

**DISEÑO DE UN PROTOTIPO DE SISTEMA DE INVENTARIO DE EQUIPOS Y
REPUESTOS POR UN CONTROL RFID EN UNA BODEGA**

CAMILO GOMEZ GUEVARA
BRAYAN OSWALDO BERBEO RUIZ



UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS
FACULTAD TECNOLÓGICA
INGENIERÍA EN TELECOMUNICACIONES
BOGOTÁ 2018

**DISEÑO DE UN PROTOTIPO DE SISTEMA DE INVENTARIO DE EQUIPOS Y
REPUESTOS POR UN CONTROL RFID EN UNA BODEGA**

CAMILO GOMEZ GUEVARA
BRAYAN OSWALDO BERBEO RUIZ

TRABAJO DE GRADO PARA OPTAR AL TÍTULO DE:
INGENIERO EN TELECOMUNICACIONES

DIRECTOR:
ING. JOSE DAVID CELY CALLEJAS



UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS

FACULTAD TECNOLÓGICA

INGENIERÍA EN TELECOMUNICACIONES

BOGOTÁ 2018

Resumen

Esta monografía consiste en el desarrollo teórico-práctico de un prototipo de control de inventarios basado en tecnología RFID junto al desarrollo de aplicativos que interactúan con una base de datos brindando información veraz, rápida y práctica a los usuarios según su nivel de acceso al software y al manejo mismo del inventario.

En primer lugar se tiene el inventario en una bodega la cual cuenta con separaciones donde se alojan las distintas partes en “pockets”; a cada uno de estos, se le asigna un “tag” RFID que sirve como identificador, el cual se capta con un lector de etiquetas ARDUINO. Esto con el fin de brindar practicidad de reconocimiento e identificación de la o las partes en el software. Una vez identificada la parte en el software se procede a realizar operaciones de actualización de datos de inventario, consultas, validaciones y/o generación de informes de transacciones para su posterior análisis.

Lo anterior cumple con la función principal de la administración de inventario, además añadiendo la generación de información utilizada de forma adecuada, está guiando la empresa en la mejora de su administración interna con rotación de inventarios, elementos de alta, media o baja demanda, inventario estático, entre otros.

Se presenta así un desarrollo por capítulos que incluye cada una de las herramientas utilizadas para llegar al cumplimiento de los objetivos del proyecto planteados, los cuales se dan a conocer en el Capítulo 1 que presenta la introducción general del mismo y sus alcances. En el Capítulo 2 se da a conocer el marco teórico que implica cada recurso utilizado como referencia del desarrollo. El Capítulo 3 muestra el proceso metodológico de elaboración del proyecto. El Capítulo 4 muestra los resultados y análisis obtenidos; y finalmente el Capítulo 5 brinda las conclusiones definitivas encontradas.

Abstract

This monograph consists of the theoretical-practical development of a prototype of inventory control based on RFID technology, together with the development of applications that interact with a database providing truthful, fast and practical information to users according to their level of access to software and to the management of the inventory.

On first place, we have the inventory in a warehouse which has separations where the different parts are housed in "pockets"; Each of these is assigned an RFID tag that serves as an identifier, which is captured with an ARDUINO label reader. This in order to provide practical recognition and identification of the party or parties in the software. Once the part in the software is identified, operations are performed to update inventory data, queries, validations and / or generation of transaction reports for further analysis.

The above meets the main function of inventory management, but adding the generation of information that is used properly is guiding the company in improving its internal management with inventory rotation, elements of high, medium or low demand, inventory static, among others.

Then, is presented the development chapters including each of the tools used to reach the fulfillment of the objectives of the project raised, which are disclosed in Chapter 1, that presents the general introduction of the project and its scope. Chapter 2 is disclosed which involves the theoretical framework used as a reference each resource development. Chapter 3 shows the methodological process of developing the project. Chapter 4 shows the results obtained; and finally Chapter 5 provides the definitive conclusions found.

Tabla de contenido

LISTA DE IMÁGENES	8
LISTA DE TABLAS	10
1. INTRODUCCIÓN	11
1.1 Planteamiento del Problema y Justificación	11
1.2 Objetivos	12
1.2.1 Objetivo General	12
1.2.2 Objetivos Específicos	12
1.3 Estado del Arte	12
1.3.1 Análisis de los beneficios de la identificación por radiofrecuencia en un centro de distribución textil colombiano.	12
1.3.2 Desarrollo e implementación de un sistema para el control e inventario continuo, utilizando tecnología rfid, para la biblioteca de la ups sede Guayaquil.	13
1.3.3 Piloto en Grecia para recolectar basura.	14
1.3.4 Sistema de agentes para control de stock de almacén basado en identificación por radiofrecuencia.	14
1.3.5 Rfid: el código de barras inteligente para bibliotecas.	15
1.3.6 Sistema de registro y control de salida de elementos mediante dispositivos rfid.	15
2. MARCO TEÓRICO	16
2.1 Descripción de la Empresa	16
2.2 Base de Datos	16
2.2.1 Conceptos	16
2.2.2 Teoría sobre bases de datos	17
2.2.2.1 Tablas y campos.	18
2.2.2.1.1 Estructura con una tabla única.	18
2.2.2.1.2 Estructura multitablas.	18
2.2.2.2 Relación de Tablas.	19
2.2.2.3 Campos relacionados.	20
2.2.2.4 Llave primaria y llave foránea.	21
2.2.2.4.1 Llave primaria.	21
2.2.2.4.2 Llave foránea	21
2.2.3 Modelamiento racional sobre una base de datos	21
2.2.4 Modelamiento entidad relación de una base de datos.	24
2.2.5 Normalización de bases de datos en procedimientos.	24

2.2.5.1	Primera forma normal.	25
2.2.5.2	Segunda forma normal.	26
2.2.5.3	Tercera forma normal.	26
2.2.5.4	EXCEPCIÓN	26
2.3	Hardware	26
2.3.1	Importación de librerías Arduino	26
2.3.2	Lectura RFID	28
	El lector MIFARE MFRC522	29
2.3.3	Comunicación serial	36
2.3.3.1	Arduino y puerto serie	36
2.3.3.2	Estructura interna y configuración de un puerto serial	38
2.3.4	Arduino	38
2.4	Visual C#	39
2.4.1	Programación Orientada a Objetos (POO)	39
2.4.1.1	Clases y Objetos	39
2.4.1.2	Responsabilidades	39
2.4.1.3	Relaciones entre clases	39
2.4.1.4	Dependencia o uso	39
2.4.1.5	Asociación	40
2.4.1.6	Generalización	40
2.4.1.7	Clasificación atributos	40
2.4.1.8	Responsabilidades de objetos y Parametrización de métodos	40
2.4.1.9	Encapsulamiento	41
2.4.1.10	Mensajes y Métodos	41
2.4.2	Tipos de Datos	41
3.	METODOLOGÍA E IMPLEMENTACIÓN	43
3.1	Modelamiento y diseño de la arquitectura del presente software	43
3.1.1	Sistema de Lectura RFID	46
3.1.2	Sistema de GUI'S (Interfaces Gráficas de Usuario)	48
3.2	Requerimientos	49
3.2.1	Funcionales	49
3.2.2	Específicas	50
3.3	Documentación	50
4.	RESULTADOS, ANÁLISIS Y OBTENCIÓN DE DATOS	50
4.1	Manejo Interfaz Gráfica	50
4.2	Validaciones de campos	54
4.3	Análisis y Obtención de Datos	57

4.3.1	Rotación de partes	57
4.3.2	Rotación Equipos	59
4.3.3	Presupuesto	60
5.	CONCLUSIONES	61
6.	BIBLIOGRAFÍA	61

Lista de Imágenes

Imagen 1. Control de entrada del producto.

Imagen 2. Control de inventario.

Imagen 3. Control de salida del producto.

Imagen 4. Esquema de un sistema RFID.

Imagen 5. Esquema de clasificación de sistemas RFID.

Imagen 6. Esquema de sensores ubicados en una región de lectura junto a un lector de mayor alcance e implementando los pequeños en los extremos donde se generan más errores de lectura debido a factores de entorno y de potencia.

Imagen 7. La arquitectura de servicio.

Imagen 8. “Los componentes principales de un sistema RFID ”

Imagen 9. Teoría sobre Base de datos.

Imagen 10. Tabla única.

Imagen 11. Estructura multitable.

Imagen 12. Estructura de las bases de datos.

Imagen 13. Relación entre tablas de bases de datos.

Imagen 14. Definición de llave primaria.

