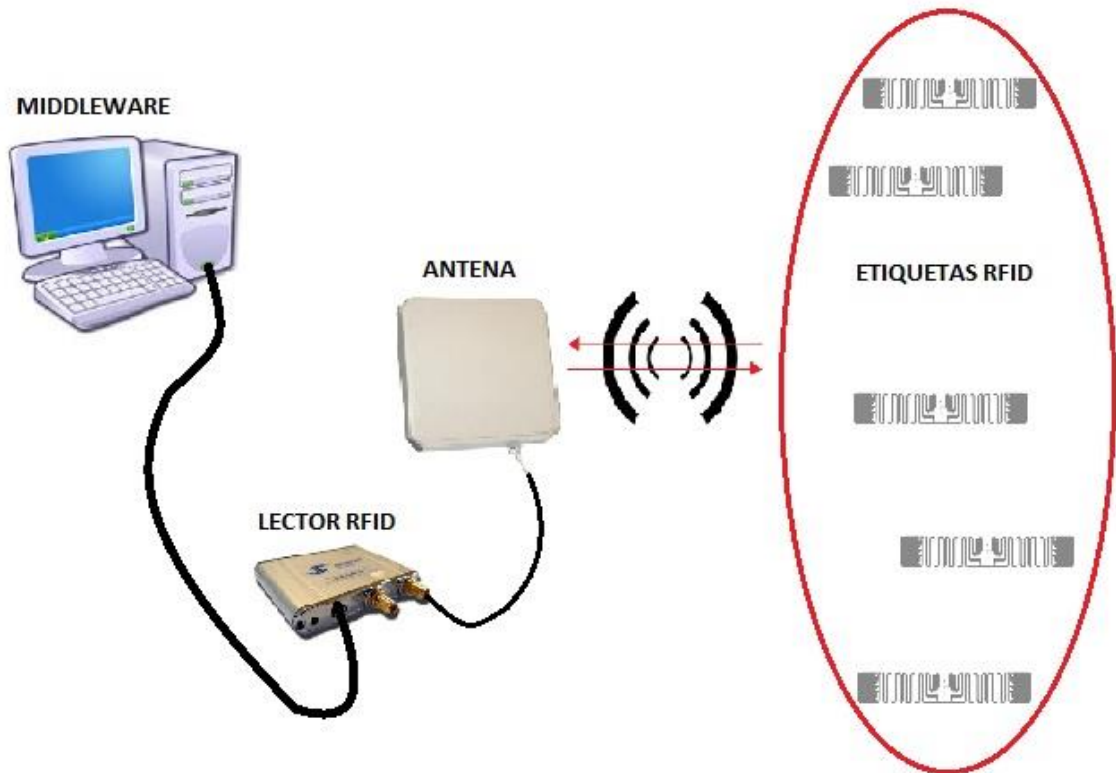


Figura 2.1: Esquema de funcionamiento de la tecnología RFID

En la figura 2.1 se puede apreciar el funcionamiento de un sistema RFID. El middleware envía una orden al lector de RFID para que este realice un tipo de operación; búsqueda, lectura o escritura de etiquetas. El lector, mediante una o varias antenas, envía peticiones por radio frecuencia a las etiquetas que se encuentren al alcance de las antenas. Cuando una etiqueta recibe una señal, esta devuelve una respuesta. La antena recoge la respuesta para que el lector decodifique ese mensaje y se lo pase finalmente al sistema de procesamiento de datos. Una vez obtenido los datos, el middleware se encarga de obtener información útil para las aplicaciones que dependan de dichas etiquetas [2].

2.3 Tipos de etiqueta

Existe una gran variedad de etiquetas RFID. Pueden ser de diferentes tamaños y fabricados con distintos materiales dependiendo del entorno donde se vaya utilizar. Pero se podría clasificar de dos maneras destacadas [3]:

Fuente de energía utilizada

Existen 3 tipos:

- **Activas:** Son tarjetas que utilizan una fuente de alimentación propia mediante una pequeña batería para suministrar corriente al circuito integrado. De esta manera, pueden transmitir la señal de respuesta al lector a una distancia mucho mayor y en entornos más complejos. Además, pueden procesar y almacenar una gran cantidad de datos. Estas etiquetas suelen ser más seguras pero caras y de mayor tamaño.
- **Pasivas:** No necesitan fuente de energía interna ya que, para poder comunicarse, usan la señal enviada por el lector cuando esta hace una petición. Este pulso electromagnético crea una corriente eléctrica que permite que el circuito integrado se active y pueda generar una respuesta al lector. Son las más baratas del mercado y suelen ser de un tamaño reducido. Normalmente, se suelen aplicar como adhesivos en infinidad de objetos o incluso insertados debajo de la piel para identificar animales.
- **Semipasivas:** Son unas etiquetas híbridas entre las activas y las pasivas. Poseen una batería como las etiquetas activas pero se comunican con el lector como las pasivas, ya que el uso de esta se utiliza primordialmente para abastecer al microchip. Su precio y tamaño ronda entre las activas y las pasivas.

Frecuencia a la que trabajan

El tipo de frecuencia establece la velocidad de transmisión de datos, la velocidad y tiempo de lectura y la distancia a la que actúan las etiquetas.

Banda de frecuencias	Denominación	Rango
125 kHz – 134 kHz	LF (Baja Frecuencia)	Hasta 45 cm.
13,553 MHz – 13,567 MHz	HF (Alta Frecuencia)	De 1 a 3 m.
400 MHz – 1.000 MHz	UHF (Ultra Alta Frecuencia)	De 3 a 10 m.
2,45 GHz – 5,4 GHz	Microondas	Más de 10 m.

Figura 2.2: Bandas de frecuencia en tecnología RFID

Generalmente, en las bandas de baja y alta frecuencia se emplean tags pasivos y en las bandas de ultra-alta frecuencia tags pasivos y activos.

2.4 Organización de la memoria

Actualmente, la mayoría de las etiquetas UHF adoptan el estándar internacional ISO 18000-6C EPC Class 1 Gen 2.

En este proyecto se ha trabajado con este tipo de tags, que tienen una determinada distribución de la memoria (es posible que algunas tarjetas difieran en ciertos bloques memoria) [4][5][6].

EPC Class 1 Gen 2

Presenta 4 bancos de memoria, del cual cada banco está formado por un número indefinido de bloques de 2 bytes de capacidad.

