



FACULTAD DE CIENCIAS

SISTEMA DE SEGUIMIENTO DE ACTIVOS CON TECNOLOGÍA RFID

(Asset tracking system with RFID
technology)

Trabajo de fin de grado para acceder al
GRADO EN INGENIERÍA INFORMÁTICA

Autor: Rubén García González

Tutora: Inés González Rodríguez

Tutor en la empresa: Ángel Tezanos

RESUMEN

El seguimiento de activos en la industria es un problema generalizado debido a la gran cantidad de stock y el amplio espacio de trabajo o almacenamiento que pueden llegar a manejar los activos. Muchas empresas optan por realizar un seguimiento manual, anotando en papel dónde se encuentran los productos o el material del que disponen. En algunos casos, esto se realiza en ordenadores, pero también teniendo que escribir de forma manual el tránsito de herramientas, equipamiento o incluso maquinaria. Esto hace que exista falta de cohesión en los datos y errores debidos a no tener una metodología de trabajo específica o simplemente a fallos humanos a la hora de introducir los datos, lo cual supone una gran pérdida de tiempo para localizar activos que se han “perdido”.

Para solucionar este problema, en este proyecto se ha realizado un sistema de seguimiento automático de activos en la fábrica de la empresa Maflow haciendo uso de la tecnología RFID. Para ello, tras un estudio y análisis previo, se monitorizan varios puntos de la fábrica, de modo que se controlen las distintas fases o estados en los que puede estar el material. Mediante una API se realiza la comunicación con el hardware lector de etiquetas y se extrae toda la información necesaria para posteriormente, poder manipularla y realizar los cálculos o ajustes necesarios. El sistema automáticamente anotará los datos correspondientes para poder realizar el seguimiento. Haciendo uso de etiquetas, se puede dar nombre y ajustar parámetros para identificar los activos de forma única. Se dispone de una interfaz gráfica en tiempo real para poder visualizar el tránsito, mapear etiquetas a nombres o modificar las ya existentes.

Palabras clave: activos, stock, metodología, tecnología RFID, API, lector de etiquetas, seguimiento, mapear.

ABSTRACT

The asset tracking in the industry is a generalized problema due to the large stock quantity and the wide workspace or storage that the assets can manage. Many companies choose to conduct a manual tracing, writing down on a paper where are the products or material they have. In some cases, this is done in computers, but also having to write down manually the transito f tools, equipement or even machinery. This makes it exist a lack of cohesion in the data and errors due to not having a specific work methodology or simply human failures when introducing the data, which suposes a big waste of time to locate “lost” assets.

To solve this problem, an automatic asset tracking system has been carried out in the factory of the company Maflow making use of RFID technology. For it, after a previous study and analysis, various points of the fabric are monitorized, so that the different phases or states in which the material can be are controlled. Through an API the communication with the tag reader is made and all the necessary information is extracted to later, being able to manipulate it and perform all calculations or necessary adjustments. The system will automatically annotate the corresponding data to being able to track the assets. Making use of tags, you can name and adjust parameters to uniquely identify the assets. A real-time graphic interface is available to view the traffic, map labels ton ames or modify existing ones.

Key words: assets, stock, methodology, RFID technology, API, tag reader, tracing, map.

ÍNDICE

1.	INTRODUCCIÓN	7
1.1	Motivación y contexto tecnológico	7
1.2	Problema y objetivo	8
1.2.1	Problema de utillajes	8
1.3	Desglose de objetivos	9
1.4	Estructura del documento	10
2.	ANÁLISIS DE REQUISITOS	11
2.1	Requisitos funcionales	11
2.2	Requisitos no funcionales	11
2.2.1	Accesibilidad	11
2.2.2	Rendimiento	11
2.2.3	Usabilidad	12
2.2.4	Seguridad	12
2.2.5	Disponibilidad	12
2.2.6	Tecnología RFID	12
2.3	Casos de uso	12
2.3.1	Diagramas de casos de uso	12
3.	HERRAMIENTAS Y TECNOLOGÍAS	15
3.1	Tecnología RFID	15
3.1.1	¿Qué es y cómo funciona?	15
3.1.2	Tipos según frecuencia	15
3.1.3	Tipos de etiquetas	16
3.2	Hardware	17
3.3	Software	21
4.	DESARROLLO DEL SOFTWARE	22
4.1	Arquitectura del Software	22
4.2	Capa de Persistencia	23
4.3	Capa de Negocio	23
4.4	Capa de Presentación	25
5.	PRUEBAS REALIZADAS	28
5.1	Consideraciones Técnicas	28
5.2	Pruebas en Oficina	29
5.3	Pruebas en Entorno Real	30

6.	CONCLUSIONES Y LINEAS FUTURAS	32
6.1	Ventajas e Inconvenientes	32
6.2	Propuesta Técnica	33
6.3	Conclusiones.....	34
	BIBLIOGRAFIA.....	36

ÍNDICE DE FIGURAS

Figura 1. Imagen del almacén con los utillajes metálicos	9
Figura 2. Representación gráfica de casos de uso	12
Figura 3. Antena RFID.....	17
Figura 4. Lector RFID	18
Figura 5. Kit de etiquetas RFID	19
Figura 6. Tag RFID Confidex Ironside.....	20
Figura 7. Tag RFID Metal Smartrac (azul)	20
Figura 8. Método init.....	24
Figura 9. Salida por consola.....	25
Figura 10. Método start que inicia la interfaz gráfica	26
Figura 11. Interfaz del programa.....	26
Figura 12. Comparativa de ganancia antena lineal vs polarización circular [12]	28
Figura 13. Utillaje metálico	30
Figura 14. Prueba vertical	30
Figura 15. Preparación para las pruebas.....	31
Figura 16. Interfaz de usuario final	33
Figura 17. Puntos de colocación de antenas en fábrica.....	34

1. INTRODUCCIÓN

En este capítulo se pretende dar una idea general del proyecto. Primero se muestra la motivación detrás de la realización de este trabajo. A continuación, se describen el problema a resolver y los objetivos que se quieren lograr. Por último, se detalla la estructura que va a seguir el documento.

1.1 Motivación y contexto tecnológico

Este trabajo ha sido desarrollado en la empresa SOINCON, empresa en la cual realicé mis prácticas curriculares. SOINCON es una empresa dedicada al desarrollo software con especial implicación en el ámbito de la industria 4.0. Trabaja con diferentes clientes construyendo aplicaciones para mejorar la productividad en los diversos ámbitos de trabajo que se pueden encontrar dentro de una industria.

Este trabajo surge dado que Maflow, una de las empresas con las que ya estábamos trabajando trasladó la necesidad de evaluar las tecnologías inalámbricas disponibles en el mercado.

En dicha empresa disponían de una serie de utillajes que se estaban moviendo constantemente por la fábrica y no tenían forma de controlar dónde estaba cada uno, con la consiguiente pérdida de productividad derivada de la búsqueda los utillajes por las distintas ubicaciones posibles. Además, se indicó la preferencia por utilizar una infraestructura que permitiera no solo posicionar los utillajes, sino también ser utilizada en un futuro en distintos ámbitos de la fábrica (posicionamiento y trayectorias de personal de fábrica, presencia en línea, ...).

Tratando de encontrar un ejemplo análogo al problema descrito para poder tener una referencia sobre como plantearlo encontré que, en las tiendas de ropa, para poder controlar la entrada y salida de prendas se dispone de unas etiquetas que se activan o desactivan en función de la necesidad del proveedor, haciendo que, si un cliente se lleva una prenda sin pagarla, al pasar por los arcos de salida de la tienda estos se activen produciendo una señal sonora. Esto es debido a que la etiqueta de la prenda dispone de tecnología RFID que permite prevenir robos y, por lo tanto, soluciona un gran problema dentro de este mercado.

Viendo la efectividad y sencillez de este sistema, se pensó en trasladar esto a la empresa, con el objetivo de controlar digitalmente la ubicación de los utillajes en fábrica de forma ágil y sencilla. Además, se indicó la preferencia por utilizar una infraestructura que no solo permita posicionar los utillajes, sino que también pueda ser utilizada en un futuro en distintos ámbitos de la fábrica (posicionamiento y trayectorias de personal de fábrica, presencia en línea, ...).

1.2 Problema y objetivo

El objetivo de este proyecto es construir un sistema que sirva como prototipo de aplicación de seguimiento de activos haciendo uso de la tecnología RFID y se aplicará a un caso concreto que es el control de utillajes metálicos dentro de una fábrica.

Además, se quiere que en un futuro este sistema pueda utilizarse en más ámbitos como por ejemplo las trayectorias de personal en fábrica o presencia en línea, como se ha mencionado antes. Como implementar todo eso sería un trabajo muy grande, este TFG está centrado en realizar una aplicación que, con ligeros cambios, pueda adaptarse a todas estas situaciones.