Imagen 15. Base de datos de Northwind de Microsoft.

Imagen 16. Estructura interna y configuración de un puerto serial.

Imagen 17. Esquema general de clases.

Imagen 18. Representación gráfica dependencia.

Imagen 19. Representación asociación de clases.

Imagen 20. Representación gráfica de relación de clases a superclases.

Imagen 21. Atributos Actor Producto desde la GUI Productos.

Imagen 22. Atributos Actor Usuario desde la GUI usuario.

Imagen 23. Atributos Actor Flujo.

Imagen 24. Modelado BD, Diagrama entidad relación y Keys.

Imagen 25. Métodos Actor Producto desde la GUI Producto.

Imagen 26. Métodos de Actor Usuario.

Imagen 27. Conexión Arduino Mega y lector RFID.

Imagen 28. GUI Usuario.

Imagen 29. GUI Producto.

Imagen 30. Procedimientos Almacenados en Workbench.

Imagen 31. Diagrama Sistema de Inventarios.

Imagen 32. Campos de registro y eliminación de usuarios.

Imagen 33. Campos de validación de usuario y contraseña para el ingreso al aplicativo.

Imagen 34. Interfaz de administración del aplicativo.

Imagen 35. Botón Agregar, utilizado para añadir partes nuevas referenciadas en el inventario.

Imagen 36. El botón Leer Tarjeta.

Imagen 37. Campos TextBox para visualizar información de componentes almacenados en el aplicativo y actualización de datos.

Imagen 38. Botón Eliminar, para borrar un componente relacionado con un ID.

Imagen 39. Botón Buscar, para acceder a información relacionada con el ID una vez se ha leído la tarjeta RFID.

Imagen 40. Salida e Ingreso de unidades para actualización de cantidades.

Imagen 41. Botón Info, para la visualización de movimientos en la BD.

Imagen 42. Conexión puerto COM3, para inicio de sistema.

Imagen 43. Comprobación de comunicación con subsistema lectura RFID y GUI producto.

Imagen 44. Ejemplo ventana emergente, campos no diligenciados.

Imagen 45. Ejemplo ventana emergente datos no correspondientes.

Imagen 46. Ejemplo ventana emergente, duplicidad de llave primaria.

Imagen 47. Parte Visual de la GUI sobre la BD.

Imagen 48. Movimientos de partes por familia de equipos.

Imagen 49. Movimientos de partes por ubicación.

Imagen 50. Movimientos de partes por usuario.

Imagen 51. Suma de equipos en ventas.

Imagen 52. Totalización de transacciones en el aplicativo.

Lista de Tablas

Tabla 1. Parámetros SPI.

Tabla 2. Datos Básicos de C#

1. Introducción

El control de inventarios es un tema común en la mayoría de empresas al ser la base para poder brindar servicios a los posibles clientes y de esta forma ser competitivos; pero esto se ve retrasado debido a la complejidad del manejo de los mismos cuando son muy extensos en tamaño (bodegas) o cantidad de los mismos lo que obliga a utilizar herramientas para administrar como ERPs (Enterprise Resource Planning) que se complementan con herramientas MRP (Material Requirement Planning) y SAP (Sistemas, Aplicaciones y Productos para procesamiento de datos) los cuales permiten a la empresa tener un control más adaptado y óptimo en su interior tanto en inventario como en otras áreas.

Estos software empresariales representan una inversión elevada de capital por lo cual, en busca de un avance tecnológico se plantea un desarrollo específico al manejo y control de inventarios mediante dispositivos de reconocimiento por radiofrecuencias (RFID) en tiempo real y que al mismo tiempo registra los movimientos internos junto a datos como la hora, ubicación del ítem y tipo de movimiento con el fin de llevar un log en caso que sea necesario verificar estos datos. Lo anterior se propone a través de una interfaz gráfica amigable al usuario en donde se pueda ver los datos anteriormente mencionados en tiempo real con el fin de ser una herramienta estratégica para el crecimiento de una empresa.

El proyecto se desarrolla en varias secciones las cuales permiten conocer su estudio, desarrollo y aplicación; sección uno: se presenta el problema del cual se quiere dar solución, este abarca una justificación por la que es importante desarrollar proyectos innovadores y accesibles en este sector.

Sección dos: en este se presenta el estudio que complementa el desarrollo del proyecto, observando las tecnologías que se implementan y permiten dar un óptimo desarrollo del proyecto

Sección tres: Se lleva a cabo la metodología utilizada para su funcionamiento, identificando de cómo el proyecto da solución al problema inicial, posicionándose como una alternativa viable para ser utilizada y mostrando resultados efectivos al realizar su implementación.

1.1 Planteamiento del Problema y Justificación

Al momento de crear y operar una empresa, los inventarios (en la medida de cada tipo de empresa) varían en comportamiento y por lo general se hace uso de herramientas prácticas como tablas de cálculos o pequeñas bases de datos que son funcionales y eficientes en la medida de las posibilidades y capacidades del momento; pero que a medida del crecimiento tienden a encontrarse con obstáculos de funcionalidad lo que provoca que operaciones de consulta o modificación se vuelvan tediosas y complicadas, lo que provoca la incursión en

errores que significan un decrecimiento en la capacidad de respuesta y competitividad en el entorno comercial.

Esto se ve solucionado la mayoría de veces con la adquisición de un software ERP el cual permite la administración de los recursos en el interior de la empresa de forma más práctica lo que reemplaza en su totalidad al sistema inicial y brinda a la empresa la habilidad de competir en el mercado. Pero esto tiene un gran impedimento que hace que una gran parte del mercado de las empresas no opten por esta solución y es su muy elevado costo; no simplemente de adquisición sino costos de soporte y de implementación ya que es un proceso que tiene que adaptarse a cada empresa y esta consultoría también lleva un valor en la mayoría de casos que provoca ser una herramienta que algunas pocas pequeñas y medianas empresas pueden adquirir dejando una cantidad inmensa de empresas incapaces de seguir la velocidad del mercado.

1.2 Objetivos

1.2.1 Objetivo General

Diseñar un sistema de inventario de equipos y repuestos monitoreado en tiempo real para una bodega con el objeto de reducir tiempos y costos operativos.

1.2.2 Objetivos Específicos

- Diseñar un sistema de posicionamiento haciendo uso de la tecnología RFID en tiempo real para los equipos y partes de inventario en la bodega que interactúe con el sistema de gestión.
- Desarrollar un software de gestión que permita al usuario interactuar con el sistema actualizando y modificando los valores del inventario de forma práctica y eficaz haciendo uso del sistema RFID y la base de datos.
- Brindar un registro que evidencie el comportamiento del inventario para un análisis y control posterior generando conocimiento y bases para decisiones empresariales.

1.3 Estado del Arte

1.3.1 Análisis de los beneficios de la identificación por radiofrecuencia en un centro de distribución textil colombiano.

La identificación por radiofrecuencia compite con la identificación utilizando códigos de barras en la gestión de la cadena de suministro. Se analizará los beneficios de RFID frente a su inmediato competidor en la gestión logística; para lograr el objetivo, se planeó y ejecutó una prueba piloto de acuerdo a la logística actual de una empresa colombiana distribuidora de textiles. Durante la prueba piloto se realiza el análisis técnico de la implementación de la tecnología RFID y se efectúa un estudio comparativo de los beneficios entre código de barras y RFID. Finalmente, se demuestra que los beneficios económicos de RFID superan las expectativas de la empresa y se sustenta la viabilidad técnica de la implementación en dicho escenario.

Realizado el levantamiento de información se formaliza el diagrama de procesos y procedimientos a evaluar; se incluye el control de entrada y salida de productos, y el sistema

de inventarios piloto. Se hace el desglose de cada uno de los procesos y se establece el orden en que se debe desarrollar dentro de la experiencia piloto. [1] Diagramas de procesos de control [1]:



Imagen 1. Control de entrada del producto.



Imagen 2. Control de inventario.



Imagen 3. Control de salida del producto.

1.3.2 Desarrollo e implementación de un sistema para el control e inventario continuo, utilizando tecnología rfid, para la biblioteca de la ups sede Guayaquil.

En la Universidad Politécnica Salesiana sede Guayaquil se planteó una tesis con el fin de desarrollar e implementar un moderno sistema de control e inventario utilizando la comunicación inalámbrica (radiofrecuencia) por medio de la tecnología RFID para ser aplicado en las tesis de la biblioteca de la sede y así facilitar y mejorar la gestión bibliotecaria, además evitar las pérdidas de estos materiales tan importantes y fuentes de consulta para los alumnos.