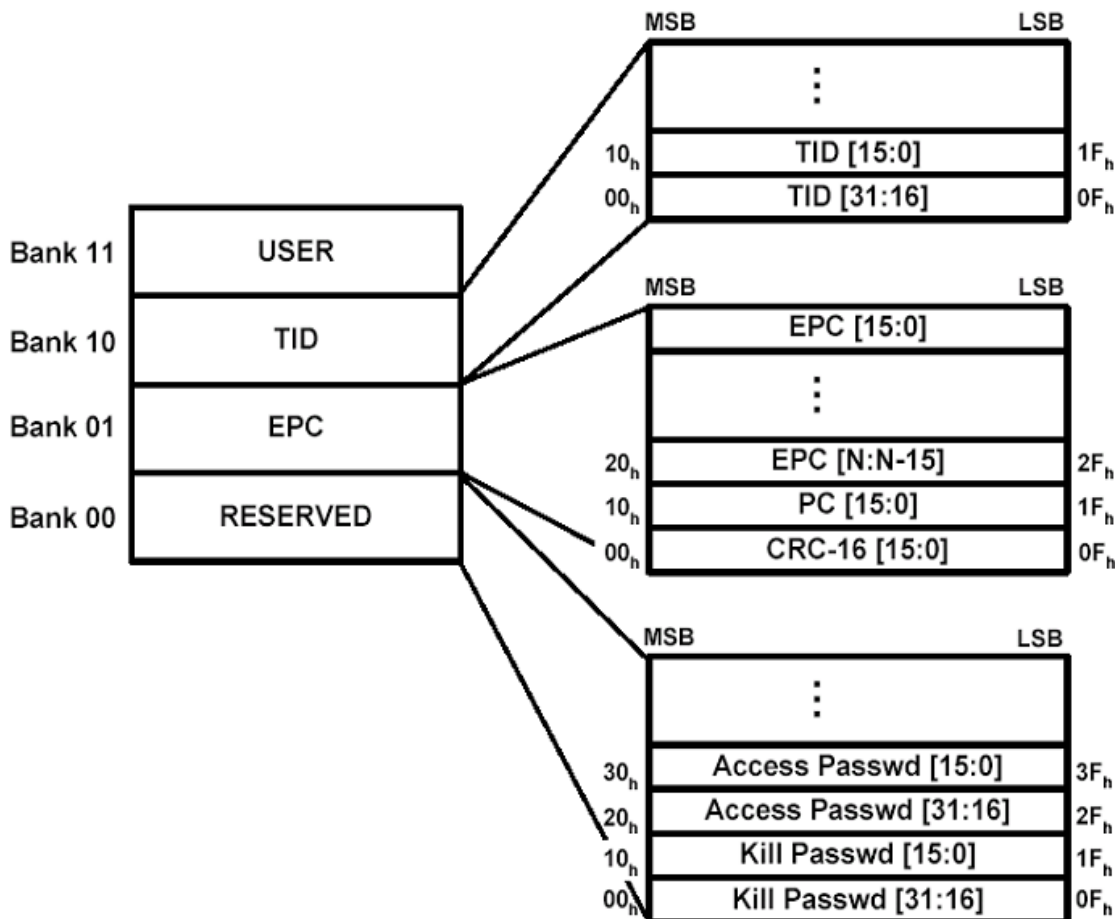


Figura 2.3: Bancos y bloques de memoria del Class 1 Gen 2

- Memoria Reservada (Bank 00)

Corresponde a la zona de memoria donde reside la contraseña de acceso y de destrucción de la etiqueta.

Los dos primeros bloques guardan la contraseña de inhabilitación, que permite al lector mediante una petición dejar la tarjeta inoperativa.

En los siguientes cuatro bytes contiguos se almacena la contraseña de acceso. Si esta contraseña existe, cada petición que haga el lector deberá ir acompañada de la misma para poder interactuar con dicha etiqueta.

Los bloques restantes están por definir y pueden ser usados como datos del usuario.

- Memoria EPC (Bank 01)

Corresponde a la zona de memoria donde reside el EPC (Electronic Product Code) e información de la configuración.

El primer bloque contiene el CRC, que indica el código de detección de errores, y el segundo el PC, que gestiona la cantidad de bits del EPC que debe devolver al lector como respuesta. Este último bloque está dividido de la siguiente manera:

- Los primeros 5 bits representan la longitud del EPC (cantidad de bloques de 16 bits que ocupa).
- El bit 6 indica si la etiqueta tiene banco de memoria del usuario o, si en cuyo caso, presenta datos del usuario.
- El bit 7 indica la existencia de XPC.
- El bit 8 identifica si la etiqueta es una aplicación EPCglobal o no (si es cero, es EPCglobal).
- Los últimos 8 bits restantes son atributos que dependen del tipo de aplicación indicado en el anterior bit.

A partir del tercer bloque reside el EPC, el cual puede variar su longitud respecto a otras etiquetas dependiendo de la cantidad de memoria que ofrezca el fabricante o del número de identificación que el usuario ha escrito. Si el usuario ha escrito un código que no ocupa la totalidad del banco de memoria, es posible reutilizar ese espacio como una extensión del banco de memoria del usuario.



- Memoria TID (Bank 10)

Corresponde a la sección de memoria donde se haya el identificador único de la etiqueta. Este identificador es introducido por el fabricante y no puede ser borrado o modificado.

A partir de dicho código, se puede obtener la clase o el modelo de la etiqueta. Se podría decir que es el número de bastidor de la tarjeta.

- Memoria del usuario (Bank 11)

Corresponde a la sección de memoria del usuario. Esta parte es opcional y algunos modelos de etiquetas pueden no ofrecer esta característica. Además, cada marca de etiqueta RFID puede ofrecer un tamaño variable de bytes.

En este banco de memoria se guarda datos que el usuario le interesaría asociar a la etiqueta, como puede ser temperatura del objeto, fecha de expiración, ubicación en forma de coordenadas o número de accesos por el lector.

2.5 Aplicaciones

A continuación se listará algunos sectores donde el uso de la tecnología RFID se está generalizando.

Sanidad

En el sector sanitario se utiliza para la localización de expedientes clínicos o de material sanitario. Además, cada día más centros hospitalarios usan muñequeras con chip para tener un seguimiento del tratamiento o para localizar a los pacientes con algún tipo de enfermedad mental, evitando que puedan acceder a determinadas zonas o escapar.

Deportes

En la carreras populares es habitual el uso de esta tecnología en dorsales o cualquier prenda que forme parte del equipamiento del corredor. Junto con un sistema de sensores conectados a un terminal y distribuidos en ciertas etapas del recorrido, los deportistas pueden obtener un seguimiento de la carrera, con resultados más precisos, sin la necesidad de ser cronometrados por el personal de la organización.

Uso militar

El ejército de los Estados Unidos ha utilizado este sistema para saber la cantidad de soldados que entraban en los hospitales en la guerra e identificarlos. De este modo, lograban tener un control de las bajas y los soldados que llegaban heridos [7].

También se emplea para tener todo el material logístico localizado y permitir el acceso a personal autorizado a ciertas zonas restringidas.

2.6 Propuesta

La idea que se propone es utilizar este sistema para que el robot sea capaz de identificar ciertas zonas del hogar mediante las etiquetas. Estas tarjetas podrían ir pegadas o atornilladas a muebles o paredes de la habitación para que el robot obtenga información útil sobre el objeto detectado y recorra la habitación en base a esa información.

3. Análisis del problema

3.1 Identificación del problema

Se pretende que el robot lleve incorporado las dos antenas, uno a cada lado, junto con el lector y que lea ciertos tipos de tags repartidos por la habitación. Además, el middleware se ejecutará en el propio sistema operativo del robot móvil.



Figura 3.1: Robot donde irá incorporado las antenas y el lector

Al implementar este sistema surgen varias preguntas entorno a la tecnología y los dispositivos con los que se va a trabajar.

Se desconoce si el lector y las antenas tienen la suficiente capacidad como para detectar etiquetas a una distancia notable que permita al robot desempeñar su función.



Figura 3.2: Lector RFID SkyeModule Nova



Figura 3.3: Antenas del lector

También no se sabe que tipo de etiqueta, dentro todas las etiquetas disponibles, dará los mejores resultados.



Figura 3.4: Etiquetas RFID flexibles de tipo pegatina



Figura 3.5: Etiquetas RFID rígidas de tipo pegatina



Figura 3.6: Etiquetas RFID rígidas para atornillar

Además, es necesario que las etiquetas dispongan de un espacio de memoria con información acorde al contexto de futuras implementaciones.

3.2 Solución propuesta

Como solución a estas dudas y requerimientos, se propone el desarrollo de una aplicación capaz de escribir y leer información de las tarjetas, comprobar la distancia de detección de una etiqueta y cambiar la configuración del lector RFID.

El objetivo de esta aplicación es ofrecer una serie de herramientas para hacer pruebas con el lector y las antenas. Así, el desarrollador puede averiguar la configuración y el tipo de etiqueta que mejor se adapta al entorno donde actuará el robot.

Además, se proporcionará una API para que el desarrollador pueda reutilizar estas funciones de comunicación con las tarjetas para la aplicación final.

4. Diseño de la solución

4.1 Arquitectura del sistema

La arquitectura necesaria para alcanzar la propuesta deseada es similar al de la figura 2.1.