También se pretende ejemplificar el sistema prototipo a un caso concreto como es el tránsito de utillajes, viendo cómo sería su aplicación a un problema real, los inconvenientes que pueden surgir y las posibles soluciones que podemos adoptar para lograr el objetivo.

1.2.1 Problema de utillajes

Con el tránsito de utillajes surgen varios problemas a solucionar. En primer lugar, los utillajes en este caso son planchas metálicas utilizadas para la fabricación (concretamente moldeo) de piezas dentro del sector del automóvil. Estas piezas son bastante pesadas con tamaño y forma variantes. Además, son lavadas y ocasionalmente tratadas con algunos líquidos ligeramente corrosivos. Esto va a ser un gran problema a la hora de escoger las etiquetas a utilizar, ya que deben ir pegadas a las placas y soportar los líquidos mencionados previamente además de enviar suficiente señal pese a las grandes interferencias que puede generar el metal.

También hay utillajes que o bien no se utilizan o bien se utilizan de forma muy ocasional o bien no se utilizan por estar obsoletos o no fabricar actualmente ese tipo de pieza. Estos utillajes están situados mayoritariamente en unas baldas colocadas en un punto de la fábrica que actúan como "almacén". Esto es un problema ya que es necesario saber dónde están todos los utillajes y los que están situados en las esquinas o muy arriba del almacén pueden dar problemas para ser localizados porque no llegue la señal (con el agravante que genera el material metálico mencionado anteriormente). El almacén al que se hace referencia se muestra en la figura 1.



Figura 1. Imagen del almacén con los utillajes metálicos

1.3 Desglose de objetivos

El problema que queremos abarcar es bastante amplio y, por lo tanto, una vez vistos los inconvenientes derivados del uso de utillajes era conveniente desglosar el objetivo principal en objetivos más sencillos para llevarlos a cabo de una forma estructurada y con un orden lógico.

El orden seguido fue el siguiente:

- Obtener documentación acerca de la tecnología RFID (cómo funciona, frecuencias, tipos, etiquetas, etc...)
- Elección del hardware (lector a utilizar, tipo de etiquetas, antena, ...)
- Elección de etiquetas (dentro del tipo seleccionado, escoger la que mejor se ajusta a las necesidades del proyecto)
- Diseño del software (capas que es necesario implementar y distribución de clases)
- Construcción de la capa de negocio parte 1 (conexión entre el lector y el ordenador para poder transmitir datos)
- Construcción de la capa de negocio parte 2 (manipulación de los datos obtenidos previamente)
- Construcción de la capa de presentación (interfaz gráfica y muestra de datos manipulados en la capa de negocio)

1.4 Estructura del documento

Esta memoria está organizada en capítulos. En cada uno de ellos se van describiendo requisitos técnicos, información sobre la tecnología y cómo se han llevado a cabo las soluciones adoptadas para resolver el problema previamente planteado.

Los capítulos restantes son los siguientes:

- ✓ **Capítulo 2: Análisis de requisitos.** Se analizan los requisitos de la aplicación tanto funcionales como no funcionales.
- ✓ **Capítulo 3: Herramientas.** Se recogen las herramientas tanto hardware como software que se ha decidido utilizar durante el proyecto. Además, se detalla el funcionamiento de la tecnología RFID que se usa durante el proyecto.
- ✓ **Capítulo 4: Implementación del modelo desarrollado.** Se describe el programa realizado para satisfacer las necesidades previamente recogidas.
- ✓ **Capítulo 5: Pruebas realizadas.** Se detallan las pruebas realizadas tanto en oficina como en fábrica para comprobar el correcto funcionamiento del sistema.
- ✓ **Capítulo 6: Conclusiones y líneas futuras.** Se analiza la efectividad de la solución obtenida y las posibles mejoras del proyecto.

2. ANÁLISIS DE REQUISITOS

En este capítulo se analizan y exponen los requisitos más importantes, tanto funcionales como no funcionales. Esta parte es el comienzo de un proyecto software y tiene como objetivo dejar claro las necesidades que se pretenden satisfacer a la hora de la implementación final. [1]

2.1 Requisitos funcionales

En este apartado se exponen los requisitos funcionales que debe cumplir el sistema. Los requisitos funcionales definen cómo debe comportarse el software ante diferentes situaciones.

Identificador	Descripción
RF00	El sistema debe identificar unívocamente cada activo
RF01	Se mostrará información de un determinado activo si este es seleccionado
RF02	Se podrán guardar los nombres mapeados de etiquetas y cargar otros previamente guardados
RF03	Se podrá renombrar los activos, añadir nuevos o eliminar los ya existentes
RF04	La interfaz mostrará la intensidad de las etiquetas que están dentro de rango
RF05	Se podrá tener un histórico con datos pasados, para poder contrastar información
RF06	Será posible navegar por la aplicación sin necesidad de registrarse
RF07	Se podrá activar y desactivar la recepción de datos en cualquier momento

Tabla 1: Requisitos funcionales del sistema

2.2 Requisitos no funcionales

Los requisitos funcionales definen propiedades emergentes o restricciones del sistema como pueden ser la respuesta en el tiempo, la fiabilidad, o la disponibilidad. A continuación, se enumeran los más significativos en este proyecto.

2.2.1 Accesibilidad

El sistema será accesible a través de un módulo, que podrá usarse de forma individual como aplicación, o bien integrarse con otros sistemas ya sea en un proyecto más grande o en una web.

2.2.2 Rendimiento

La aplicación deberá satisfacer el RF04 en tiempo real, así como los datos actualizados en caso de solicitar información de un activo concreto. Los datos editados se deberán mostrar en el programa instantáneamente y quedar reflejados en el histórico.

2.2.3 Usabilidad

La aplicación interaccionará con los usuarios a través de una interfaz sencilla, con una terminología clara y concisa que describa unívocamente el propósito de cada acción, de forma que, cualquier usuario sin experiencia ni formación previa pueda visualizar y manejar el sistema.

2.2.4 Seguridad

El sistema exportará los datos de los utillajes (última vez visto, ubicación, etc) a la ubicación escogida por el usuario en un archivo con formato de texto, para evitar la pérdida de datos. Al no disponer de usuarios, no se requiere protección de datos de usuario.

2.2.5 Disponibilidad

El sistema deberá estar disponible las 24 horas del día monitorizando el tráfico de activos. En caso de caída del sistema o fallos, mediante los ficheros de texto se dispondrá del histórico de información que permitirá restaurar el sistema lo antes posible.

2.2.6 Tecnología RFID

Para la realización del sistema, se deberá usar tecnología RFID.

2.3 Casos de uso

En este apartado se detallarán los casos de uso más importantes para el desarrollo del sistema. En cuanto a los actores, no se va a detallar debido a que se trata de un programa “prototipo”. De esta forma, el único actor que tendríamos sería el de “administrador” aunque no va a necesitar de credenciales, y siempre se va a tener este rol.

2.3.1 Diagramas de casos de uso

Los diagramas de casos de uso definen las posibilidades de interacción entre el usuario y el sistema [1]. Se mostrarán en forma de tabla para facilitar su visualización y comprensión. Adicionalmente, la figura 2 muestra una representación gráfica de los principales casos de uso del sistema a desarrollar.

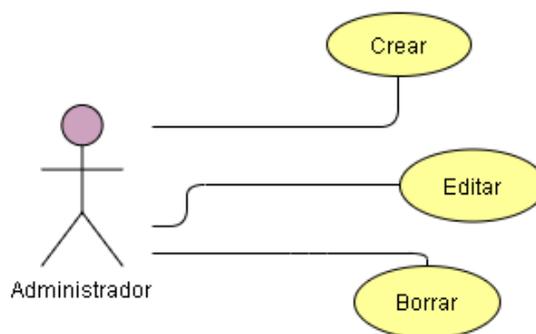


Figura 2. Representación gráfica de casos de uso

Nombre	Crear
Actor principal	Administrador
Actores secundarios	
Descripción	El usuario puede añadir un nuevo ítem asignándole un nombre y haciéndolo corresponder con un ID de etiqueta para facilitar su seguimiento.
Evento de activación	El usuario hace click en “Nuevo”
Precondición	No puede haber un mismo ítem asignado a varios nombres, ni tampoco un mismo nombre asignado a varios ítems, la relación debe ser “one to one”
Garantías de éxito	La pareja ítem-nombre queda registrada
Escenario principal	<ol style="list-style-type: none"> 1. La interfaz muestra varios campos a rellenar: El ID del ítem, y el nombre que se le quiere asignar. 2. Se rellenan los campos 3. Se aceptan los cambios 4. El sistema guarda los datos en memoria 5. La ventana desplegada desaparece
Extensiones	<p>3a. No se encuentra el ID introducido</p> <ol style="list-style-type: none"> 1. Se pide que se introduzca un ID válido <p>3b. El ID introducido ya ha sido asignado</p> <ol style="list-style-type: none"> 1. Se indica que ese ID ya ha sido asignado y que debe borrarse primero la asignación