Un nuevo y moderno sistema de gestión para préstamos, control de inventario, ingreso de nuevas tesis, control de salidas no autorizadas, todo realizado mediante comunicación inalámbrica por radiofrecuencia. La idea de este proyecto consiste en aplicar la tecnología RFID y hallar los mecanismos que permitan resolver necesidades actuales de manera rápida, precisa y cómoda.[2]

1.3.3 Piloto en Grecia para recolectar basura.

Con el fin de poder competir en el sector de la recolección de basuras, una empresa griega aplicó durante tres meses un piloto de control mediante RFID; debido a las restricciones de lectura de las etiquetas se diseñó un sistema que permite al momento de vaciar el contenedor en el camión recolector, leer en ese preciso momento la etiqueta ya que se tiene un rango de un metro de cobertura para este proceso de identificación garantizando que solo se lea cuando el contenedor está vaciándose y no simplemente cuando el camión pase cerca de este.

Este sistema funciona con un RFID ubicado en el contenedor de basura el cual se comunica con un teléfono móvil Qtek s200 PDA ubicada en el camión recolector a través del lector ubicado en el extremo trasero del camión; el cual mediante comunicación GPRS sube información al servidor registrando la hora exacta a la cual cada etiqueta fue leída y mediante un desarrollo de software en lenguaje .NET y Delphi con el fin de interactuar con la base de datos de estos contenedores y poder conocer el lugar y hora que fueron vaciadas.

Como principal característica se halla el poder tener un control sobre las rutas y los contenedores para obtener datos de eficiencia, rutas y sus respectivos tiempos; con el fin de obtener datos tabulables y que en su respectivo análisis pueden generar un conocimiento de mejora de rutas y control de tiempos para la productividad general. Este sistema no está restringido a la comunicación GPRS ya que existen pilotos del mismo sistema que solamente descarga la información de las rutas una vez llega a la base principal y se conecta a la red local de la compañía.[5]

1.3.4 Sistema de agentes para control de stock de almacén basado en identificación por radiofrecuencia.

En la Universidad de Castilla-La Mancha realizó un estudio sobre los sistemas de agentes para el control de stock de almacenes basado en identificación por radiofrecuencia (RFID).

Para saber qué sistema de agentes y control para un stock se debe tener información exacta y fiable del mismo como ubicación del producto, que tipos de productos hay en el stock, etc. La tecnología empleada actualmente en los stocks (Códigos de barras) no brindan los requisitos exigidos para una debida automatización y control, para ello se requiere prestaciones más elevadas, de información más exacta en tiempo real, de acuerdo a la

precisión exigida del sistema. Estas características nos las brinda la tecnología RFID y está siendo cada vez más solicitada ofreciendo la información satisfactoriamente.

Si la tecnología RFID se aporta con la tecnología de última generación como los sistemas de gestión de la información (Information Management System) se pueden identificar y gestionar la información exacta de un producto. [8]

1.3.5 Rfid: el código de barras inteligente para bibliotecas.

Aplicando esta tecnología de identificación en ambientes de almacenaje de documentos como lo son las bibliotecas se pueden llegar a obtener buenos resultados como se está implementando en Chile, donde algunas bibliotecas ya poseen una administración de documentos mediante radiofrecuencia.

Diferencian algunas etiquetas como las pasivas que no tienen fuente de alimentación propia; por lo cual cuando el lector se acerca a ellas, gracias al campo que se genera pueden alimentar el circuito interno y generar una respuesta al lector. La ventaja de estas etiquetas yace en que pueden ser de un tamaño muy reducido, incluso llegando a estar ubicadas bajo la piel humana en algunas pruebas.

Las semi-pasivas a pesar que comparten un funcionamiento muy similar las pasivas, si poseen una pequeña batería que alimenta de forma constante el circuito, eliminando la necesidad de utilizar una antena que reciba la señal del lector para convertirla en energía eléctrica y ampliando el rango de respuesta.

Las etiquetas activas, poseen una fuente incorporada lo que permite un rango de lectura mayor a las anteriores y memorias de información más grandes lo cual permite que el manejo de datos que pueden tener les dé una ventaja comparadas a las pasivas. Existen del tamaño de una moneda en adelante y la batería puede durar en el mejor de los casos hasta 2 años según fabricante.

Su aplicación en bibliotecas permite tener un rango de lectura amplio que se ve reflejado en la omisión del contacto con los objetos allí almacenados; no se tendrían que sacar ni manipular los libros para hacer la lectura de los mismos. Añadir que el código de barras permite identificar un producto, comparado a la etiqueta RFID que permite identificar un producto en particular o lotes del mismo, haciendo un análisis y un control más eficiente.

Lo anterior sumado a características como la seguridad de copias de las etiquetas, la reutilización de estas y la mayor información que puede almacenar cada etiqueta lo que hace que su aplicación y uso sea más eficiente en comparación de los códigos de barras convencionales.[9]

1.3.6 Sistema de registro y control de salida de elementos mediante dispositivos rfid.

La Pontificia Universidad Javeriana Facultad de ingeniería está implementando en sus laboratorios estrategias para brindar seguridad sobre los activos de la universidad, para esto el departamento de electrónica recurre al registro de entrada y salida de dichos elementos, antes de hacer la implementación de esta tecnología, los estudiantes de la institución

realizaron ciertos estudios demostrando la necesidad de implementación. Los estudios realizados fueron la entrada y salida de activos lo cual se convierte en un incremento continuo de las tareas para los laboratoristas, la falta de información en la marcación de los activos y el difícil acceso a los mismos por parte de los laboratoristas.

Atendiendo a estos contratiempos para el préstamo de los materiales de laboratorio desarrollaron e implementaron una solución que busca satisfacer la logística de elementos de la Universidad el cual tiene la opción de completar el control de acceso de personal al laboratorio y un software de registros, como resultado obtuvieron un producto que permite el control de los elementos y posiblemente implementado para el control de acceso a las personas de toda la institución. [10]

2. Marco Teórico

2.1 Descripción de la Empresa

La empresa para la cual se desarrolló el proyecto es un centro de venta y de servicio de equipos Kodak Alaris los cuales son utilizados en la digitalización de documentos. Se realiza tanto venta como alquiler de los equipos a empresas privadas y públicas, como a personas naturales o jurídicas.

El centro de servicio de la empresa se encarga de atender las llamadas de garantía de los equipos vendidos por los distintos canales del mayorista en Colombia, por lo que se necesita contar con un Stock de equipos que supla las ventas y partes para reparaciones correspondientes. Es aquí donde entra la importancia del control de las partes, ya que no son de fácil acceso en el mercado y solo existen 3 empresas permitidas para la importación y comercialización de las mismas a nivel nacional.

Es en la disponibilidad de las partes donde se centra un punto crítico para los clientes que utilizan los equipos para tareas fundamentales flujo de sus procesos o distintas áreas de su infraestructura. Además, estos repuestos pueden tardar de dos a tres semanas en el proceso de importación y sus precios son elevados.

2.2 Base de Datos

2.2.1 Conceptos

El objetivo principal de las bases de datos es el de unificar los datos que se manejan y los programas o aplicaciones que los manejan. Anteriormente los programas se codificaban junto con los datos, es decir, se diseñaban para la aplicación concreta que los iba a manejar, lo que desembocaba en una dependencia de los programas respecto a los datos, ya que la estructura de los ficheros va incluida dentro del programa, y cualquier cambio en la estructura del fichero provocaba modificar y compilar programas. Además, cada aplicación utiliza ficheros que pueden ser comunes a otras de la misma organización, por lo que se produce una REDUNDANCIA de la información, que provoca mayor ocupación de memoria, laboriosos programas de actualización (unificar datos recogidos por las aplicaciones de los diferentes departamentos), e inconsistencia de datos (no son correctos) si los datos no fueron bien actualizados en todos los programas. Con las bases de datos, se busca independizar los datos y las aplicaciones, es decir, mantenerlos en espacios diferentes. Los datos residen en memoria y los programas mediante un sistema gestor de bases de datos, manipulan la información.

2.2.2 Teoría sobre bases de datos

Una base de datos es un conjunto de información organizada de manera que pueda ser utilizada eficientemente. Un directorio telefónico, un diccionario, un calendario o un libro de recetas son ejemplos de bases de datos. La información en una base está organizada en forma de registros. Cada registro contiene toda la información sobre una persona o un elemento de la base. Por ejemplo, cada registro en el directorio telefónico contiene el nombre, dirección y número telefónico de una persona. Cada registro contiene campos. Un campo se utiliza para almacenar una información particular. Por ejemplo, en el directorio telefónico un campo almacena el nombre, otro campo almacena la dirección y otro campo almacena el número telefónico de la persona. Cada registro contiene cada uno de estos campos y cada registro puede tener información en esos campos.