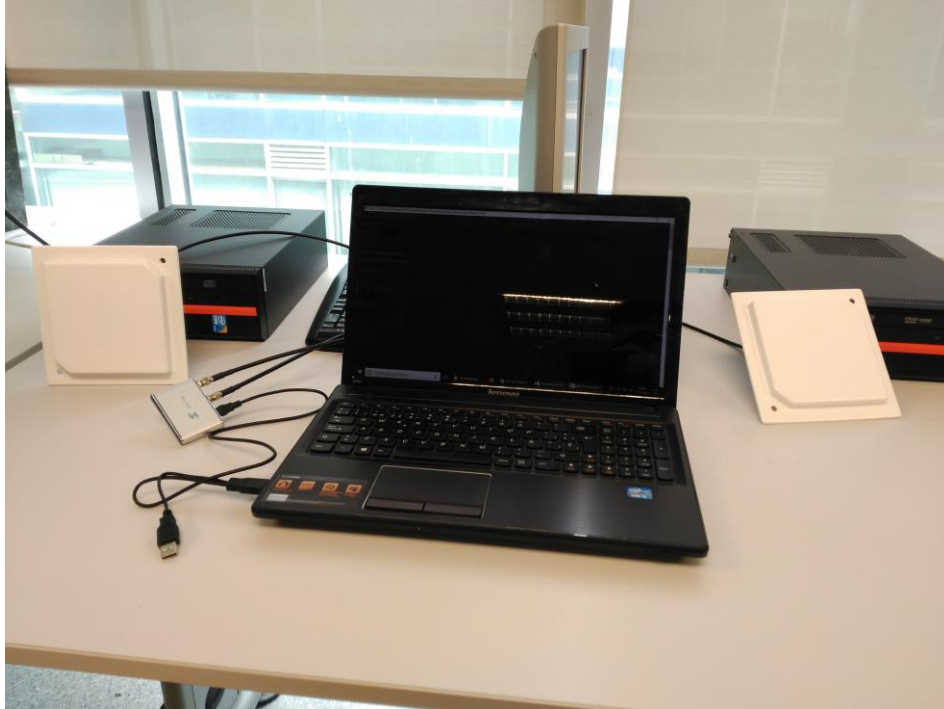


Figura 4.1: Configuración de los componentes

Las antenas están conectadas a los dos puertos del lector RFID, y este último conectado mediante cable USB al terminal.

La diferencia radica en que el middleware se ejecuta en una terminal gráfica (como un portátil), en vez del sistema operativo del robot.

4.2 Diseño detallado

Los siguientes elementos son los que interactúan con el programa principal. A continuación, se detallará la función que tiene cada clase.

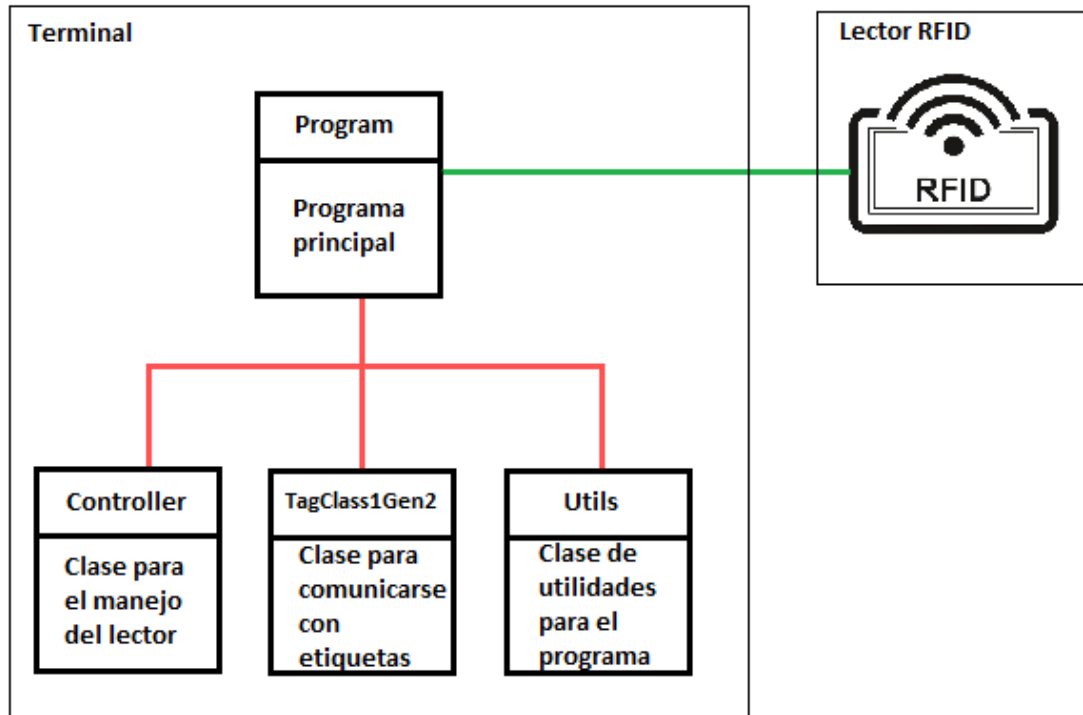


Figura 4.2: Arquitectura software

Clase Program

Es el punto de partida para la ejecución del programa. Tiene la lógica que permite el funcionamiento y la navegación de los menús.

Clase Controller

Es el encargado del manejo del lector. Con esta clase el programa puede comunicarse con el lector para modificar la configuración a la que opera. Por ejemplo, facilita cambiar la frecuencia a la que trabaja o el puerto de la antena que escucha.

Clase TagClass1Gen2

Esta clase comprende todos los métodos relacionados con la comunicación de las etiquetas. Posibilita la detección de las tarjetas y la lectura y escritura de los bancos de memoria.

Clase Utils

Conjunto de métodos que utiliza el programa principal para mostrar todo tipo de avisos, menús e información presente en la aplicación. Además, hace uso de métodos para el tratamiento y conversiones de datos.

4.3 Tecnología utilizada

Skynetek .NET API

Es una interfaz que ofrece *Skynetek*, creado bajo el framework .NET, para poder mandar órdenes a sus lectores RFID, llamados *SkyeModule readers* [8] [9].



Figura 4.3: Logo de Skynetek

Posee un conjunto de clases que encapsulan los códigos de respuesta y de petición que el lector es capaz de entender. Además, contiene la implementación del protocolo de comunicación propio de Skynetek, llamado *STPv3 protocol*.

C#

Es un lenguaje de programación orientado a objetos desarrollado por Microsoft que forma parte del framework .NET [10].



Figura 4.4: Logo de C#

Se ha utilizado este lenguaje porque, dentro todos los lenguajes soportados por .NET, es el que más versatilidad aporta a la hora de desarrollar una aplicación, su sintaxis deriva de C/C++ y su modelo de objetos es similar al de Java.

Microsoft Visual Studio

Es un entorno de desarrollo integrado que soporta variedad de lenguajes de programación [11] y el que mejor soporta el framework .NET.



Figura 4.5: Logo de Microsoft Visual Studio

Permite escribir código eficiente con rapidez y depurar de manera fácil para identificar y corregir errores de código.

5. Desarrollo de la solución propuesta

En este capítulo se mostrarán las pantallas de la herramienta y se describirá las múltiples opciones que contiene.

5.1 Menu principal

Al ejecutar el programa aparece el siguiente menú, en el que podremos acceder a ellas mediante las teclas indicadas.

```
Skyetek USB Device Created
Skyetek Reader Created

Search Tags -> 'S'
Read Tag -> 'R'
Write Tag -> 'W'
Loop Mode -> 'L'
Show Reader Configuration -> 'C'
Modify Reader Configuration -> 'M'
Close Device -> 'Esc'
```

Figura 5.1: Menú principal

Search Tags: Búsqueda y registro de etiquetas.

Read Tag: Lectura de la configuración y contenido de una etiqueta.

Write Tag: Escritura de la configuración y contenido de una etiqueta.

Loop Mode: Test para comprobar la distancia a la que se puede comunicar el lector con la etiqueta RFID.