Tabla 1. Caso de uso “crear”

Nombre	Borrar
Actor principal	Administrador
Actores secundarios	
Descripción	El usuario puede eliminar una asignación elemento-nombre
Evento de activación	El usuario hace click en “Borrar”
Precondición	Debe haber algún elemento asignado para poder borrarlo
Garantías de éxito	La pareja elemento-nombre queda eliminada
Escenario principal	<ol style="list-style-type: none"> 1. La interfaz muestra un campo a rellenar, el de nombre 2. Se rellena el campo 3. Se aceptan los cambios 4. La ventana emergente desaparece
Extensiones	<p>3a. No se encuentra el nombre introducido</p> <ol style="list-style-type: none"> 1. Se pide que se introduzca un nombre válido

Tabla 2. Caso de uso “borrar”

Rubén García González
SISTEMA DE SEGUIMIENTO DE ACTIVOS CON TECNOLOGÍA RFID

Nombre	Editar
Actor principal	Administrador
Actores secundarios	
Descripción	El usuario puede editar una asignación elemento-nombre
Evento de activación	El usuario hace click en "Editar"
Precondición	Debe haber algún elemento asignado para poder editarlo
Garantías de éxito	La pareja elemento-nombre queda reestablecida
Escenario principal	<ol style="list-style-type: none">1. La interfaz muestra los campos a rellenar: nombre, ID2. Se rellenan los campos3. Se aceptan los cambios4. La ventana emergente desaparece
Extensiones	<ol style="list-style-type: none">3a. No se encuentra el nombre introducido<ol style="list-style-type: none">1. Se pide que se introduzca un nombre asignado3.b El nombre nuevo ya ha sido asignado a otro elemento<ol style="list-style-type: none">1. Se pide que se introduzca otro nombre

Tabla 3. Caso de uso "editar"

3. HERRAMIENTAS Y TECNOLOGÍAS

A continuación, se presentan las herramientas tanto hardware como software utilizadas para llevar a cabo el proyecto. En primer lugar, se describe la tecnología RFID para explicar su funcionamiento.

3.1 Tecnología RFID

En este apartado se explica en detalle cómo funciona la tecnología RFID que ha sido la utilizada para llevar a cabo el desarrollo del proyecto. Se analizarán sus ventajas e inconvenientes y se explicará por qué se ha elegido esta opción.

3.1.1 ¿Qué es y cómo funciona?

La tecnología RFID, cuyas siglas corresponden a “**R**adio **F**requency **I**dentification”, es un sistema inalámbrico para comunicación entre objetos. Uno de los objetos emite señales de radio (con distintos tipos de frecuencia, como veremos a continuación) y el otro responde y actúa en función de la señal recibida. [3]

3.1.2 Tipos según frecuencia

Dentro del marco de la tecnología RFID, existen diferentes tipos dependiendo de la frecuencia a la que se trabaja. Actualmente hay tres tipos que podemos diferenciar, cada uno con unas características diferentes.

Baja Frecuencia (LF):

Su franja de funcionamiento está entre los 30 y los 300Khz. Las lecturas en este rango tienen un alcance corto, en torno a unos 10cm, con una velocidad de lectura generalmente lenta. Esta frecuencia proporciona una gran resistencia a las interferencias externas. Típicamente se utiliza en sistemas de control de animales o controles de acceso.

Alta Frecuencia (HF):

Corresponde a una franja de funcionamiento entre los 3 y los 30 Mhz (generalmente 13,56Mhz). Su rango de lectura tiene un alcance entre los 10 y los 100cm, los sistemas de este tipo tienen una velocidad de lectura superior a los de baja frecuencia. Las interferencias afectan de forma moderada. Se suele utilizar en pagos, gestión de eventos o aplicaciones con intercambio de datos.

Ultra-Alta Frecuencia (UHF):

La franja de funcionamiento entre los 300Mhz a 3Ghz (generalmente entre 900 y 915Mhz). El rango de lectura puede llegar a más de 12 metros. Los sistemas de estas características tienen una velocidad de lectura muy alta, pero son muy sensibles a las interferencias. Estos sistemas son más fáciles y económicos de fabricar. Su uso es muy variado, desde sistemas de inventarios en tiendas (tiendas de ropa, por ejemplo) hasta identificación de medicamentos, seguimiento de camiones o control de equipaje en aeropuertos, entre otras.

3.1.3 Tipos de etiquetas

Además de la frecuencia con la que se trabaje, los tags o etiquetas, que son los objetos que van a identificar de forma unívoca a cada uno de los activos, pueden ser de tres clases distintas dependiendo de su fuente de alimentación.

Activas:

Tienen fuente de alimentación interna, por lo tanto, tienen una disponibilidad de lectura continua y requieren de muy poca intensidad de señal por parte del lector para poder comunicarse. Gracias a esto, tienen un gran alcance de más de 100 metros, y una capacidad de monitorización continua.

Pasivas:

No tienen fuente de alimentación, funcionan por energía transferida desde el lector por radiofrecuencia y, por lo tanto, dependen de la intensidad de la señal del lector para poder comunicarse (necesitan una fuerte intensidad), y la fortaleza de la señal es bastante baja. Su alcance es de hasta 10 metros, pero solo permite leer y transmitir datos cuando la etiqueta recibe suficiente intensidad de señal por parte del lector.

Semi-Pasivas:

Utilizan una fuente de alimentación interna para encenderse, pero utilizan la energía emitida por el lector para comunicarse mediante retrodispersión. Por lo tanto, dependen también de la intensidad de la señal del lector para poder comunicarse, aunque la fortaleza de la señal es moderada. Su alcance es de hasta 100 metros, pero solo puede transmitir información cuando recibe señal del lector.

3.2 Hardware

Para el hardware, se contactó con la empresa **Dipole RFID** que suministró el material necesario, además de resolver algunas dudas referentes al uso de la tecnología (interferencias, pruebas posteriores, etc). En este caso se optó por un sistema RFID UHF, debido a que se buscaba un rango de lectura efectivo de en torno a 5 o 6 metros y unas velocidades de lectura altas. Además, actualmente las etiquetas de los sistemas UHF tienen una mayor resistencia a las interferencias debido al desarrollo tecnológico en los materiales utilizados para su fabricación, que las permiten adaptarse para trabajar en diferentes entornos (se profundiza más posteriormente). También, los sistemas UHF son mucho más baratos en coste y en un entorno industrial en el que lógicamente también se busca el beneficio económico es importante la rentabilidad.

El sistema consta de los siguientes dispositivos:

Antena RFID Mini Panel S8655P Laird:

Esta antena es la que nos va a permitir localizar las etiquetas y los activos siempre y cuando se encuentren dentro de su campo de acción. Hemos optado por una polarización circular debido a que nos aporta una mayor flexibilidad a la hora de su colocación y es más versátil a la hora de trabajar en diferentes entornos. La antena mencionada se muestra en la figura 3.

Características principales: [4]

- **Frecuencia:** ETSI (EU):865 - 870 MHz & FCC (US): 902 - 928 MHz
- **Ganancia:** 5,5 dBic
- **Polarización:** Circular a derecha e izquierda
- **Apertura:** 70°
- **Tamaño:**13,2 x 13,2 x 1,8 cm

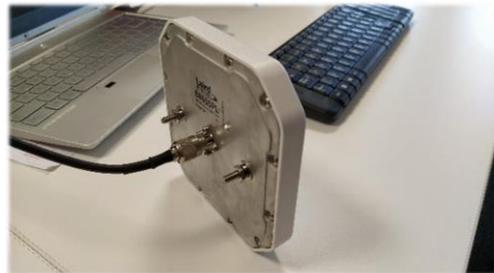


Figura 3. Antena RFID

Lector RFID ISC.MRU102-A:

Se trata de un lector UHF en caja con interface RS232 (asíncrono) con 3 conectores SMA para antenas UHF externas y con 1 antena de 50 Ohm integrada.

Este lector UHF ha sido escogido debido a su conexión por puerto serie RS232. Esto se debe a que los ordenadores industriales ya instalados en la empresa utilizaban este puerto y por lo tanto, era necesario para poder realizar la conexión. De este mismo modelo existen otras dos variantes que serían el de MRU102-USB y el MRU102-PoE(Power Over Ethernet) que con unos cambios mínimos en el software desarrollado se podrían usar en otros entornos.

Para poder conectar este lector al ordenador y realizar pruebas, se ha utilizado también un adaptador RS232 – USB. Además, este lector dispone de un cargador ya que debe estar conectado a la corriente para poder trabajar (se muestra el montaje completo usado para pruebas posteriormente). El lector descrito se muestra en la figura 4.