El nombre de un campo generalmente identifica la información almacenada en el campo. Por ejemplo, los campos pueden llamarse Nombre, Dirección o Número telefónico. Cada campo tiene un tipo que identifica la clase de información que puede almacenar: números, fechas, caracteres alfanuméricos y otros. Como cada campo contiene un tipo específico de datos, usted puede realizar cálculos y otras operaciones con la información guardada en ellos. Por ejemplo, puede sumar los números de dos campos. Puede comparar la fecha de un campo con la de otro. Puede mostrar el nombre de una persona (almacenado en un campo) después de su apellido (almacenado en otro campo) para construir la primera línea de una etiqueta de correo. El conjunto de registros que utilizan los mismos campos conforma una tabla. Una base de datos puede contener muchas tablas. La siguiente imagen muestra cómo se relacionan estos conceptos.

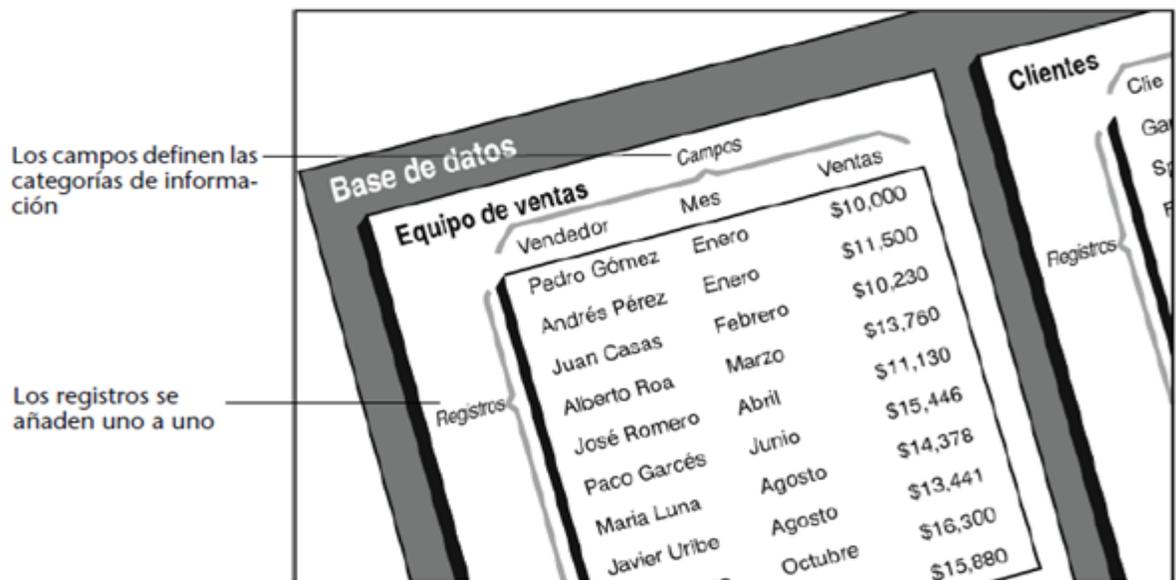


Imagen 9. Teoría sobre Base de datos.

2.2.2.1 Tablas y campos.

2.2.2.1.1 Estructura con una tabla única.

Algunas bases de datos tienen una tabla única. Puede utilizar una tabla única para categorías como empleados, empresas, inventario, etc. Puede definir hasta 32 767 campos por tabla, por ejemplo la imagen 10.



MUSICOS	
nombre	A
año_Nacimiento	2 ¹⁶
pais_Nacimiento	A
año_Defuncion	2 ¹⁶
notas	T

Imagen 10. Tabla única.

2.2.2.1.2 Estructura multitaslas.

Una base de datos que utiliza más de una tabla puede almacenar muchos más datos y acceder a la información de forma más eficiente. Una buena regla para tener en cuenta es que los diferentes tipos de información deben almacenarse en diferentes tablas.

Una base de datos que guarda información sobre empleados y empresas es un buen ejemplo. Los registros de los empleados y las empresas se guardan en tablas diferentes. Si la dirección de una empresa cambia, sólo debe modificar el registro de esta empresa. No tiene que escribir la dirección nueva para todas las personas que trabajen en esa empresa. La siguiente imagen representa la estructura de una base de datos con dos tablas relacionadas. La flecha entre el campo [EMPLEADOS] Empresa y el campo [EMPRESAS] Nombre representa esta relación:



Imagen 11. Estructura multitasla.

Los datos de cada empleado se almacenan en la tabla [EMPLEADOS]. Los datos sobre las empresas se almacenan en la tabla [EMPRESAS].

2.2.2.2 Relación de Tablas.

Generalmente las tablas tienen en común algunos datos. Por ejemplo, imagine que crea una base para almacenar información sobre los empleados y las empresas que los emplean. La estructura de la base, representada a continuación, tiene una tabla [EMPLEADOS] para almacenar la información relativa a los empleados y una tabla [EMPRESAS] para almacenar la información relativa a las empresas.

EMPLEADOS	
Nombre	🔍
Apellido	🔍
Direccion	🔍
Codigo_postal	🔍
Telefono	🔍
Empresa	🔍
Fecha_Contratacion	📅
Foto	🖼️
Ciudad	🔍
Cargo	🔍
Salario	0.5
Departamento	🔍

EMPRESAS	
Nombre	🔍
Direccion	📍
Ciudad	🔍
Codigo_postal	🔍
Telefono	🔍
Estado	🔍
Subsidiario	🏢

Imagen 12. Estructura de las bases de datos.

Aunque útil, la información almacenada en cada tabla no corresponde completamente a sus necesidades. Cuando visualiza un registro de la tabla [EMPLEADOS], debe poder acceder a la información sobre la empresa para la cual el empleado trabaja y cuando mira un registro de la tabla [EMPRESAS], necesita poder ver la información sobre los empleados que trabajan para esa empresa.

Para permitir a dos tablas compartir información de esta manera, las tablas pueden estar relacionadas entre ellas, en otras palabras, una relación puede establecerse entre los datos de cada tabla.

Crear las relaciones entre las tablas permite:

- Almacenar datos eficientemente.
- Actualizar los datos en un lugar y ver el cambio reflejado en todas las partes donde se utilizan los datos.
- Ver información relacionada.
- Realizar consultas y ordenaciones en una tabla, con base en los datos almacenados en otra tabla.
- Crear, modificar o eliminar los registros de tablas relacionadas.

La siguiente imagen muestra una relación creada entre la tabla [EMPLEADOS] y la tabla [EMPRESAS] en el editor de estructura:



Imagen 13. Relación entre tablas de bases de datos.

La tabla [EMPLEADOS] contiene un registro por persona. La tabla [EMPRESAS] contiene un registro por empresa. La relación entre estas dos tablas le permite visualizar, introducir, modificar o eliminar información de las dos tablas. Por ejemplo:

Cuando el registro de un empleado está en pantalla, puede mostrar la información de la empresa correspondiente, la dirección, ciudad, estado, código postal y el número telefónico de la empresa.

Cuando añade un nuevo empleado, puede asociar el registro de la persona con el registro de la empresa correspondiente (si la empresa ya existe en la base) o crear el registro de una nueva empresa mientras crea el registro del empleado. Para cada empresa, usted puede ver o modificar la información de cada una de las personas que trabajan en ella, nombre, cargo, número telefónico y otros datos. También puede añadir un nuevo registro de persona desde el registro de la empresa.

2.2.2.3 Campos relacionados.

Gracias a los campos relacionados, los campos que conectan dos tablas en una relación, usted puede mostrar la información de las tablas relacionadas.

En el ejemplo anterior, el campo Empresa de la tabla [EMPLEADOS] y el campo Nombre de la tabla [EMPRESAS] relacionan las dos tablas. El campo Nombre en la tabla [EMPRESAS] es el campo llave primaria de la tabla [EMPRESAS]. Este campo identifica de manera única cada registro de empresa. Un campo llave primaria debe tener los atributos Indexado y Único. Si el campo llave primaria no tiene el atributo Indexado. El campo Empresa de la tabla [EMPLEADOS] es un campo llave foránea. Cada valor del campo llave foránea corresponde a un sólo valor del campo llave primaria en la tabla relacionada.