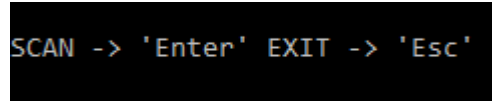
Show Reader Configuration: Muestra la configuración del lector.

Modify Reader Configuration: Permite cambiar la configuración del lector.

Close Device: Cierra la comunicación con el dispositivo y sale del programa.

5.2 Búsqueda de etiquetas

En la opción 'Search Tags' aparece la siguiente pantalla que permite buscar etiquetas para su posterior registro en la sesión.

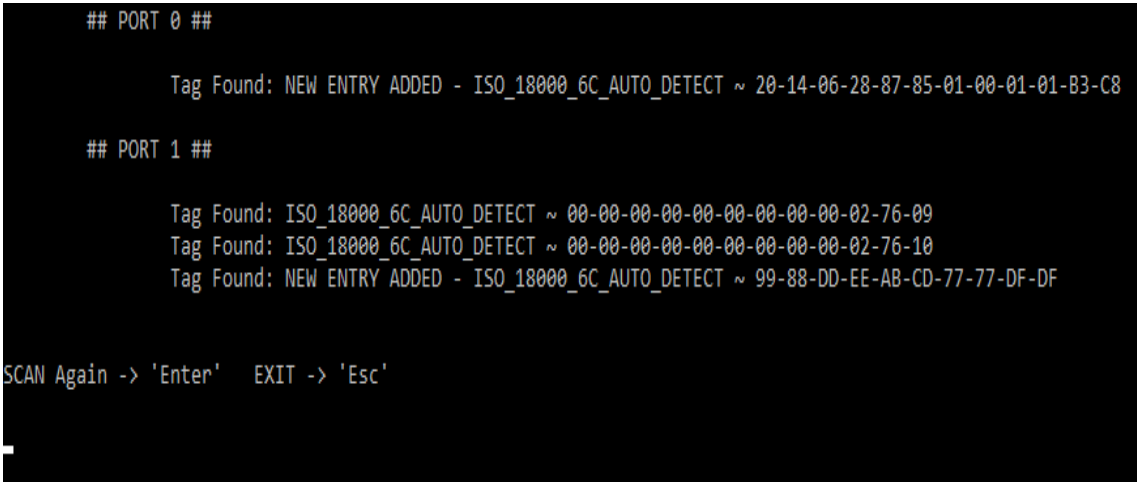


```
SCAN -> 'Enter' EXIT -> 'Esc'
```

Figura 5.2: Opción 'Search Tags'

Cabe destacar que es necesario realizar esta acción para poder utilizar las opciones que requieran interacción con una etiqueta.

Una vez pulsado 'enter', la herramienta empezará a buscar las etiquetas que estén a rango de las dos antenas y mostrará la siguiente información.



```
## PORT 0 ##  
  
Tag Found: NEW ENTRY ADDED - ISO_18000_6C_AUTO_DETECT ~ 20-14-06-28-87-85-01-00-01-01-B3-C8  
  
## PORT 1 ##  
  
Tag Found: ISO_18000_6C_AUTO_DETECT ~ 00-00-00-00-00-00-00-00-02-76-09  
Tag Found: ISO_18000_6C_AUTO_DETECT ~ 00-00-00-00-00-00-00-00-02-76-10  
Tag Found: NEW ENTRY ADDED - ISO_18000_6C_AUTO_DETECT ~ 99-88-DD-EE-AB-CD-77-77-DF-DF  
  
SCAN Again -> 'Enter' EXIT -> 'Esc'
```

Figura 5.3: Búsqueda finalizada

En esta visualización se muestra desde que puerto se ha detectado la etiqueta (puerto al que está asociado una antena), el tipo de tarjeta encontrada y su número de identificación.

Además, indica si ha sido un nuevo registro mediante el aviso 'NEW ENTRY ADDED'.

5.3 Lectura de etiquetas

La opción *'Read Tag'* permite leer el contenido de todos los bancos de memoria de una etiqueta dada y desglosa la información para facilitar el análisis del mismo [12].

```
Pick an option . . .
  00-00-00-00-00-00-00-00-02-76-10
>> 00-00-00-00-00-00-00-00-02-76-09
  20-14-06-28-87-85-01-00-01-01-B3-C8
  99-88-DD-EE-AB-CD-77-77-DF-DF
Select your choice with the arrow keys.

Press [ Esc ] to exit
```

Figura 5.4: Selección de etiqueta

Primero se elige una etiqueta de entre las registradas anteriormente. Una vez hecha la selección, es aconsejable mantener la etiqueta cerca de la antena para que el lector pueda leer cada bloque del banco de memoria sin ocasionar errores de respuesta de la etiqueta hacia el lector.

```
TAG: 99-88-DD-EE-AB-CD-77-77-DF-DF
#### USER MEMORY BANK ####
Data: [ 33-33-10-10 ]
#### TID MEMORY BANK ####
Data: [ E2-80-11-00-20-00-5A-89-E8-35-02-51 ]
      Class: E2 Mask: 801 Model: 100
#### EPC MEMORY BANK ####
Data: [ CB-D2-2C-00-99-88-DD-EE-AB-CD-77-77-DF-DF-00-00-00-00-AA-AA ]
      EPC ID: 99-88-DD-EE-AB-CD-77-77-DF-DF
      EPC DATA: 00-00-00-00-AA-AA
      CRC-16: CB-D2
      EPC Length ID: 5 blocks
      User Memory Bank present/Contains Data: YES
      XPC: NO
      [ EPCglobal application ]
      Attribute Bits: 0
#### RESERVED MEMORY BANK ####
Data: [ 00-00-00-00-00-00-00-00-00-04-A0-15-7A ]
      Kill Password: 00-00-00-00
      Access Password: 00-00-00-00
      Reserved Data: 00-00-04-A0-15-7A
Read Another Tag -> 'Enter'      Exit -> 'Esc'
```

Figura 5.5: Resultado de la lectura

Desarrollo de una aplicación para la localización de robots al cuidado de personas en sus hogares

Se puede observar todos los bancos de memoria y su contenido en representación hexadecimal, separado cada semi-bloque por un guión.

En cada zona de memoria se muestra la siguiente información (excepto en el banco de memoria del usuario, ya que esta es responsabilidad del programador):

Reserved memory bank

Kill Password: La contraseña que debe enviarse en la petición para inhabilitar el uso de la tarjeta. En caso de ser '00-00-00-00', la etiqueta carece de contraseña.

Access Password: La contraseña que debe enviarse en la petición para acceder a la memoria. En caso de ser '00-00-00-00', la etiqueta carece de contraseña de acceso y se puede leer de la etiqueta libremente.

Reserved Data: Memoria reservada de la etiqueta.

EPC memory bank

EPC ID: Identificación de la propia etiqueta.

EPC DATA: Datos introducidos por el usuario.

CRC-16: Código de detección de errores.

EPC Length ID: Número de bloques que ocupa el EPC ID.

User Memory Bank presents/contains data: Indica si el banco de memoria del usuario contiene o presenta datos.

XPC: Indicador del Protocolo de Control Extendido.

TID memory bank

Class: Clase de etiqueta

Mask: Máscara de la etiqueta

Model: Modelo de la etiqueta

Es posible que una etiqueta no disponga de memoria para el usuario. En cuyo caso, se mostrará el siguiente aviso.

```
TAG: 00-00-00-00-00-00-00-00-00-02-76-09
Error to read User Data Bank
#### TID MEMORY BANK ####
Data: [ 52 00 24 11 02 00 00 00 00 00 ]
```

Figura 5.6: Aviso al leer el banco de memoria

5.4 Escritura de etiquetas

La opción *'Write Tag'* posibilita la modificación de una etiqueta. Primero de todo, elegimos una de ellas como en la figura 3.4.