Características principales: [5]

- **Dimensiones (W x H x D):** 145 mm x 85 mm x 27 mm
- **Consumo de Potencia:** max 7 W
- **Potencia de Salida:** 50 mW a 500 mW, configurable
- **Conexión de antenas:** 3 x externas (SMA-Female - 50 Ω)
1 x antena integrada
- **Interfaz:** RS232-V24
- **Protocolo Software:** FEIG Reader Protocol
- **Modos de Protocolo:** ISO Host Mode, Buffered Read Mode, Scan Mode
- **Extras:** Monitorización de la Temperatura, RSSI
- **Rango Operativo de Temperatura:** -25°C a 55°C (12 V a 24 V)



Figura 4. Lector RFID

Etiquetas:

A la hora de escoger las etiquetas, se ha optado por unas **pasivas**, debido principalmente a dos factores. Por un lado, su coste unitario es mucho más bajo (entre 2 y 5 céntimos la unidad) frente a los altos costos de las etiquetas **activas** o **semi-pasivas** (entre 30 y 50 euros aproximadamente). Por otro lado, el hecho de tener baterías implica tener que renovarlas periódicamente, y esto es una problemática cuando queremos tenerlo todo automatizado y no estar pendientes de ello. Además, con las etiquetas pasivas ya podemos obtener la distancia que estamos buscando.

Dentro de las etiquetas pasivas hay muchos tipos dependiendo del proceso de fabricación (si se trata de etiquetas encapsuladas, el material con el que se han fabricado, etc.). Estas pueden ofrecer diferentes prestaciones en entornos exigentes (a la hora de trabajar con metales, con líquidos o con presión entre otros). Por lo tanto, el mismo proveedor citado anteriormente envió también un kit con múltiples tipos de etiquetas (véase la figura 5) para poder ir probando cuál se ajustaba mejor a las necesidades.



Figura 5. Kit de etiquetas RFID

Debido a que se quería trabajar sobre planchas metálicas, las cuales provocan muchas reflexiones, se optó por utilizar unas etiquetas encapsuladas. Concretamente, hubo dos tipos que funcionaron muy bien en este caso (se explican posteriormente las pruebas) y que son los que se muestran a continuación (figuras 6 y 7).

Tag RFID Confidex Ironside:

Características principales: [6]

- **Frecuencia:** 860 - 960 MHz
- **Tamaño Etiqueta:** 51,5 mm x 47,5 mm x 10 mm (2.03" x 1.87" x 0.39")
- **Temperatura de funcionamiento:** -40 °C a +85 °C / -40 °F a +185 °F
- **Rango de lectura (2W ERP)***
 - EU on metal up to 9 m / 30 ft
 - US on metal up to 7 m / 23 ft
 - JPN on metal up to 8m / 26 ft
 - Off metal read ranges 3-4 m / 10-13 ft



Figura 6. Tag RFID Confidex Ironside

Tag RFID Metal Smartrac MAXDURA OUTDOOR:

Características principales: [7]

- **Frecuencia:** 865 - 868 MHz (ETSI) & 902-928 MHz (FCC)
- **Tamaño Etiqueta:** 90 mm x 34 mm x 7 mm
- **Temperatura de funcionamiento:** -40 °C a +85 °C / -40 °F a +185 °F
- **Rango de lectura (2W ERP)**
 - EU on metal up to 9 m / 30 ft
 - US on metal up to 7 m / 23 ft
 - JPN on metal up to 8m / 26 ft
 - off metal read ranges 3-4 m / 10-13 ft



Figura 7. Tag RFID Metal Smartrac (azul)

3.3 Software

Se presentan a continuación las herramientas software que han sido utilizadas en el desarrollo del proyecto.

Java:

Java es un lenguaje de programación orientado a objetos [8]. Ha sido el escogido para la realización de este trabajo debido a su flexibilidad y posibilidad de escalado. Todas las clases del proyecto han sido realizadas utilizando este lenguaje.

Eclipse:

Eclipse es la plataforma software utilizada para desarrollar el código [9]. Se han utilizado varias versiones de Eclipse a lo largo del proyecto y ha sido escogido debido a que se trata de un software gratuito y su facilidad de uso tanto para el desarrollo como para la realización de pruebas.

JavaFX:

JavaFX es una familia de productos que permite crear aplicaciones gráficas de escritorio [10]. Está integrado con Java y ha sido escogido debido a que la integración era sencilla y permite darle un formato visual a los datos con los que estaba trabajando.

Se ha utilizado la versión 14 de JavaFX (paquete javafx). También se ha utilizado el conjunto de plugins de e(fx)clipse 3.6.0 descargable desde dentro de Eclipse (Eclipse Marketplace).

JavaFX Scene Builder 2.0:

JavaFX Scene Builder es una herramienta de diseño visual que permite crear interfaces gráficas [11]. Esta herramienta ha sido utilizada para realizar la creación de las interfaces de una forma visual e intuitiva, mejorando la fluidez a la hora de integrar los elementos en el diseño. En este caso se ha utilizado la versión 2.0

CSS:

CSS (Cascading Style Sheet) es un lenguaje que permite definir hojas de estilo para presentar información [12]. Esta herramienta ha sido utilizada para definir diferentes formas de mostrar el texto y cambiar los colores de la interfaz.

Visual Paradigm:

Visual Paradigm es una aplicación online que permite elaborar diferentes diagramas de forma sencilla [13]. Ha sido utilizado para la realización de la representación gráfica de casos de uso.

4. DESARROLLO DEL SOFTWARE

En este capítulo se detalla el desarrollo del proyecto, así como las diferentes capas de la implementación y la lógica que reside detrás de cada una.

4.1 Arquitectura del Software

Para este proyecto se ha utilizado la metodología SCRUM [15], que es una metodología ágil para el desarrollo software. Para ello, se ha optado por realizar sprints de 1 semana de duración, con una reunión cada lunes para evaluar el estado de proyecto y tomar decisiones.

Esta reunión se llevaba a cabo con el jefe de departamento de la empresa, que ha sido a su vez el tutor, y quien me ha proporcionado feedback durante la implementación del proyecto.

Periódicamente (cada 2 o 3 semanas aproximadamente) nos reuníamos también con el **product owner**, que era el directivo de la empresa que se encargaba de gestionar los utillajes y el tránsito por la fábrica para que él también nos diese una valoración del estado del proyecto y las funcionalidades a implementar.

El proyecto tuvo una duración aproximada de 12 semanas durante las cuales, a principio de cada una, se escogían las funcionalidades a implementar y se proporcionaba feedback el cual podía verse reflejado en forma de modificaciones para futuros sprints.

El orden de implementación fue el mismo que el seguido en este documento a la hora de describir las capas. Es decir, en los primeros sprints (5 semanas aprox) se fueron implementando funcionalidades de la capa de negocioreferentes a la conexión entre el software y el hardware lector de etiquetas, protocolo de comunicación, velocidad de transferencia, etc.

Posteriormente se implementaron funcionalidades también de la capa de negocio, pero esta vez referentes a cómo se iban a estructurar los datos o qué información era importante y cómo debía ser procesada para que sea fácil de entender para el usuario final (algoritmos a utilizar para los cálculos, normalización de datos, medias, etc.). Esta etapa duró aproximadamente 4 sprints.

Por último y con una duración de 3 sprints, se realizaron las funcionalidades de la capa de presentación. Se construyó la interfaz gráfica, se fueron corrigiendo opciones como por ejemplo poder editar el nombre de las etiquetas, los colores, borrado, etc., y se presentaron los datos de una forma visual y entendible para el usuario.

Para la realización del software se ha utilizado una arquitectura en tres capas basada en el patrón MVC (Modelo - Vista - Controlador) [1] donde:

- ✓ La parte del modelo es la que realiza la conexión con el lector y se mantiene transfiriendo datos. Además, se incluye un paquete Útil para realizar conversiones de fechas extraídas del lector y poder operar con ellas
- ✓ La vista corresponde a la parte de JavaFX que es la que ha permitido visualizar los datos previamente recogidos y presentarlos de una forma clara y sencilla.
- ✓ En la parte de controlador tenemos clases específicas que ajustan parámetros de configuración y la clase principal que es la que inicia la aplicación.

Este modelo permite mayor flexibilidad y hace que el programa sea modular lo cual facilita su integración en otros proyectos. Cada capa utilizará los elementos generados por las capas adyacentes, de esta forma, si fuese necesario hacer modificaciones solo haría falta cambiar el comportamiento de una de las capas y no se afectaría al resto.

Concretamente se han utilizado tres capas: datos, negocio y presentación, cuyo contenido se detalla a continuación.

4.2 Capa de Persistencia

Al tratarse de una aplicación prototipo, para la capa de persistencia se ha optado por un sistema de almacenamiento de información más sencillo que una base de datos.