2.2.2.4 Llave primaria y llave foránea.

2.2.2.4.1 Llave primaria.

La clave o llave primaria es un campo, o grupo de campos que identifica en forma única un registro. Ningún otro registro puede tener la misma llave primaria. La llave primaria se utiliza para distinguir un registro con el fin de que se pueda tener acceso a ellos, organizarlos y manipularlos. En el caso de un registro de un empleado, el número de este representa un ejemplo de una llave primaria.

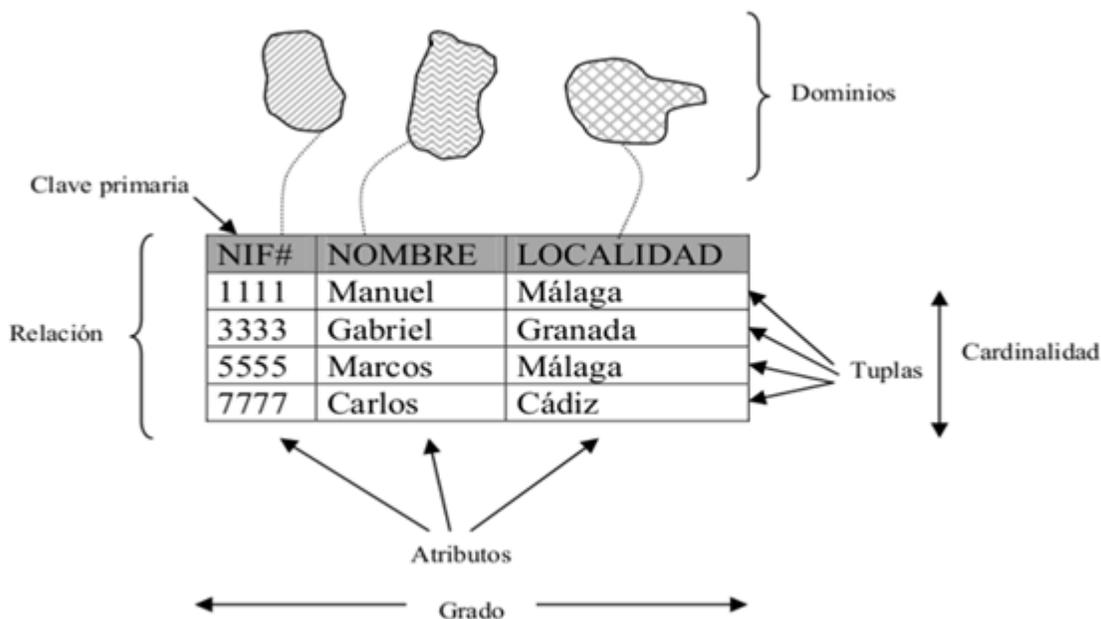


Imagen 14. Definición de llave primaria.

2.2.2.4.2 Llave foránea

Una llave foránea en una base de datos relacional es una llave que se usa en una tabla secundaria y que coincide con la llave primaria en una tabla primaria relacionada. Las llaves foráneas pueden tener valores duplicados (multiplicidad) en la tabla secundaria, mientras que para las llaves primarias eso no es posible. El uso apropiado de llaves foráneas permite exigir la integridad referencial.

2.2.3 Modelamiento racional sobre una base de datos

El diseño de una base de datos consiste en definir la estructura de los datos que debe tener un sistema de información determinado. Para ello se suelen seguir por regla general unas fases en el proceso de diseño, definiendo para ello el modelo conceptual, el lógico y el físico. En el diseño conceptual se hace una descripción de alto nivel de la estructura de la base de datos, independientemente del SGBD (Sistema Gestor de Bases de Datos) que se vaya a utilizar para manipularla. Su objetivo es describir el contenido de información de la base de datos y no las estructuras de almacenamiento que se necesitarán para manejar dicha información.

El diseño lógico parte del resultado del diseño conceptual y da como resultado una descripción de la estructura de la base de datos en términos de las estructuras de datos que puede procesar un tipo de SGBD. El diseño lógico depende del tipo de SGBD que se vaya a utilizar, se adapta a la tecnología que se debe emplear, pero no depende del producto concreto. En el caso de bases de datos convencionales relacionales (basadas en SQL para entendernos), el diseño lógico consiste en definir las tablas que existirán, las relaciones entre ellas, normalizarlas, etc...

El diseño físico parte del lógico y da como resultado una descripción de la implementación de una base de datos en memoria secundaria: las estructuras de almacenamiento y los métodos utilizados para tener un acceso eficiente a los datos. Aquí el objetivo es conseguir una mayor eficiencia, y se tienen en cuenta aspectos concretos del SGBD sobre el que se vaya a implementar. Por regla general esto es transparente para el usuario, aunque conocer cómo se implementa ayuda a optimizar el rendimiento y la escalabilidad del sistema.

En el modelo relacional las dos capas de diseño conceptual y lógico, se parecen mucho. Generalmente se implementan mediante diagramas de Entidad/Relación (modelo conceptual) y tablas y relaciones entre éstas (modelo lógico). Este es el modelo utilizado por los sistemas gestores de datos más habituales (SQL Server, Oracle, MySQL...).

El modelo relacional de bases de datos se rige por algunas normas sencillas:

- Todos los datos se representan en forma de tablas (también llamadas “relaciones”, ver nota anterior). Incluso los resultados de consultar otras tablas. La tabla es además la unidad de almacenamiento principal.
- Las tablas están compuestas por filas (o registros) y columnas (o campos) que almacenan cada uno de los registros (la información sobre una entidad concreta, considerados una unidad).
- Las filas y las columnas, en principio, carecen de orden a la hora de ser almacenadas. Aunque en la implementación del diseño físico de cada SGBD esto no suele ser así. Por ejemplo, en SQL Server si añadimos una clave de tipo "Clustered" a una tabla haremos que los datos se ordenen físicamente por el campo correspondiente.
- El orden de las columnas lo determina cada consulta (que se realizan usando SQL).
- Cada tabla debe poseer una clave primaria, esto es, un identificador único de cada registro compuesto por una o más columnas.
- Para establecer una relación entre dos tablas es necesario incluir, en forma de columna, en una de ellas la clave primaria de la otra. A esta columna se le llama clave externa. Ambos conceptos de clave son extremadamente importantes en el diseño de bases de datos.

Basándose en estos principios se diseñan las diferentes bases de datos relacionales, definiendo un diseño conceptual y un diseño lógico, que luego se implementa en el diseño físico usando para ello el gestor de bases de datos de nuestra elección (por ejemplo SQL Server).

Por ejemplo, consideremos la conocida base de datos de Northwind de Microsoft. Esta base de datos representa un sistema sencillo de gestión de pedidos para una empresa ficticia. Existen conceptos que hay que manejar como: proveedores, empleados, clientes, empresas de transporte, regiones geográficas, y por supuesto pedidos y productos. El diseño conceptual de la base de datos para manejar toda esta información se puede ver en la siguiente figura, denominada diagrama Entidad/Relación o simplemente diagrama E-R:

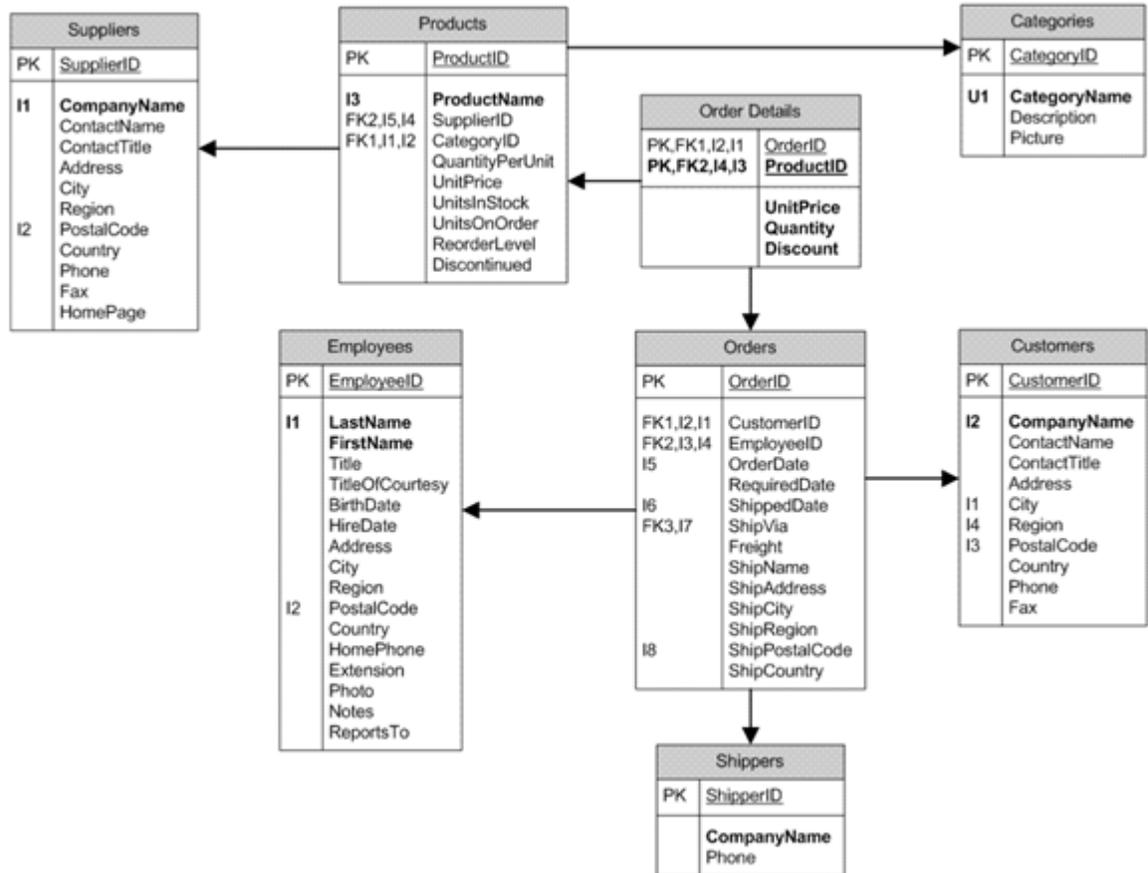


Imagen 15. Base de datos de Northwind de Microsoft.

Como vemos existen tablas para representar cada una de estas entidades del mundo real: Proveedores (Suppliers), Productos, Categorías de productos, Empleados, Clientes, Transportistas (Shippers), y Pedidos (Orders) con sus correspondientes líneas de detalle (Order Details). Además están relacionadas entre ellas de modo que, por ejemplo, un producto pertenece a una determinada categoría (se relacionan por el campo CategoryID) y un proveedor (SupplierID), y lo mismo con las demás tablas.

Cada tabla posee una serie de campos que representan valores que queremos almacenar para cada entidad. Por ejemplo, un producto posee los siguientes atributos que se traducen en los campos correspondientes para almacenar su información: Nombre (ProductName), Proveedor (SupplierID, que identifica al proveedor), Categoría a la que pertenece (CategoryID), Cantidad de producto por cada unidad a la venta (QuantityPerUnit), Precio unitario (UnitPrice), Unidades que quedan en stock (UnitsInStock), Unidades de ese producto que están actualmente en pedidos (UnitsOnOrder), qué cantidad debe haber para

que se vuelva a solicitar más producto al proveedor (ReorderLevel) y si está descatalogado o no (Discontinued).

Los campos marcados con "PK" indican aquellos que son claves primarias, es decir, que identifican de manera única a cada entidad. Por ejemplo, ProductID es el identificador único del producto, que será por regla general un número entero que se va incrementando cada vez que introducimos un nuevo producto (1, 2, 3, etc...).

Los campos marcados como "FK" son claves foráneas o claves externas. Indican campos que van a almacenar claves primarias de otras tablas de modo que se puedan relacionar con la tabla actual. Por ejemplo, en la tabla de productos el campo CategoryID está marcado como "FK" porque en él se guardará el identificador único de la categoría asociada al producto actual. En otras palabras: ese campo almacenará el valor de la clave primaria (PK) de la tabla de categorías que identifica a la categoría en la que está ese producto.

Los campos marcados con indicadores que empiezan por "I" (ej.: "I1") se refieren a índices. Los índices generan información adicional para facilitar la localización más rápida de registros basándose en esos campos. Por ejemplo, en la tabla de empleados (Employees) existe un índice "I1" del que forman parte los campos Nombre y Apellidos (en negrita además porque serán también valores únicos) y que indica que se va a facilitar la localización de los clientes mediante esos datos. También tiene otro índice "I2" en el campo del código postal para localizar más rápidamente a todos los clientes de una determinada zona.

Los campos marcados con indicadores que empiezan con "U" (por ejemplo U1) se refieren a campo que deben ser únicos. Por ejemplo, en la tabla de categorías el nombre de ésta (CategoryName) debe ser único, es decir, no puede haber -lógicamente- dos categorías con el mismo nombre.

2.2.4 Modelamiento entidad relación de una base de datos.

La entidad es cualquier clase de objeto o conjunto de elementos presentes o no, en un contexto determinado dado por el sistema de información o las funciones y procesos que se definen en un plan de automatización. Dicho de otra forma, las entidades las constituyen las tablas de la base de datos que permiten el almacenamiento de los ejemplares o registros del sistema, quedando recogidos bajo la denominación o título de la tabla o entidad. Por ejemplo, la entidad usuarios guarda los datos personales de los usuarios de la biblioteca, la entidad catálogo registra todos los libros catalogados, la entidad circulación todos los libros prestados y devueltos y así sucesivamente con todos los casos.

2.2.5 Normalización de bases de datos en procedimientos.

La normalización es el proceso de organizar los datos de una base de datos. Se incluye la creación de tablas y el establecimiento de relaciones entre ellas según reglas diseñadas tanto para proteger los datos como para hacer que la base de datos sea más flexible al eliminar la redundancia y las dependencias incoherentes.

Los datos redundantes desperdician el espacio de disco y crean problemas de mantenimiento. Si hay que cambiar datos que existen en más de un lugar, se deben cambiar de la misma forma exactamente en todas sus ubicaciones. Un cambio en la dirección de un cliente es mucho más fácil de implementar si los datos sólo se almacenan en la tabla Clientes y no en algún otro lugar de la base de datos.

¿Qué es una "dependencia incoherente"? Aunque es intuitivo para un usuario mirar en la tabla Clientes para buscar la dirección de un cliente en particular, puede no tener sentido mirar allí el salario del empleado que llama a ese cliente. El salario del empleado está relacionado con el empleado, o depende de él, y por lo tanto se debería pasar a la tabla Empleados. Las dependencias incoherentes pueden dificultar el acceso porque la ruta para encontrar los datos puede no estar o estar interrumpida.

Hay algunas reglas en la normalización de una base de datos. Cada regla se denomina una "forma normal". Si se cumple la primera regla, se dice que la base de datos está en la "primera forma normal". Si se cumplen las tres primeras reglas, la base de datos se considera que está en la "tercera forma normal". Aunque son posibles otros niveles de normalización, la tercera forma normal se considera el máximo nivel necesario para la mayor parte de las aplicaciones.

Al igual que con otras muchas reglas y especificaciones formales, en los escenarios reales no siempre se cumplen los estándares de forma perfecta. En general, la normalización requiere tablas adicionales y algunos clientes consideran éste un trabajo considerable. Si decide infringir una de las tres primeras reglas de la normalización, asegúrese de que su aplicación se anticipa a los problemas que puedan aparecer, como la existencia de datos redundantes y de dependencias incoherentes.

2.2.5.1 Primera forma normal.

- Elimine los grupos repetidos de las tablas individuales.
- Cree una tabla independiente para cada conjunto de datos relacionados.
- Identifique cada conjunto de datos relacionados con una clave principal.

No use varios campos en una sola tabla para almacenar datos similares. Por ejemplo, para realizar el seguimiento de un elemento del inventario que proviene de dos orígenes posibles, un registro del inventario puede contener campos para el Código de proveedor 1 y para el Código de proveedor 2.

¿Qué ocurre cuando se agrega un tercer proveedor? Agregar un campo no es la respuesta, requiere modificaciones en las tablas y el programa, y no admite fácilmente un número variable de proveedores. En su lugar, coloque toda la información de los proveedores en una tabla independiente denominada Proveedores y después vincule el inventario a los proveedores con el número de elemento como clave, o los proveedores al inventario con el código de proveedor como clave.

2.2.5.2 Segunda forma normal.

- Cree tablas independientes para conjuntos de valores que se apliquen a varios registros.
- Relacione estas tablas con una clave externa.

Los registros no deben depender de nada que no sea una clave principal de una tabla, una clave compuesta si es necesario. Por ejemplo, considere la dirección de un cliente en un sistema de contabilidad. La dirección se necesita en la tabla Clientes, pero también en las tablas Pedidos, Envíos, Facturas, Cuentas por cobrar y Colecciones. En lugar de almacenar la dirección de un cliente como una entrada independiente en cada una de estas tablas, almacénese en un lugar, ya sea en la tabla Clientes o en una tabla Direcciones independiente.