Una vez indicado el identificador, aparecerá el siguiente menú. Con el que se podrá acceder a sus opciones tecleando los números asociados a ellos.

```
TAG -> 99-88-DD-EE-AB-CD-77-77-DF-DF

[1] Write new EPC
[2] Write 1 Block on EPC Bank
[3] Write on USER Bank
[4] Write 1 Block on USER Bank
[5] Change Access Password
[6] Change Kill Password

Press 'Esc' to exit
```

Figura 5.7: Menú de escritura

Write new EPC

```
TAG -> 99-88-DD-EE-AB-CD-77-77-DF-DF

[1] Write new EPC
[2] Write 1 Block on EPC Bank
[3] Write on USER Bank
[4] Write 1 Block on USER Bank
[5] Change Access Password
[6] Change Kill Password

Press 'Esc' to exit

EPC MEMORY BANK

#####
# 1009: AA-AA #
# 1008: 00-00 #
# 1007: 00-00 #
# 1006: DF-DF #
# 1005: 77-77 #
# 1004: AB-CD #
# 1003: DD-EE #
# 1002: 99-88 #
# 1001: 2C-00 #
# 1000: CB-D2 #
#####

Press Enter with empty value to exit

EPC Memory bank: CB D2 2C 00 99 88 DD EE AB CD 77 77 DF DF 00 00 00 00 AA AA
Current EPC: 99 88 DD EE AB CD 77 77 DF DF
New EPC: _
```

Figura 5.8: Opción *'Write new EPC'*

Desarrollo de una aplicación para la localización de robots al cuidado de personas en sus hogares

Se puede observar cada bloque de la memoria y el actual identificador de la etiqueta.

Para introducir el nuevo identificador, debe tener el mismo formato que el id mostrado (con espacios por cada semi-bloque).

```
EPC Memory bank: CB D2 2C 00 99 88 DD EE AB CD
Current EPC: 99 88 DD EE AB CD 77 77 DF DF
New EPC: 99 88 DD EE AB CD 77 77 6B C5 10 2E
```

Figura 5.9: Inserción del nuevo EPC

El identificador deseado puede tener una longitud distinta a la actual. En caso de que se quisiera salir de la opción *'Write new EPC'*, se puede presionar la tecla *'Enter'* sin ningún valor dado.

```
EPC Memory bank: CB D2 2C 00 99 88 DD EE AB CD 77 77 DF DF 00 00 00 00 AA AA
Current EPC: 99 88 DD EE AB CD 77 77 DF DF
New EPC: 99 88 DD EE AB CD 77 77 6B C5 10 2E
Number of Bytes Written: 12
EPC MEMORY BANK

#####
# 1009: AA-AA #
# 1008: 00-00 #
# 1007: 10-2E #
# 1006: 6B-C5 #
# 1005: 77-77 #
# 1004: AB-CD #
# 1003: DD-EE #
# 1002: 99-88 #
# 1001: 34-00 #
# 1000: 97-A0 #
#####

New EPC Memory bank: 97-A0-34-00-99-88-DD-EE-AB-CD-77-77-6B-C5-10-2E-00-00-AA-AA

Write Another Tag -> 'Enter'   Exit -> 'Esc'
```

Figura 5.10: EPC actualizado

Si la operación se realiza con éxito, se presenta como quedaría la nueva configuración de la memoria y se actualiza la entrada en el registro de etiquetas por la nueva.

Write 1 block on EPC Bank

En esta opción se puede editar cualquier bloque de memoria que no pertenezca a los bytes de configuración.

```
TAG -> 99-88-DD-EE-AB-CD-77-77-6B-C5-10-2E

[1] Write new EPC
[2] Write 1 Block on EPC Bank
[3] Write on USER Bank
[4] Write 1 Block on USER Bank
[5] Change Access Password
[6] Change Kill Password

Press 'Esc' to exit

#####
# 1009: AA-AA # |
>> # 1008: 00-00 # | EPC DATA
# 1007: 10-2E # |
# 1006: 6B-C5 # |
# 1005: 77-77 # |
# 1004: AB-CD # |
# 1003: DD-EE # |
# 1002: 99-88 # | EPC
# 1001: 34-00 # |
# 1000: 97-A0 # | RESERVED
#####

Select your choice with the arrow keys.

_
```

Figura 5.11: Editar un bloque de memoria

Lo primero de todo, se elige un bloque de memoria y se inserta el valor deseado.

```
Press Enter with empty value to exit

EPC Block: 00-00
New EPC Block: 4D 3F_
```

Figura 5.12: Inserción de un nuevo bloque

Una vez cambiado la memoria, se muestra la nueva configuración.

```
EPC Block: 6B-C5
New EPC Block: FF FF
Block written
NEW TID: 99-88-DD-EE-AB-CD-77-77-FF-FF-10-2E
EPC MEMORY BANK

#####
# 1009: AA-AA #
# 1008: 4D-3F #
# 1007: 10-2E #
# 1006: FF-FF #
# 1005: 77-77 #
# 1004: AB-CD #
# 1003: DD-EE #
# 1002: 99-88 #
# 1001: 34-00 #
# 1000: 99-AA #
#####

Write Another Tag -> 'Enter'   Exit -> 'Esc'
```

Figura 5.13: Banco de memoria EPC actualizado

En caso de que se haya modificado un bloque perteneciente al identificador, se avisará del cambio realizado.

Write on USER Bank

Esta acción admite la escritura de datos en el banco de memoria del usuario.

```
TAG -> 99-88-DD-EE-AB-CD-77-77-BB-BB-10-2E

[1] Write new EPC
[2] Write 1 Block on EPC Bank
[3] Write on USER Bank
[4] Write 1 Block on USER Bank
[5] Change Access Password
[6] Change Kill Password

Press 'Esc' to exit

USER MEMORY BANK

#####
# 3001: 10-10 #
# 3000: 33-33 #
#####

Press Enter with empty value to exit

USER Memory bank: 33 33 10 10

New USER Data:
```

Figura 5.14: Escritura en la memoria del usuario

Para realizar esta acción es necesario pasar todos los bytes, al igual que en la opción *'Write new EPC'*. De esta manera, facilita la escritura de múltiples bytes en una sola operación.

```
TAG -> 99-88-DD-EE-AB-CD-77-77-BB-BB-10-2E

[1] Write new EPC
[2] Write 1 Block on EPC Bank
[3] Write on USER Bank
[4] Write 1 Block on USER Bank
[5] Change Access Password
[6] Change Kill Password

Press 'Esc' to exit

USER MEMORY BANK

#####
# 3001: 10-10 #
# 3000: 33-33 #
#####

Press Enter with empty value to exit

USER Memory bank: 33 33 10 10

New USER Data: 00 00 AA AA
Number of Bytes Written: 4
USER MEMORY BANK

#####
# 3001: AA-AA #
# 3000: 00-00 #
#####

New USER Memory bank: 00-00-AA-AA

Write Another Tag -> 'Enter'   Exit -> 'Esc'
```

Figura 5.15: Memoria del usuario actualizada

Como es habitual, se muestra la nueva configuración con el número de bytes que han sido escritos con éxito.

Write 1 block on USER Bank

Al igual que la opción *'Write 1 block on EPC Bank'*. Soporta la escritura de un sólo bloque elegido con su posterior visualización del cambio generado.



```
[6] Change Kill Password
Press 'Esc' to exit

#####
# 3001: AA-AA #
>> # 3000: 00-00 #
#####

Select your choice with the arrow keys.
```

Figura 5.16: Elección de bloque en memoria de usuario

```
Press Enter with empty value to exit
USER Block: 00-00
New USER Block: 00 5B
Block written
USER MEMORY BANK

#####
# 3001: AA-AA #
# 3000: 00-5B #
#####

Write Another Tag -> 'Enter'   Exit -> 'Esc'
```

Figura 5.17: Cambios generados en memoria de usuario

Change Access Password

```
Press Enter with empty value to exit
Current Access Password: 00-00-00-00
New Access Password: 45 82 5A 9C
```

Figura 5.18: Cambio de contraseña de acceso

Permite la modificación de la contraseña de acceso de la etiqueta. En este caso, es necesario pasar una contraseña que ocupe la totalidad de los 2 bytes disponibles.