En este caso se ha recurrido almacenamiento de datos en archivos con extensión “csv”. Este almacenamiento se hace de forma doblemente periódica. Es decir, en el programa se puede escoger con qué frecuencia se escriben datos en el fichero. Además, también se puede cambiar la frecuencia con la que se cierra un fichero y se genera otro nuevo. Por ejemplo, se puede configurar para que se cree un fichero nuevo cada 30 minutos, y que en cada fichero se estén escribiendo datos cada 5 segundos.

De esta forma se puede disponer de un histórico de datos y cada fichero se crea con el nombre de la fecha y hora en la que fue creado. En los ficheros se puede guardar toda la información extraída de las etiquetas (intensidad de lectura, antena que la leyó, fecha y hora de lectura, ID de la etiqueta y nombre asignado) siendo esto configurable en el caso de que solo se quieran guardar determinados campos.

4.3 Capa de Negocio

Esta capa corresponde a la conexión con el dispositivo lector de etiquetas, así como el procesado de estas y el uso de la API suministrada por el proveedor para el manejo del hardware.

Para la conexión con el dispositivo, se utiliza un método específico para el tipo de conexión que se esté realizando. En nuestro caso, hemos utilizado una conexión por puerto serie RS-232 para poder conectarnos posteriormente a ordenadores industriales y poder instalar el sistema en un entorno real. Cabe destacar que el ordenador en el que se estaba desarrollando el software no disponía de puerto serie RS-232 y por lo tanto fue necesario un convertidor de puerto serie a USB para poder conectar el lector. El método usado para iniciar la conexión con el lector y empezar a recibir datos se puede ver en la figura 8.

```
1.  /*
2.   * Metodo que inicia la configuracion del reader y empieza a recibir datos
3.   */
4.  public void init(RFID_ClientConf config) throws RFID_Exception {
5.      Instant actual;
6.      _conf = config;
7.      RFID_Config conf = new RFID_Config();
8.      conf.setCom(3); // Puerto COM en el que se encuentra el lector
9.      conf.setFrecuency(1); // Frecuencia (en segundos) a la que muestrea el dispositivo (no acepta
10.     // decimales, 100ms de error)
11.     conf.setTolerance(200); // La propia tolerancia del dispositivo es de -55dBm
12.     conf.setPort(1500); // Puerto sobre el que se va a operar (a partir del 1024)
13.
14.     // Se anhaden las antenas que se vayan a utilizar
15.     // NO ANHADIR LA INTERNA DEL LECTOR (4), BUG DE FUNCIONAMIENTO
16.     conf.addAerial(3);
17.     actual = Instant.now();
18.
19.     _reader = new RFID_Reader();
20.     _reader.setup(conf);
21.     _reader.init();
22.     _reader.start();
23.     _reader.addRFIDReadListener((RFID_ReadEvent arg) -> {
24.         /*
25.          * Operaciones que se realizan con los datos cuando se reciben
26.          */
27.         System.out.println(Instant.now() + " -> " + arg.returnLst().size());
28.         for (List<RFID_Sample> samples : arg.returnLst().values())
29.             for (RFID_Sample sample : samples) {
30.                 rafaga(sample);
31.                 System.out.println("    " + sample);
32.             }
33.         if (Instant.now().isAfter(actual.plusSeconds(_conf.getWait()))) {
34.             actualiza();
35.         }
36.     });
37. }
38. }
```

Figura 8. Método init

En este método podemos ver como primero se configuran manualmente algunos parámetros para la realización de pruebas como pueden ser el puerto RS232, la frecuencia de muestreo, la tolerancia del dispositivo (no es necesario indicarlo), el puerto para la transferencia de datos y las antenas que se van a utilizar.

Una vez conectado podemos leer el contenido del lector a través de una serie de objetos proporcionados por la API. Estos objetos corresponden a listas donde se almacena la información de cada etiqueta. De esta forma se pueden leer los datos correspondientes a las etiquetas que estén dentro del rango del lector en ese instante.

Posteriormente, mediante las funciones que se llaman con el objeto “_reader”, que corresponde al lector, se inician las comunicaciones y se empiezan a transmitir datos mediante el último bucle que encontramos en esta parte del código.

Esta acción solo sirve para cada instante, por lo tanto, es necesario tener este método ejecutándose de forma continua para poder tener un registro completo de lo que se está leyendo. Para ello se guarda la información en objetos, que contienen todos los datos referentes a una etiqueta.

Esto quiere decir que, una vez leída una etiqueta, si esta es nueva se crea un objeto que almacena tanto su ID como su nombre en caso de tenerlo. Además, se crean listas que contendrán los instantes en los que ha sido leída la etiqueta, la intensidad de la señal recibida en ese instante y un número correspondiente a la antena que lo ha leído. En el caso de que la etiqueta fuese leída por varias antenas, se crearía una lista por cada antena. De esta forma logramos obtener un registro de lo que se está leyendo no solo en un instante determinado, si no en un rango de tiempo que podemos configurar (podemos pedir por ejemplo que nos muestre la media de los últimos 10 segundos).

Podemos ir viendo la ejecución del programa y los resultados que se van mostrando por consola en la figura 9. Esto se hizo previamente a la interfaz para poder depurar el código de una forma más efectiva. Esta salida por consola se ha mantenido finalmente para facilitar la detección de bugs dentro del programa. Se pueden ver los datos brutos recibidos por el lector que van a ser posteriormente representados gráficamente.

```
2020-02-26T11:00:25.837Z -> 3
3 E2886D1200000021EEB7D74 2020-02-26T11:00:24.800Z -31
3 E2886D1200000021EEB7D74 2020-02-26T11:00:25.137Z -30
3 E2886D1200000021EEB7D74 2020-02-26T11:00:25.506Z -30
3 E2881170000020983986928 2020-02-26T11:00:24.800Z -37
3 E2881170000020983986928 2020-02-26T11:00:25.137Z -36
3 E2881170000020983986928 2020-02-26T11:00:25.506Z -37
3 E2881170000020983986936 2020-02-26T11:00:24.800Z -37
3 E2881170000020983986936 2020-02-26T11:00:25.137Z -37
3 E2881170000020983986936 2020-02-26T11:00:25.506Z -35
2020-02-26T11:00:26.844Z -> 3
3 E2886D1200000021EEB7D74 2020-02-26T11:00:25.838Z -30
3 E2886D1200000021EEB7D74 2020-02-26T11:00:26.174Z -29
3 E2886D1200000021EEB7D74 2020-02-26T11:00:26.514Z -30
3 E2881170000020983986928 2020-02-26T11:00:25.838Z -36
3 E2881170000020983986928 2020-02-26T11:00:26.174Z -37
3 E2881170000020983986928 2020-02-26T11:00:26.514Z -39
3 E2881170000020983986936 2020-02-26T11:00:25.838Z -35
3 E2881170000020983986936 2020-02-26T11:00:26.174Z -37
3 E2881170000020983986936 2020-02-26T11:00:26.514Z -36
2020-02-26T11:00:27.943Z -> 3
3 E2886D1200000021EEB7D74 2020-02-26T11:00:26.844Z -30
3 E2886D1200000021EEB7D74 2020-02-26T11:00:27.167Z -30
3 E2886D1200000021EEB7D74 2020-02-26T11:00:27.490Z -32
3 E2886D1200000021EEB7D74 2020-02-26T11:00:27.814Z -30
3 E2881170000020983986928 2020-02-26T11:00:26.844Z -37
3 E2881170000020983986928 2020-02-26T11:00:27.167Z -37
3 E2881170000020983986928 2020-02-26T11:00:27.490Z -37
3 E2881170000020983986928 2020-02-26T11:00:27.814Z -30
3 E2881170000020983986936 2020-02-26T11:00:26.844Z -31
3 E2881170000020983986936 2020-02-26T11:00:27.167Z -31
3 E2881170000020983986936 2020-02-26T11:00:27.490Z -34
3 E2881170000020983986936 2020-02-26T11:00:27.814Z -33
```

Figura 9. Salida por consola

4.4 Capa de Presentación

Esta capa contiene los elementos necesarios para presentar la información de una forma visual y clara para que el usuario pueda ver lo que está sucediendo en el programa.

Para facilitar la lectura de información al usuario, esta capa interactúa de forma transparente con la capa de negocio permitiendo que la información se vaya actualizando y refrescando automáticamente. El método que inicia la interfaz gráfica se puede ver en la figura 10.

```
1.  @Override
2.  public void start(Stage primaryStage) {
3.      this.primaryStage = primaryStage;
4.      this.primaryStage.setTitle("RFID Demo");
5.
6.      // Set the application icon.
7.      this.primaryStage.getIcons().add(new Image("file:resources/images/rfid-02-512.png"));
8.
9.      Platform.runLater(() -> {
10.         initRootLayout();
11.         showTagOverview();
12.
13.         Timer timer = new Timer (500, (ActionEvent e) -> {
14.             _controller.refresh();
15.         });
16.         timer.start();
17.     });
18. }
```

Figura 10. Método start que inicia la interfaz gráfica

Este método inicia el panel principal mediante el método **"initRootLayout()"**, que a su vez inicia el resto de paneles y asigna la hoja de estilo. Además, inicia el timer que va a ir refrescando la información de la interfaz periódicamente.