2.2.5.3 Tercera forma normal.

- Elimine los campos que no dependen de la clave.

Los valores de un registro que no sean parte de la clave de ese registro no pertenecen a la tabla. En general, siempre que el contenido de un grupo de campos pueda aplicarse a más de un único registro de la tabla, considere colocar estos campos en una tabla independiente. Por ejemplo, en una tabla Contratación de empleados, puede incluirse el nombre de la universidad y la dirección de un candidato. Pero necesita una lista completa de universidades para enviar mensajes de correo electrónico en grupo. Si la información de las universidades se almacena en la tabla Candidatos, no hay forma de enumerar las universidades que no tengan candidatos en ese momento. Cree una tabla Universidades independiente y vinculada a la tabla Candidatos con el código de universidad como clave.

2.2.5.4 EXCEPCIÓN

Cumplir la tercera forma normal, aunque en teoría es deseable, no siempre es práctico. Si tiene una tabla Clientes y desea eliminar todas las dependencias posibles entre los campos, debe crear tablas independientes para las ciudades, códigos postales, representantes de venta, clases de clientes y cualquier otro factor que pueda estar duplicado en varios registros. En teoría, la normalización merece el trabajo que supone. Sin embargo, muchas tablas pequeñas pueden degradar el rendimiento o superar la capacidad de memoria o de archivos abiertos.

2.3 Hardware

2.3.1 Importación de librerías Arduino

LIBRERÍA <SPI.h>

Esta librería le permite comunicarse con dispositivos SPI, con el Arduino como dispositivo maestro.

Serial Peripheral Interface (SPI) es un protocolo de datos en serie síncrono utilizado por los microcontroladores para comunicarse rápidamente con uno o más dispositivos periféricos en distancias cortas. También se puede usar para la comunicación entre dos microcontroladores.

Con una conexión SPI, siempre hay un dispositivo maestro (generalmente un microcontrolador) que controla los dispositivos periféricos. Normalmente, hay tres líneas comunes a todos los dispositivos:

- MISO (Master In Slave Out) - La línea esclava para enviar datos al maestro,
- MOSI (Master Out Slave In) - La línea Master para enviar datos a los periféricos,
- SCK (Serial Clock) - Los pulsos de reloj que sincronizan la transmisión de datos generada por el maestro y una línea específica para cada dispositivo:
- SS (Slave Select): el pin en cada dispositivo que el maestro puede usar para habilitar y deshabilitar dispositivos específicos.

Cuando el pin de Selección de esclavo de un dispositivo está bajo, se comunica con el maestro. Cuando es alto, ignora al maestro. Esto le permite tener múltiples dispositivos SPI compartiendo las mismas líneas MISO, MOSI y CLK.

Para escribir el código de un nuevo dispositivo SPI, debe tener en cuenta algunas cosas:

- ¿Cuál es la velocidad máxima de SPI que su dispositivo puede usar? Esto es controlado por el primer parámetro en SPISettings. Si está utilizando un chip con una clasificación de 15 MHz. Arduino usará automáticamente la mejor velocidad que sea igual o menor que el número que usa con SPISettings.
- ¿Los datos se desplazan primero en el bit más significativo (MSB) o en el bit menos significativo (LSB)? Esto se controla mediante el segundo parámetro SPISettings, ya sea MSBFIRST o LSBFIRST. La mayoría de los chips SPI usan el primer orden de datos de MSB.
- ¿El reloj de datos está inactivo cuando está alto o bajo? ¿Hay muestras en el borde ascendente o descendente de los pulsos de reloj? Estos modos están controlados por el tercer parámetro en SPISettings.

El estándar SPI está suelto y cada dispositivo lo implementa de forma un poco diferente. Esto significa que debe prestar especial atención a la hoja de datos del dispositivo al escribir su código.

En términos generales, hay cuatro modos de transmisión. Estos modos controlan si los datos se desplazan hacia adentro y hacia afuera en el borde ascendente o descendente de la señal del reloj de datos (llamada fase del reloj), y si el reloj está inactivo cuando está alto o bajo (llamado polaridad del reloj). Los cuatro modos combinan polaridad y fase según esta tabla:

Modo	Polaridad del reloj (CPOL)	Fase de reloj (CPHA)	Salida Edge	Captura de datos
SPI_MODE0	0	0	Q ue cae	Creciente

SPI_MODE1	0	1	C reciente	Que cae
SPI_MODE2	1	0	C reciente	Que cae
SPI_MODE3	1	1	Q ue cae	Creciente

Tabla 1. Parámetros SPI.

Una vez que tenga sus parámetros SPI, use `SPI.beginTransaction ()` para comenzar a usar el puerto SPI. El puerto SPI se configurará con todas sus configuraciones. La forma más simple y eficiente de usar `SPISettings` está directamente dentro de `SPI.beginTransaction ()`. Por ejemplo:

`SPI.beginTransaction(SPISettings(14000000, MSBFIRST, SPI_MODE0));` Si otras bibliotecas usan SPI de las interrupciones, se les impedirá acceder a SPI hasta que llame `SPI.endTransaction()`. La configuración de SPI se aplica al comienzo de la transacción y `SPI.endTransaction()` no cambia la configuración de SPI. A menos que usted, o alguna biblioteca, llame a `startTransaction` por segunda vez, la configuración se mantiene. Debe intentar minimizar el tiempo entre antes de llamar `SPI.endTransaction()`, para obtener la mejor compatibilidad si su programa se usa junto con otras bibliotecas que usan SPI.

Con la mayoría de los dispositivos SPI, después `SPI.beginTransaction()`, escribirá el PIN de selección esclavo BAJO, llamará `SPI.transfer()` cualquier número de veces para transferir datos, luego escribirá el pin SS ALTO, y finalmente llamará `SPI.endTransaction()`.

2.3.2 Lectura RFID

Los lectores de RFID o “interrogadores” utilizan ondas de radio para leer la información almacenada en la etiqueta. El lector puede enviar a la etiqueta la orden de transmitir la información que tiene almacenada, o bien la etiqueta puede transmitir la información que contiene periódicamente, en espera de que algún lector la detecte. En definitiva, el lector es el dispositivo que interactúa con el tag o etiqueta. Los lectores de radiofrecuencia constan de:

- Una antena.
- Un módulo de radiofrecuencia o controlador.
- Una unidad de control.

Las antenas del lector son el componente más sensible de un sistema RFID y pueden variar en función de la aplicación concreta. Así, las antenas pueden ser:

- De sobremesa: se incorporan a los puntos de identificación y control o terminales punto de servicio, para la lectura y escritura de las etiquetas.
- De puerta: se disponen en pedestales en el suelo e incluyen elementos de alarma con fines antirrobo y para control de acceso.
- De diseño a medida (góndolas, expositores...).

Variar la ubicación de la antena del lector es una de las formas más fáciles de ajuste cuando se localizan problemas de un sistema, y al mismo tiempo resulta una de las tareas más difíciles de llevar a cabo de forma correcta.

La antena del lector debe ser colocada en una posición que permita optimizar tanto la transmisión de energía hacia la etiqueta como la recepción de los datos emitidos. Las limitaciones legislativas del nivel de potencia de los lectores hacen que la ubicación de las antenas sea fundamental para alcanzar un alto grado de lectura. El funcionamiento básico del lector suele ser muy sencillo. La antena del lector crea un campo magnético cuyo radio de acción varía dependiendo de la forma de la antena, de la potencia del lector y de la frecuencia en la que emita. En el caso de etiquetas pasivas, el lector genera un campo de radiofrecuencia que activa la etiqueta alimentando el chip para que pueda transmitir los datos almacenados. En el caso de etiquetas activas la propia etiqueta puede iniciar la comunicación, pues no necesita la energía del lector. Cuando la etiqueta entra en contacto con el campo magnético creado por el lector, reacciona automáticamente y envía la información a éste. El lector la decodifica y la pone en contacto con el sistema de control, a través de una aplicación que se ejecuta en su unidad de control y que se conoce como ONS (Object Name Service). Algunos sistemas utilizan encriptación de clave pública para codificar la información que fluye entre la etiqueta y el lector con objeto de conseguir mayor seguridad ante posibles escuchas maliciosas y obtener cierta privacidad. Un lector puede utilizarse también para reescribir sobre una etiqueta, siempre y cuando el lector esté habilitado para ello y la etiqueta sea “reescribible”. Naturalmente, estos lectores con doble función tienen un precio superior al de aquellos que sólo leen. Los lectores pueden ser portátiles o fijos, y también pueden estar integrados en otros equipos como terminales handheld (PDAs, Lectores de mano, etc.), grúas horquillas, o cualquier equipo que se considere estratégico.[28]

El lector MIFARE MFRC522

MIFARE es una tecnología de tarjetas inalámbricas propiedad de NXP Semiconductors. Es uno de los estándares más implantados como tarjetas inteligentes sin contacto (TSIC).