Change Kill Password

```
Press 'Esc' to exit
Press Enter with empty value to exit
Current Kill Password: 00-00-00-00
New Kill Password: 22 4D 88 67
```

Figura 5.19: Cambio de contraseña de inhabilitación

Permite la modificación de la contraseña de inhabilitación de la etiqueta. Al igual que la contraseña de acceso, es necesario escribir los 4 semi-bloques para realizar la acción.

5.5 Test de distancia

A la hora de implementar una aplicación que requiere de comunicaciones, es interesante averiguar cuáles son las capacidades que ofrece el hardware con el que se trabaja.

La detección de etiquetas depende de la configuración del lector, las características de las antenas, la clase y modelo de las etiquetas y de la cantidad de obstáculos que existen en el entorno.

```
Pick an option . . .
 00-00-00-00-00-00-00-02-76-10
 00-00-00-00-00-00-00-02-76-09
>> 20-14-06-28-87-85-01-00-01-01-B3-C8
 99-88-DD-EE-AB-CD-77-77-BB-BB-10-2E
Select your choice with the arrow keys.

Press [ Esc ] to exit

DETECTING TAG -> 20-14-06-28-87-85-01-00-01-01-B3-C8

-----
|                               NOT FOUND                               |
-----
```

Figura 5.20: Test de distancia

Esta función permite comprobar a qué distancia la antena del lector detecta una etiqueta en particular. El lector envía peticiones constantemente a la etiqueta elegida y se mantiene a la espera de una respuesta por parte de ella.

```
Pick an option . . .
 00-00-00-00-00-00-00-00-02-76-10
 00-00-00-00-00-00-00-00-02-76-09
>> 20-14-06-28-87-85-01-00-01-01-B3-C8
 99-88-DD-EE-AB-CD-77-77-BB-BB-10-2E
Select your choice with the arrow keys.

Press [ Esc ] to exit

DETECTING TAG -> 20-14-06-28-87-85-01-00-01-01-B3-C8

#####
###          D E T E C T E D          ###
#####
```

Figura 5.21: Detección de la etiqueta

Cuando el lector reciba una respuesta, mostrará el aviso *'DETECTED'*. En caso contrario, se mantendrá el mensaje *'NOT FOUND'*.

5.6 Configuración del lector

La opción *'Show Reader Configuration'* muestra la configuración actual a la que trabaja el lector, y la opción *'Modify Reader Configuration'* permitirá hacer cambios en el lector. A continuación, se describen las características.

```
Reader Name: SkyeModule Nova
Product Code: 0019
Serial Number: 00000000
Firmware Version: 01010045
Hardware Version: 00000001

Host Interface Type: USB

Regulatory Mode: ETSIEurope

PowerLevel: 27,0 dBm
          Default: 27,0 dBm

Current Frequency: 867,5 Mhz
          Default: 915 Mhz
Start Frequency: 865,7 Mhz
          Default: 902,3 Mhz
Stop Frequency: 867,5 Mhz
          Default: 927,7 Mhz

Hop Channel Spacing: 600 Khz
          Default: 100 Khz

Frequency Hopping Sequence: PseudoRandom
          Default: PseudoRandom

Modulation Depth: 100%
          Default: 100%

Retry Count: 10
          Default: 10

User Port Direction: 0
          Default: 0

User Port Value: 0
          Default: 15

RID: FF-FF-FF-FF

Tag Type Set To: ISO_18000_6C_AUTO_DETECT

Exit -> 'Esc'
```

Figura 5.22: Configuración del lector (*'Show Reader Configuration'*)

```
Pick an option . . .
>> RegulatoryMode
    HostInterfaceType
    FrequencyHoppingSequence
    PowerLevel
    ModulationDepth
    RetryCount
    Exit
Select your choice with the arrow keys.
Press [ Esc ] to exit
```

Figura 5.23: Menú configuración del lector (*'Modify Reader Configuration'*)

Reader Name: Nombre del lector RFID

Product Code: Código del modelo RFID SkyeTek

Serial number: Número de serie

Firmware Version: Versión del software cargado en el lector

Hardware Version: Versión del lector RFID

Host Interface Type: Tipo de la interfaz de comunicación del lector. Normalmente se utilizará vía USB, por lo que no será necesario hacer cambios en este apartado.

```
Pick an option . . .
>> TTL
    SPI
    I2C
    USB
    Exit
Select your choice with the arrow keys.
Press [ Esc ] to exit
```

Figura 5.24: Menú *'Host Interface Type'*

Regulatory Mode: Cambia la configuración de las frecuencias en base al estándar de la región seleccionada.

```
Pick an option . . .
>> NoPulseShaping
    ETSIEurope
    Korea
    Exit
Select your choice with the arrow keys.
Press [ Esc ] to exit
```

Figura 5.25: Menú *'RegulatoryMode'*

PowerLevel: Potencia a la que opera el lector, expresada en dBm. Además, contiene el valor por defecto en caso de reinicio del dispositivo.

Se puede insertar un nuevo valor con '*SetNewValue*', reiniciar al valor por defecto con '*SetDefaultValue*' o cambiar el valor por defecto con '*ModifyDefaultValue*'.

```
Pick an option . . .
>> SetNewValue
    SetDefaultValue
    ModifyDefaultValue
    Exit
Select your choice with the arrow keys.

Press [ Esc ] to exit

RF output Power level may be set in steps of 0,1 dB from 10 dBm to 27 dBm

Press Enter with empty value to exit
Decimal values has to be with ','
New Power Level: 15,8

Power Level set to 15,8
```

Figura 5.26: Menú '*PowerLevel*'

Current Frequency, *Start Frequency* y *Stop Frequency*: Muestra las frecuencias, expresada en megahercios, a la que trabaja el lector y todos los valores por defecto; frecuencia actual, frecuencia límite inferior y frecuencia límite superior respectivamente.

Para hacer cambios en las frecuencias es necesario utilizar la opción '*RegulatoryMode*' para que haga los cambios acorde a los estándares establecidos.

Hop Channel Spacing: Salto de frecuencia a la hora de hacer el barrido de peticiones a las etiquetas cercanas.

Frequency Hopping Sequence: Cambia entre tipo de estrategia en la secuencia de salto. Puede ser aleatorio (*PseudoRandom*) o secuencial (*sequentialHoppingMode*)

```
Pick an option . . .
>> sequentialHoppingMode
    PseudoRandom
    Exit
Select your choice with the arrow keys.

Press [ Esc ] to exit
```

Figura 5.27: Estrategias *Frequency Hopping Sequence*

Modulation Depth: Cambia la profundidad de modulación. Siendo el mínimo 10% y el máximo 100%.

```
Pick an option . . .
  SetNewValue
  SetDefaultValue
>> ModifyDefaultValue
  Exit
Select your choice with the arrow keys.

Press [ Esc ] to exit

Press Enter with empty value to exit
The values for the modulation depths are in steps of 10, ranging from 10% to 100%

New Default Modulation Depth: 80
Default Modulation Depth set to 80
```

Figura 5.28: Menú 'ModulationDepth'

Al igual que 'PowerLevel', se puede insertar un nuevo valor con 'SetNewValue', reiniciar al valor por defecto con 'SetDefaultValue' o cambiar el valor por defecto con 'ModifyDefaultValue'.

Retry Count: Número de veces que una orden se reintenta internamente en el lector para comunicarse con las etiquetas.

```
Pick an option . . .
>> SetNewValue
  SetDefaultValue
  ModifyDefaultValue
  Exit
Select your choice with the arrow keys.

Press [ Esc ] to exit

New Retry Count: 5
Retry Count set to 5
```

Figura 5.29: Menú 'RetryCount'

Como es habitual, también existe la opción de reiniciar o insertar el valor por defecto.

User Port Direction: Dirección del puerto de la antena a la que escucha el lector.

User Port Value: Valor del puerto de la antena.

RID: Identificador del lector.

Tag Type Set To: Modelo de etiqueta RFID con la que va a comunicarse.