Una vez abierta la interfaz, la pantalla principal se encuentran dos paneles. El primero, situado a la izquierda, muestra todas las etiquetas que se están leyendo actualmente. El panel derecho aparecerá vacío por defecto, pero si hacemos click sobre una etiqueta del panel izquierdo, mostrará toda la información relativa a esta.

En la figura 11, se muestra una captura del programa en ejecución donde se pueden ver varios campos con una barra de progreso que representa la intensidad de la señal que está recibiendo el tag en cada momento. Se muestra también su nombre (o *default* en el caso de que el tag no esté asociado a ningún nombre), el ID (único) del tag, y la antena/s por la que está siendo leído. A un lado se muestran los detalles del tag seleccionado (nombre representativo, ID, media de los dB, fecha de la última vez que ha sido detectado por una antena y el número/nombre de la antena que lo ha detectado).

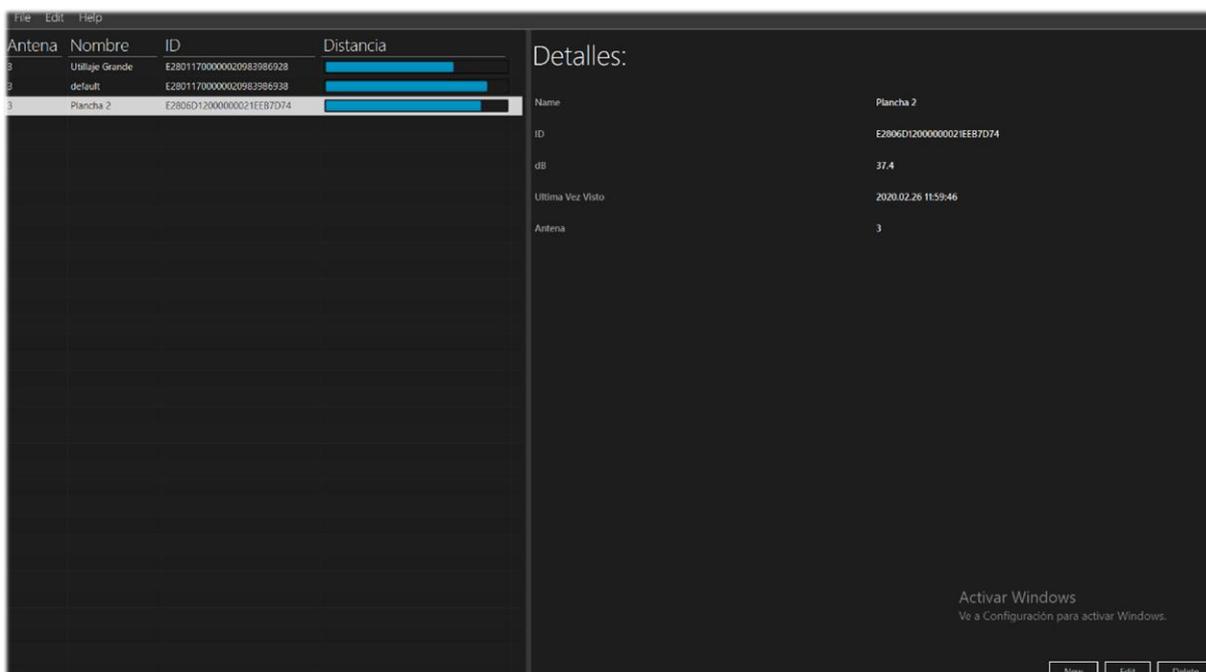


Figura 11. Interfaz del programa

En la parte inferior derecha se muestran 3 botones:

- New: Permite añadir una nueva etiqueta manualmente y asignarle un nombre
- Edit: Seleccionando previamente una etiqueta, permite modificar el nombre de esta o asignarle uno si no lo tiene.
- Delete: Borra la etiqueta seleccionada de la lista, por ejemplo, si quieres que una etiqueta concreta deje de aparecer gráficamente.

Los tres botones tienen las opciones de cancelar y aceptar y están programados para indicar al usuario en caso de fallo. Le indican al usuario si debe seleccionar una etiqueta en caso de que no lo haya hecho, si el ID que se quiere añadir o modificar no ha sido encontrado, o si el nombre que quiere asignar a una etiqueta ya está en uso por otra.

Arriba se muestran también 3 opciones que son:

- File: Permite abrir un fichero “.csv” de configuración con parejas de nombre-ID.
- Edit: En este caso podría servir para editar aspectos de la interfaz, cómo se muestran las cosas, etc. No ha sido implementado funcionalmente.
- Help: Mostraría información del programa, al ser un prototipo, tampoco ha sido implementada su funcionalidad.

5. PRUEBAS REALIZADAS

5.1 Consideraciones Técnicas

Hay varios aspectos que debemos tener en cuenta al utilizar tecnología RFID; uno de ellos es la **atenuación de la señal en el espacio libre**. Esta fórmula [16], aunque corresponde a unas condiciones “ideales”, nos puede dar una aproximación a los resultados que vamos a obtener en las pruebas:

$$L_{bf} = 20 \log\left(\frac{4\pi d}{\lambda}\right)$$

donde:

- ✓ L_{bf} : pérdida básica de transmisión en el espacio libre (dB).
- ✓ d : distancia (en nuestro caso, un máximo de 4m).
- ✓ λ : longitud de onda (aproximadamente 900MHz).
- ✓ d y λ se expresan en las mismas unidades.

Hay que tener en cuenta también la direccionalidad de la antena que estamos utilizando, en nuestro caso, es una antena de polarización circular a izquierda y derecha.

En la figura 12 se muestra una gráfica comparando la ganancia de una antena lineal frente a una antena con polarización circular en cuanto al ángulo de posicionamiento. La ganancia en la zona de operación de la antena lineal (roja) es mayor (permite alcanzar mayor distancia), pero la antena circular permite que en determinados ángulos la lectura sea posible (mientras que la lineal se perdería la señal por completo).

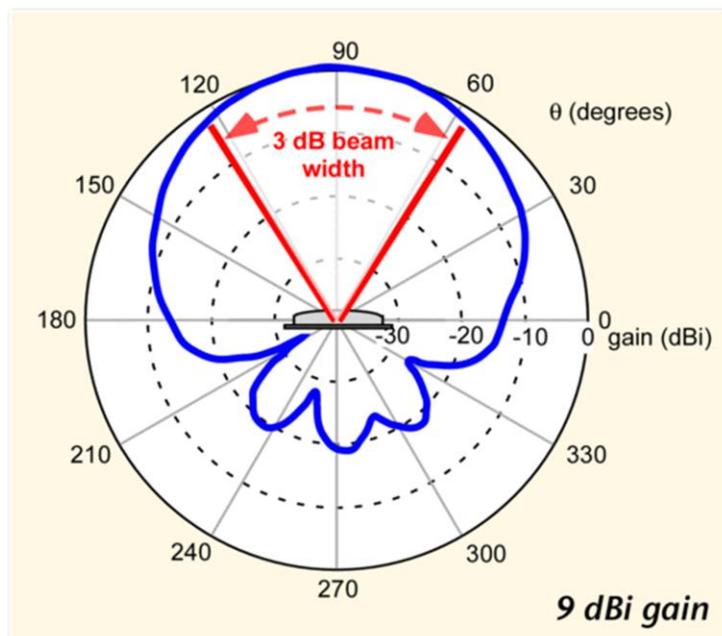


Figura 12. Comparativa de ganancia antena lineal vs polarización circular [12]

También es muy importante tener en cuenta que las etiquetas van a ser colocadas sobre utillajes que cuentan con planchas de metal de gran espesor y con peso, tamaño y formas variables. Esto hace que las pérdidas de señal recibida sean prácticamente incalculables (pese a que las etiquetas están preparadas para operar sobre metales y eso favorece su rendimiento, no deja de ser un factor limitante).

5.2 Pruebas en Oficina

Previamente a la instalación y prueba de los equipos y el software en fábrica, se han realizado diversas pruebas en las instalaciones de SOINCON, con el fin de disponer de una primera aproximación en un entorno no industrial. Las pruebas se han realizado tanto sobre superficies no metálicas, como sobre un utillaje piloto. Las pruebas realizadas en las instalaciones de SOINCON han sido las siguientes:

- ✓ **Prueba vertical.** Se ha colocado la antena en el techo de la oficina, con una altura de aproximadamente dos metros y medio, y se han colocados los distintos tags en el suelo. Posteriormente, y con el fin de comprobar la direccionalidad de la antena, se han ido desplazando las etiquetas hasta quedar fuera del rango de lectura de la antena. En este escenario, las lecturas se han realizado correctamente tanto con el utillaje fijo como en movimiento.