6. Pruebas

En este capítulo se presentarán las pruebas que se han realizado para confirmar que la herramienta funciona correctamente.

6.1 Lectura de etiquetas

Como podemos observar, al leer de distintas tarjetas, recuperamos diferentes composiciones de bancos de memoria.



Figura 6.1: Etiqueta flexible de tipo pegatina

Al leer la etiqueta de la figura 6.1, obtenemos un banco de memoria para el usuario de 4 bytes de capacidad y un banco de memoria de EPC con una gran dimensión.

En la memoria del EPC muestra un identificador bastante largo que no llega a completar todos los bytes disponibles, por lo que se podría utilizar esa memoria como una extensión de la memoria del usuario.

Una ventaja de tener un tamaño extenso de memoria de EPC es que permite la creación de un identificador único, ya que hay más probabilidades de que el identificador no se vuelva a repetir con otra tarjeta.

```
TAG: 99-88-DD-EE-AB-CD-77-77-DF-DF
#### USER MEMORY BANK ####
Data: [ 33-33-10-10 ]
#### TID MEMORY BANK ####
Data: [ E2-80-11-00-20-00-5A-89-E8-35-02-51 ]
      Class: E2  Mask: 801  Model: 100
#### EPC MEMORY BANK ####
Data: [ CB-D2-2C-00-99-88-DD-EE-AB-CD-77-77-DF-DF-00-00-00-00-AA-AA ]
      EPC ID: 99-88-DD-EE-AB-CD-77-77-DF-DF
      EPC DATA: 00-00-00-00-AA-AA
      CRC-16: CB-D2
      EPC Length ID: 5 blocks
      User Memory Bank present/Contains Data: YES
      XPC: NO
      [ EPCglobal application ]
      Attribute Bits: 0
#### RESERVED MEMORY BANK ####
Data: [ 00-00-00-00-00-00-00-00-00-04-A0-15-7A ]
      Kill Password: 00-00-00-00
      Access Password: 00-00-00-00
      Reserved Data: 00-00-04-A0-15-7A
Read Another Tag -> 'Enter'      Exit -> 'Esc'
```

Figura 6.2: Lectura de etiqueta flexible

Pero no siempre tendremos la posibilidad de almacenar datos en memoria. Como se puede ver en la figura 6.4, se ha hecho una lectura de la etiqueta de la figura 6.3.



Figura 6.3: Etiqueta rígida de tipo pegatina

Esta tarjeta no dispone de memoria del usuario, por lo que aparece un error el cual el lector dice que no es capaz de leer esa zona de memoria. También la memoria del EPC es más pequeña y su identificador ocupa la totalidad de esta. En caso de que se necesitara almacenar información, se debería acortar la longitud del identificador y reutilizar ese espacio que deja vacío.

```

TAG: 00-00-00-00-00-00-00-00-02-76-09
Error to read User Data Bank
#### TID MEMORY BANK ####
Data: [ E2-00-34-11-03-B8-00-00-00-02 ]
      Class: E2 Mask: 003 Model: 411
#### EPC MEMORY BANK ####
Data: [ 50-1B-30-00-00-00-00-00-00-00-00-02-76-09 ]
      EPC ID: 00-00-00-00-00-00-00-00-02-76-09
      EPC DATA:
      CRC-16: 50-1B
      EPC Length ID: 6 blocks
      User Memory Bank present/Contains Data: NO
      XPC: NO
      [ EPCglobal application ]
      Attribute Bits: 0
#### RESERVED MEMORY BANK ####
Data: [ 00-00-00-00-00-00-00-00 ]
      Kill Password: 00-00-00-00
      Access Password: 00-00-00-00
      Reserved Data:
Read Another Tag -> 'Enter'      Exit -> 'Esc'

```

Figura 6.4: Lectura de etiqueta rígida de tipo pegatina

Por otra parte, tenemos la etiqueta rígida. Adecuada para entornos industriales, no es de esperar que ofrezca la mejores características de entre toda la batería de etiquetas con las que se ha trabajado.

6.2 Escritura de etiquetas

Para hacer las pruebas de escritura de datos, se ha utilizado la etiqueta de la figura 6.1. Esta es la configuración que tenía antes de todos los cambios.

```
TAG: 99-88-DD-EE-AB-CD-77-77-DF-DF
#### USER MEMORY BANK ####
Data: [ 33-33-10-10 ]
#### TID MEMORY BANK ####
Data: [ E2-80-11-00-20-00-5A-89-E8-35-02-51 ]
      Class: E2 Mask: 801 Model: 100
#### EPC MEMORY BANK ####
Data: [ CB-D2-2C-00-99-88-DD-EE-AB-CD-77-77-DF-DF-00-00-00-00-AA-AA ]
      EPC ID: 99-88-DD-EE-AB-CD-77-77-DF-DF
      EPC DATA: 00-00-00-00-AA-AA
      CRC-16: CB-D2
      EPC Length ID: 5 blocks
      User Memory Bank present/Contains Data: YES
      XPC: NO
      [ EPCglobal application ]
      Attribute Bits: 0
#### RESERVED MEMORY BANK ####
Data: [ 00-00-00-00-00-00-00-00-00-00-04-A0-15-7A ]
      Kill Password: 00-00-00-00
      Access Password: 00-00-00-00
      Reserved Data: 00-00-04-A0-15-7A
Read Another Tag -> 'Enter'      Exit -> 'Esc'
```

Figura 6.7: Configuración de la etiqueta antes de los cambios

En la siguiente imagen se puede observar las modificaciones realizadas con éxito. Se ha cambiado ambas contraseñas y se han modificado los datos del usuario. Incluso se ha alterado los dos últimos bloques del identificador y añadido dos más.

```
TAG: 99-88-DD-EE-AB-CD-77-77-BB-BB-10-2E
#### USER MEMORY BANK ####
Data: [ 00-5B-AA-AA ]
#### TID MEMORY BANK ####
Data: [ E2-80-11-00-20-00-5A-89-E8-35-02-51 ]
      Class: E2 Mask: 801 Model: 100
#### EPC MEMORY BANK ####
Data: [ 01-0F-30-00-99-88-DD-EE-AB-CD-77-77-BB-BB-10-2E-4D-3F-AA-AA ]
      EPC ID: 99-88-DD-EE-AB-CD-77-77-BB-BB-10-2E
      EPC DATA: 4D-3F-AA-AA
      CRC-16: 01-0F
      EPC Length ID: 6 blocks
      User Memory Bank present/Contains Data: NO
      XPC: NO
      [ EPCglobal application ]
      Attribute Bits: 0
#### RESERVED MEMORY BANK ####
Data: [ 22-4D-88-67-45-82-5A-9C-00-00-04-A0-15-7A ]
      Kill Password: 22-4D-88-67
      Access Password: 45-82-5A-9C
      Reserved Data: 00-00-04-A0-15-7A
Read Another Tag -> 'Enter'      Exit -> 'Esc'
```

Figura 6.8: Configuración después de los cambios

Cabe destacar que el apartado de *'User Memory Bank present/Contains Data'* indica que no contiene datos en la memoria de usuario. Esto es porque utiliza un bit del primer bloque como indicador. Un cero insertado en ese bit representa que no dispone de datos del usuario, a pesar de haber almacenado datos. Esta característica se le puede restar importancia porque quizás no resulte atractiva para la aplicación que se quiere desarrollar.

6.3 Detección de etiquetas

Se han hecho pruebas con toda la colección de etiquetas disponibles para comprobar la distancia a la que las antenas pueden detectar una etiqueta.

Las etiquetas que mejores resultados han conseguido son las rígidas, seguido de las tarjetas rígidas con pegatina y las flexibles.

Tipos de etiqueta	Distancia alcanzada
Pegatina	< 4 m
Rígidas con pegatina	> 4 m.
Rígida	6 m

Figura 6.9: Resultados de la distancia a la que pueden ser detectadas las etiquetas

Los tags rígidos han logrado alcanzar una distancia de 6 metros con una detección casi continua. Era de esperar que este tipo de etiquetas dieran buenos resultados ya que están diseñados para usarse en el ámbito industrial.