También se ha comprobado que la antena no es completamente direccional, de forma tal que, ante un paso relativamente rápido del utillaje bajo el arco de la antena, esta debería de ser capaz de leer la etiqueta con cierto margen de distancia (entre uno y tres metros, dependiendo de la altura).

- ✓ **Prueba horizontal.** Se ha analizado el mismo escenario que en las pruebas anteriores, pero colocando antena y etiquetas en posición horizontal. Los resultados arrojados son similares a los obtenidos en el escenario vertical.
- ✓ **Prueba con obstáculo.** Para simular la existencia de obstáculos, se ha empleado un tabique vertical y se ha colocado la antena en horizontal, con las etiquetas al otro lado del tabique. En este caso, el obstáculo ha impedido la correcta lectura de la etiqueta.
- ✓ **Prueba sobre utillaje (vertical).** La última prueba realizada es la más próxima al escenario previsto en las instalaciones de Maflow. Se han colocado las distintas etiquetas sobre un utillaje proporcionado por Maflow (figura 13), y se ha colocado la antena de nuevo en el techo de la oficina. Los resultados arrojados son ligeramente mejores que los obtenidos con la primera prueba (prueba vertical), habiéndose realizado lecturas con similar o superior intensidad.

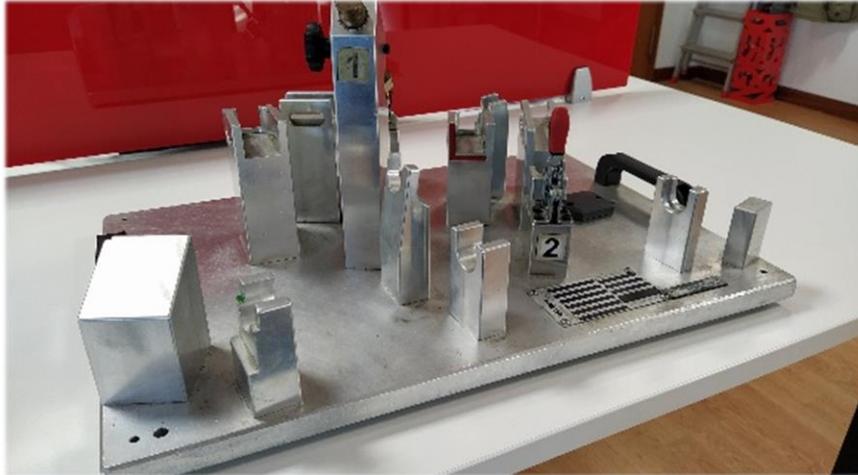


Figura 13. Utillaje metálico

5.3 Pruebas en Entorno Real

En las instalaciones de Maflow, se han realizado diversas pruebas para verificar si los resultados obtenidos en un entorno de pruebas son equivalentes a los resultados sobre el terreno, verificando la efectividad de este sistema en diferentes puntos clave para controlar la entrada/salida de utillaje.

La ubicación seleccionada para la realización de las pruebas en la fábrica (pasillo junto a almacén de utillajes) responde a la criticidad de esta: se trata de una de las zonas donde mayor tránsito de utillajes existe, siendo un punto clave a controlar.

Las pruebas realizadas se han desarrollado intentando emular una situación real del tránsito del utillaje en la fábrica, de forma tal que desde la posición de la antena (sobre la viga metálica / arco) se pudiesen leer los tags al cruzar el arco el utillaje:

- ✓ **Prueba vertical.** En primer lugar, se ha colocado la antena lectora sobre una viga metálica situada a unos 2,8 metros del suelo. La antena se ha orientado hacia el suelo debido a que los utillajes deben pasar por debajo de esta viga metálica. Para comprobar las lecturas y su intensidad, se ha situado un utillaje de prueba justo debajo de la antena, y se han colocado las distintas etiquetas sobre el utillaje (figura 14). Los resultados obtenidos ofrecen lecturas de en torno al 40% de la señal de forma constante, suficiente para obtener los datos satisfactoriamente.



Figura 14. Prueba vertical

- ✓ **Prueba vertical con movimiento.** A continuación, se han utilizado varios utillajes (cada uno con un tag RFID) y se ha procedido a realizar una simulación de tránsito de utillajes por debajo del arco, con una velocidad aproximada de 5 kilómetros por hora (paso ligero) y una frecuencia de lectura de 1 milisegundo (figura 15). En este caso, los resultados obtenidos han sido similares a los de la prueba vertical (40%) al pasar los utillajes justo por debajo del arco, pero sensiblemente peores (en torno al 20% de señal y de forma intermitente) en el rango de ± 20 grados con respecto a la vertical.
- ✓ **Análisis horizontal.** Para cerrar el análisis, volvimos a realizar la prueba de dirección horizontal, comprobando de nuevo que, debido a las características físicas del utillaje y la variedad de utillajes existentes, no es posible registrar de forma fidedigna el paso de estos por las zonas susceptibles de monitorización.



Figura 15. Preparación para las pruebas

6. CONCLUSIONES Y LINEAS FUTURAS

En este apartado se describe el cumplimiento de los objetivos fijados y la utilidad de las soluciones obtenidas. Además, se proponen mejoras para el sistema y posibles aplicaciones posteriores.

6.1 Ventajas e Inconvenientes

A continuación, se detallan las ventajas e inconvenientes del sistema que se han ido observando durante el desarrollo de las pruebas mencionadas anteriormente. Esto nos permite valorar de una forma más objetiva la efectividad del sistema desarrollado sobre el problema planteado al principio de este TFG.

Puntos a favor:

- Las etiquetas permiten la monitorización sobre utillajes metálicos con un rango de lectura de hasta 3 metros.
- El uso de las antenas permite suficiente flexibilidad ya que no son completamente direccionales y permite la monitorización de dos puntos (que están uno enfrente de otro) con la misma antena.
- Las etiquetas son suficientemente resistentes como para aguantar entornos difíciles.
- Al emplear etiquetas pasivas, no requieren de baterías de alimentación ni revisiones periódicas.
- Las etiquetas tienen un coste bajo en comparación con las alternativas existentes (entre 20 y 50 céntimos por etiqueta).

Puntos en contra:

- Es preciso instalar multitud de lectores por la distancia entre los distintos puntos de monitorización.
- El alcance depende de la orientación y la direccionalidad de la antena.

6.2 Propuesta Técnica

A continuación, se muestra la propuesta realizada para la implantación del sistema en un entorno real. Esta propuesta fue enviada como parte de un informe para la aceptación del sistema como proyecto piloto para la empresa Maflow (informe en el cual yo trabajé y realicé la parte referente a la tecnología RFID y su implantación).

Se propone la colocación de un conjunto de antenas en los puntos a monitorizar de la fábrica, pegando diferentes etiquetas RFID en los utillajes susceptibles de monitorización junto con un software de captura de datos que realice las siguientes operaciones:

- ✓ Captura de los datos del tránsito de utillajes y almacenamiento de información en base de datos.
- ✓ Procesado de información y aplicación de lógica de cálculo de trayectorias según potencia de la señal recibida, determinando posición inicial y final del utillaje.
- ✓ Tratamiento estadístico del dato.

El software de captura y tratamiento de datos se integraría con el software desarrollado previamente, complementando la funcionalidad de este.

En la figura 16 se muestra una captura de pantalla del software de control de intercambio de utillajes actual (Este software **NO** ha sido realizado por mí pero se muestra como ejemplo de integración del sistema desarrollado en un proyecto mayor):

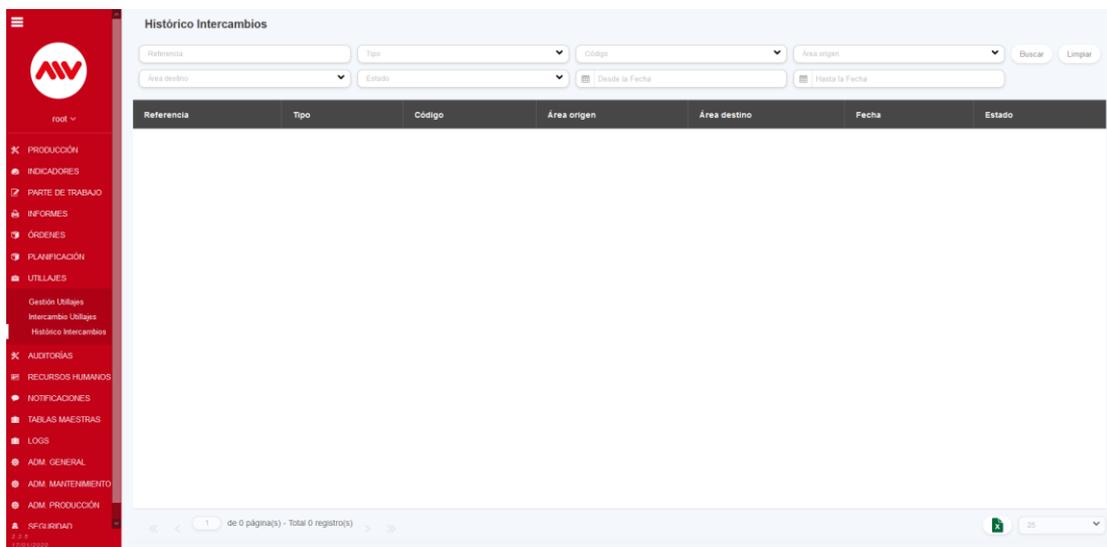


Figura 16. Interfaz de usuario final

En la figura 17 se muestran las ubicaciones donde sería necesario situar antenas.