En cambio, las demás etiquetas rondaban distancias de 4 metros con una detección intermitente. Las etiquetas de tipo pegatina, su distancia máxima fue de 4 metros. Por otra parte, las etiquetas rígidas con pegatina mejoraron un poco el rango de detección respecto a las de tipo pegatina, pero no lo suficiente como para marcar una diferencia notable.

Los resultados obtenidos pueden variar por la frecuencia a la que trabajan las antenas, el grado de inclinación o la calidad de la antena. Incluso la cantidad de ruido del entorno puede afectar a los resultados, por lo que es posible que no se haya alcanzado todo el potencial que puede otorgar cada una de las etiquetas.

Por eso mismo, estas pruebas son una estimación para poder seleccionar las mejores tarjetas RFID de entre toda la batería de etiquetas disponibles.

7. Conclusiones

En este capítulo se realizará un resumen de todo lo desarrollado e investigado, de los problemas surgidos a lo largo del proyecto y de las mejoras que pueden hacerse en la herramienta.

7.1 Trabajo realizado

Al inicio de este proyecto se establecieron unos objetivos claros para la creación de una API y una aplicación software.

Para ello se tuvo que hacer un estudio sobre la composición de la memoria de las tarjetas y comprender el funcionamiento del lector RFID con el que se estaba trabajando.

La librería implementada ha sido utilizada para la creación de la aplicación, y sus diversas funciones han servido de utilidad para crear nuevas opciones que no habían sido definidas al principio en la herramienta.

En lo que respecta a la detección de etiquetas, se puede observar en la opción *'Search tags'* que la API es capaz de detectar etiquetas y de distinguir de qué antena proviene la respuesta de la tarjeta (característica muy importante para poder saber la orientación del robot).

Además, se puede confirmar que las funciones creadas para la lectura y escritura de código en las tarjetas funciona perfectamente. Como muestra las opciones *'Read tag'* y *'Write tag'*, permiten leer y escribir datos en cada bloque de memoria.

Y por último y no menos importante, permite visualizar y hacer cambios de la configuración del lector como se puede ver en las opciones *'Show reader configuration'* y *'Modify reader configuration'*. Funciones muy importantes para poder hacer pruebas con el lector y así obtener el mayor potencial del dispositivo a la hora de incorporarlo al robot.

Como se ha podido visualizar a lo largo del trabajo, podemos concluir que la API desarrollada cumple con los objetivos que se marcaron al comienzo de este trabajo.

Por otra parte, los requisitos establecidos para la creación de la herramienta han sido cumplidos. Como se puede observar en el capítulo 5, se ha desarrollado una interfaz que permita visualizar la información, tanto de las etiquetas como del lector, de una manera estructurada y desglosada. Además, facilita la escritura de código en la tarjetas y la realización de pruebas de distancia de detección (otra característica importante a la hora de seleccionar cual será la etiqueta que mejor se adapta para el sistema de localización).

En resumen, se puede concluir que todos los objetivos marcados se han completado satisfactoriamente, aunque todo es susceptible de ser mejorado. La realización de este trabajo ha sido muy reconfortante porque la intención principal de trabajar en un proyecto de estas características era la de desarrollar una aplicación relacionada con la rama de la robótica. Con una tecnología totalmente desconocida que me empujara a investigar y que me supusiera un reto a la hora de resolver problemas.

7.2 Problemas

Uno de los principales problemas que han surgido durante el desarrollo ha sido el desconocimiento de la tecnología RFID. Al ser una tecnología que no se ha estudiado en la carrera, ha supuesto que una parte del tiempo del desarrollo se invirtiera en investigar e informarse más que en desarrollar.

Además, se ha tenido que hacer un estudio del funcionamiento del lector RFID, donde han surgido varios problemas inesperados.

Habitualmente, el fabricante facilita los manuales de sus productos y el manual de la API con algunas demostraciones que sirvan de guía para el desarrollo. Pero en este caso el manual de la API no coincidía con la librería que se iba a utilizar. Se intentó acceder a la página web del fabricante y descargar toda la documentación necesaria, pero el material no se encontraba disponible.

Adicionalmente, la librería con la que se ha trabajado presentaba algunos fallos de comportamiento por la falta de mantenimiento del software por parte de la empresa. Por lo que se ha tenido que idear nuevas soluciones para corregir esos defectos.

Todo esto ha implicado que la curva de aprendizaje se ralentizará, causando una replanificación de las tareas por el aumento del tiempo dedicado.

7.3 Posibles mejoras

La actual herramienta utiliza la ventana del símbolo del sistema para navegar entre los menús y mandar órdenes al lector. Una de las posibles mejoras de la aplicación sería crear una interfaz más atractiva y dinámica.

Esto mejoraría la experiencia del usuario, ya que podría acceder a las opciones más rápido y le permitiría analizar mejor los resultados de las pruebas.

Otra mejora podría ser la creación de una base de datos de etiquetas detectadas. Actualmente, las etiquetas detectadas se mantienen registradas en sesión. Por lo que una vez que se cierra la aplicación, se pierde los datos y sería necesario volver a escanearlas para manipularlas.

Referencias

- [1] Wikipedia. *RFID*
<<https://es.wikipedia.org/wiki/RFID>> [Consulta: 20 de Octubre de 2017]
- [2] RFID 4u. *Cheat Sheet – RFID Architecture Components*
<<https://rfid4u.com/explore/rfid-certification-prep/cheat-sheet-rfid-architecture-components/>> [Consulta: 20 de Octubre de 2017]
- [3] Smiley, S. (2016). “Active RFID vs. Passive RFID: What’s the Difference?” en *RFID INSIDER Tracking the rfid industry*
<<https://blog.atlasrfidstore.com/active-rfid-vs-passive-rfid>> [Consulta: 23 de Octubre de 2017]
- [4] RFID 4u. *EPC Gen2 Memory*
<<https://rfid4u.com/rfid-basics-resources/epc-gen2-memory/>> [Consulta: 23 de Octubre de 2017]
- [5] Smiley, S. (2017). “17 Things You Might Not Know About Gen 2 RFID Tag Memory Banks” en *RFID INSIDER Tracking the rfid industry*
<<https://blog.atlasrfidstore.com/17-things-might-not-know-gen-2-rfid-tag-memory-banks>> [Consulta: 23 de Octubre de 2017]
- [6] Traub, K. (2012). “RFID Tag Data Standards” en *RFID Journal LIVE! Track and Manage Everything*
<http://www.rfidjournal.net/masterPresentations/rfid_live2012/pw/traub_apr3_230_ITPr of.pdf> [Consulta: 23 de Octubre de 2017]
- [7] Yoshida, J. (2003). “U.S. Navy uses RFID technology to track wounded in Iraq” en *EETimes Connecting the Global Electronics Community*
<https://www.eetimes.com/document.asp?doc_id=1145896> [Consulta: 20 de Junio de 2018]
- [8] Jadak, A Novanta Company. *Nova Documentation Category*
<<https://www.jadaktech.com/documentation/rfid/nova/>> [Consulta: 8 de Noviembre de 2017]
- [9] Jadak, A Novanta Company. *SkyeModule Nova User Guide*
<<https://www.jadaktech.com/wp-content/uploads/2017/06/Nova-User-Guide-06122017.pdf>> [Consulta: 8 de Noviembre de 2017]
- [10] Pagina de Microsoft. *Guía de C#*
<<https://docs.microsoft.com/es-es/dotnet/csharp/>> [Consulta: 13 de Noviembre de 2017]
- [11] Pagina de Microsoft. *Visual Studio*
<<https://visualstudio.microsoft.com/es/>> [Consulta: 13 de Noviembre de 2017]
- [12] Ken Traub Consulting LLC. *Encoders and Decoders for the Electronic Product Code (EPC) and RFID*
<<http://www.kentraub.com/tools.htm>> [Consulta: 17 de Enero de 2018]