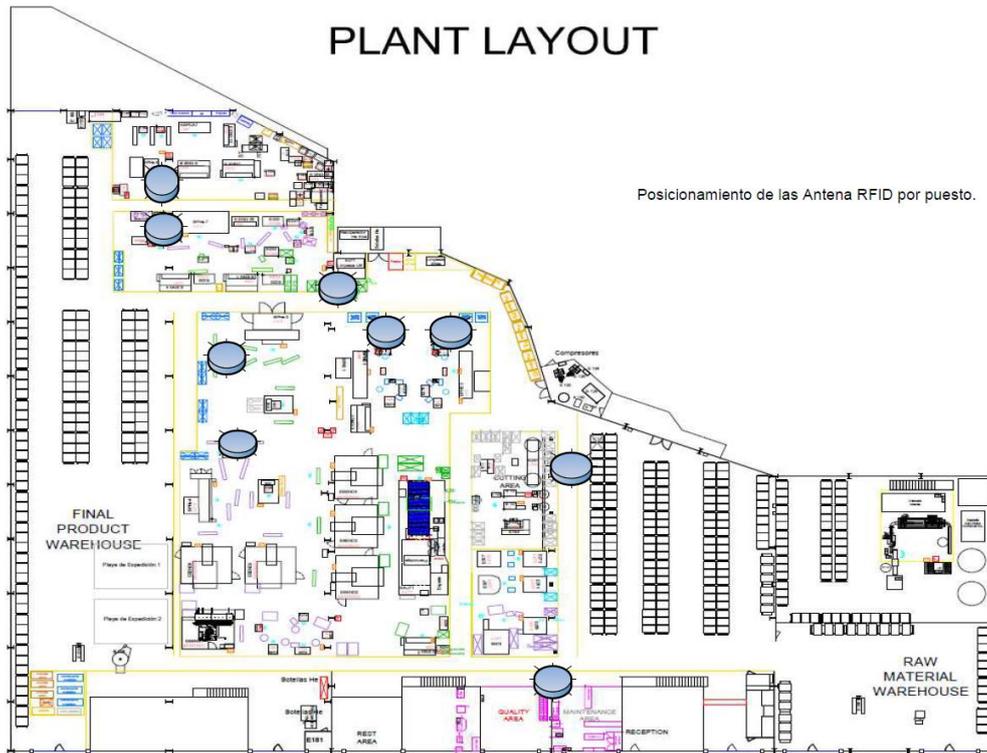


Figura 17. Puntos de colocación de antenas en fábrica

Cada una de las antenas deberá ir asociada a un dispositivo receptor. Un único dispositivo receptor puede procesar hasta tres antenas de forma simultánea.

6.3 Conclusiones

La tecnología RFID permite la trazabilidad de activos incluso en entornos muy exigentes. Esta característica, junto a su bajo coste, hace que su integración en otros sistemas sea sencilla y asequible.

Con el prototipo realizado podemos ver cómo se puede aprovechar su uso en un entorno real y de esta forma mejorar el flujo de trabajo dentro de una empresa, reduciendo drásticamente el tiempo empleado en localizar los activos y mejorando por tanto la productividad.

El uso de la tecnología RFID puede ser aplicable a muchos campos además de fábricas como por ejemplo cadenas de montaje, envasado de productos por lotes o control de stock entre otros.

La flexibilidad del programa permite minimizar los cambios necesarios para adaptarlo a cualquier sistema. Su diseño por capas hace que las modificaciones en una de las capas no afecten al resto.

Viendo estos puntos, se ha conseguido el objetivo marcado de realizar una aplicación funcional que pueda operar en un entorno real y permitir el seguimiento de activos de forma correcta. Aun así, cabe destacar algunos puntos a mejorar para que la aplicación sea más completa.

- Se podría realizar una conexión con base de datos para almacenar y obtener datos de ella y no tener que estar pendientes de ficheros que se pueden perder, mejorando la consistencia y tolerancia a fallos de la aplicación.
- Un sistema de usuarios mejoraría la seguridad del programa. Esto es algo que yo no he tenido en cuenta, pero podríamos conseguir acceso restringido para ciertas personas mediante el uso de usuarios y contraseñas.
- Se debería de mejorar la interfaz gráfica para que parezca más profesional.
- Podría ser interesante implementar mapas de calor por zonas dependiendo del tráfico de activos en cada una de ellas, además de un mapa donde se pueda visualizar gráficamente dónde se encuentra cada uno de los activos, además de un buscador por nombre que permitiese localizar en el mapa uno concreto.

BIBLIOGRAFIA

- [1] Ian Sommerville, *Ingeniería del Software*, Pearson Educación Mexico 2011.
- [2] Alicia Durango, *Diseño de Software*, IT Campus Academy 2014.
- [3] DIPOLE RFID. Tecnología RFID [sitio web]. [Consulta: 14 mayo 2020]. Disponible en: <https://www.dipolerfid.es/tecnologia-RFID>
- [4] Laird Technologies, Antena RFID Mini Panel S8655P Laird. [sitio web]. [Consulta: 17 mayo 2020]. Disponible en: https://www.dipolerfid.es/files/product/pdf_es/27/Laird%20Cp%20mini%20panel%20IP67.pdf
- [5] Feig Electronic, Lector RFID ISC.MRU102-A. [sitio web]. [Consulta: 17 mayo 2020]. Disponible en: https://www.feig.de/fileadmin/user_upload/Downloads/Datenblaetter/FEIG ELECTRONIC Datasheet IDISCMRU102_Egypt.pdf
- [6] Confidex, Tag RFID Confidex Ironside. [sitio web]. [Consulta: 18 mayo 2020]. Disponible en: https://www.dipolerfid.es/files/product/pdf_es/252/Ironside_Datasheet.pdf
- [7] Smartrac, Tag RFID Metal Smartrac MAXDURA OUTDOOR. [sitio web]. [Consulta: 18 mayo 2020]. Disponible en: https://www.dipolerfid.es/files/product/pdf_es/190/0095_SMARTRAC_MAXDURA_OUTDOOR.pdf
- [8] Oracle, JAVA. 2020. [sitio web]. [consulta 23 mayo 2020]. Disponible en: <https://www.java.com/es/>
- [9] Eclipse Foundation, ECLIPSE. 2020. [sitio web]. [consulta 23 mayo 2020]. Disponible en: <https://www.eclipse.org/>
- [10] Oracle, JAVAFX. 2020. [sitio web]. [consulta 23 mayo 2020]. Disponible en: <https://openjfx.io/>
- [11] Oracle JAVAFX SCENE BUILDER 2.0. 2020. [sitio web]. [consulta 23 mayo 2020]. Disponible en: <https://www.oracle.com/java/technologies/javafx-scene-builder-source-code.html>
- [12] W3SCHOOLS, CSS Tutorial, 2020. [sitio web]. [consulta 23 mayo 2020]. Disponible en: <https://www.w3schools.com/css/>
- [13] Visual Paradigm Online Diagrams. 2020. [sitio web]. [consulta 28 julio 2020]. Disponible en: <https://online.visual-paradigm.com/es/diagrams/solutions/free-use-case-diagram-tool/>
- [14] Wikipedia, Comparativa ganancia antena con polarización circular vs lineal 2020. [sitio web]. [consulta 16 julio 2020]. Disponible en: <https://es.wikipedia.org/wiki/Antena>

[15] Manuel Trigas Gallego, Metodología SCRUM. [sitio web]. [consulta 19 julio 2020].

Disponible en:

<http://openaccess.uoc.edu/webapps/o2/bitstream/10609/17885/1/mtrigasTFC0612memoria.pdf>

[16] Electrónica, Cálculo de atenuación en el espacio libre. [sitio web]. [consulta 19 julio

2020]. Disponible en: <https://www.telectronika.com/herramientas/perdida-en-el-espacio-libre/#:~:text=P%C3%A9rdida%20en%20el%20Espacio%20Libre&text=La%20atenuaci%C3%B3n%20en%20espacio%20libre,propagaci%C3%B3n%20de%20la%20onda%20electromagn%C3%A9tica>.