El Mifare MFRC522 es un lector de tarjetas RFID que incorpora comunicación por bus SPI, bus I2C y UART, por lo que es sencillo de conectar con Arduino. El MFRC522 soporta las tarjetas Mifate S50, Mifare S70, Mifare UltraLight, Mifare Pro y Mifare Desfire.

El lector MFRC522 opera en la frecuencia de 13.56Mhz y tiene una distancia de lectura de 0 a 60. El MFRC522 tiene un consumo de 13-26 mA durante la escritura, 10-13mA en stanby e inferior a 80uA en modo sleep. La tensión de alimentación es de 3.3V.

El MFRC522 suele suministrarse con tarjetas o llaveros MIFARE Classic 1K. Este tipo de tarjetas son, esencialmente, un sistema de almacenamiento donde la memoria está dividida en bloques, con mecanismos simple para el acceso a la información. El MIFARE Classic 1K dispone de 1024 bytes de memoria divididos en 16 sectores de 64 bytes, cada uno protegido por dos claves llamadas A y B. Cada una puede ser programada individualmente para permitir o bloquear operaciones lectura o escritura.

Cada sector reserva una cierta memoria para las claves A y B, por lo que este espacio normalmente no puede ser empleado para guardar datos, lo que reduce la cantidad de memoria disponible en una MIFARE Classic 1K a 752 bytes. La memoria EEPROM de las tarjetas MIFARE Classic puede soportar más de 100.000 ciclos de escritura, y pueden mantener la memoria durante más de 10 años sin recibir alimentación.

2.3.2.1 RFID.

La tecnología RFID (Radio Frequency Identification) es un sistema de almacenamiento de datos remotos que usa dispositivos conocidos mayormente como etiquetas o tags RFID. El propósito fundamental de esta tecnología es ser capaz de obtener el identificador de un elemento mediante ondas de radio. Las etiquetas RFID pueden ser unas etiquetas normales (normalmente etiquetas autoadhesivas) a las que se les introduce un inlay RFID (Fig.1). Estos inlays contienen antenas para permitirles recibir y responder a peticiones por radiofrecuencia desde un emisor-receptor RFID. Las etiquetas suelen ser pasivas, que significa que no necesitan alimentación eléctrica interna. [26]



Imagen 8 (Inlays RFID).

La tecnología de identificación por Radiofrecuencia RFID es una de las tecnologías de comunicación que ha experimentado un crecimiento más acelerado y constante en los últimos tiempos.

Las posibilidades que RFID ofrece son: la lectura a distancia de la información contenida en una etiqueta sin necesidad de contacto físico, y la capacidad para realizar múltiples lecturas

simultáneamente (y en algunos casos escrituras). Esto abre la puerta a un amplio conjunto de aplicaciones en una gran variedad de ámbitos. Desde soluciones de trazabilidad y control de inventarios en un almacén o cadena de montaje, hasta la localización, seguimiento e identificación de objetos e incluso personas, o la seguridad en controles de acceso.

Ya son muchas las empresas que apoyan la implantación de RFID. Al tratarse de una tecnología que puede aportar importantes ventajas en muchos ámbitos de aplicación, se puede esperar que en un futuro no muy lejano, llegue a convertirse en una de las principales. [27]

2.3.2.2 Componentes de un sistema rfid

Un sistema RFDI básico consta de dos elementos fundamentales:

- Una etiqueta, tag o transpondedor RFID. Es el elemento en el que se incorpora el chip electrónico que contiene la información necesaria para identificar al producto. Contiene también una microantena que le permite recibir y transmitir la información a través de radiofrecuencia.
- Un lector RFID. Se trata de un equipo capaz de recibir y procesar la información procedente de los tags RFID. Este equipo cuenta con una antena que le permite recibir y transmitir peticiones de información a través de radiofrecuencia. Además de estos dos elementos imprescindibles, un sistema RFID puede estar formado por varios componentes más, como estaciones de programación de etiquetas (que podría ser la misma que la de lectura, ya que es habitual encontrar en el mercado estaciones de lectura/escritura), lectores de circulación, bases de datos, etc.

2.3.2.3 Etiquetas rfid

Básicamente, una etiqueta RFID está compuesta por un chip unido a una antena. Dependiendo de las necesidades de la aplicación puede presentar diferentes formatos (rígido, flexible, adhesivo, etc.), diferentes tamaños, funcionalidades, estándares y, por lo tanto, diferentes precios. En cuanto a las dimensiones de las etiquetas, existe una amplia gama de ofertas y posibilidades en el mercado que van desde unos pocos milímetros hasta decenas de centímetros, dependiendo de la aplicación a la que se destinen.

Las etiquetas RFID pueden ser de tres tipos distintos dependiendo del lugar del que provenga la energía que utilizan para tramitar la respuesta. Así, las etiquetas RFID pueden ser pasivas, si no tienen fuente de alimentación propia, semipasivas, si utilizan una pequeña batería asociada, y activas, si tienen su propia fuente de alimentación. Las etiquetas RFID pasivas no tienen fuente de alimentación propia. La corriente eléctrica necesaria para su funcionamiento se obtiene por inducción en su antena de la señal de radiofrecuencia procedente de la petición de lectura de la estación lectora. De esta manera, cuando el lector interroga a la etiqueta, genera un campo magnético que produce en la microantena del tag un campo eléctrico suficiente que permite generar la energía necesaria para que el circuito CMOS7, integrado en la etiqueta, pueda transmitir una respuesta. Debido a esto, la señal de respuesta tiene un tiempo de vida bastante corto; además, en la práctica, las distancias de lectura son relativamente reducidas (varían entre unos pocos milímetros hasta distancias cercanas a 1 metro). Las etiquetas RFID semipasivas son muy similares a las pasivas, salvo

que incorporan además una pequeña batería. Esta batería permite al circuito integrado en la etiqueta estar constantemente alimentado, y elimina la necesidad de diseñar una antena para recoger potencia de una señal entrante (si bien la antena es necesaria para la transmisión/recepción de las señales). Estas etiquetas responden más rápidamente, por lo que son más fuertes en el ratio de lectura en comparación con las etiquetas pasivas. Asimismo, también disponen de un rango de lectura (distancia) ligeramente superior al de las etiquetas pasivas. Las etiquetas RFID activas poseen una fuente de alimentación propia y alcanzan rangos de lectura mucho mayores que las etiquetas pasivas y semipasivas. Así, este tipo de etiquetas pueden ser leídas a distancias superiores a los 100 metros. Su tamaño es lógicamente mayor que el de los otros dos tipos, aunque puede ser inferior el tamaño de una moneda. Estas etiquetas activas pueden incorporar también una pequeña memoria adicional que les permite almacenar mayor cantidad de información. La gran limitación que presenta este tipo de etiquetas es que su funcionamiento está ligado a su batería, por lo que su tiempo de vida se limita a la vida de la batería.[28]

2.3.2.4 Tipos de sistemas rfid

Según la frecuencia de trabajo que se utilice en un sistema RFID tendremos distintos tipos de sistemas, lo que a su vez dará lugar a significativas diferencias funcionales. La frecuencia de trabajo es, de este modo, un factor crítico que determina en gran medida el comportamiento de un sistema RFID en cuanto a su manera de funcionar, sus capacidades y sus aplicaciones.

Así, tenemos los siguientes tipos de sistemas RFID:

- Sistemas RFID de baja frecuencia, conocidos como sistemas RFID LF (Low Frequency). Su frecuencia de funcionamiento, en Europa 8, es 125KHz.
- Sistemas RFID de alta frecuencia, conocidos como sistemas RFID HF (High Frequency). Su frecuencia de funcionamiento es 13,56 MHz.
- Sistemas RFID UHF, son aquellos sistemas que trabajan en frecuencias de la banda UHF 9 (Ultra High Frequency). Los sistemas RFID UHF trabajan, en España, en 869 MHz 10.
- Sistemas RFID de microondas, son aquellos sistemas RFID que trabajan a frecuencias mayores de los 2,4 GHz. Estos sistemas actualmente son muy poco utilizados. [29]

FRECUENCIAS RFID

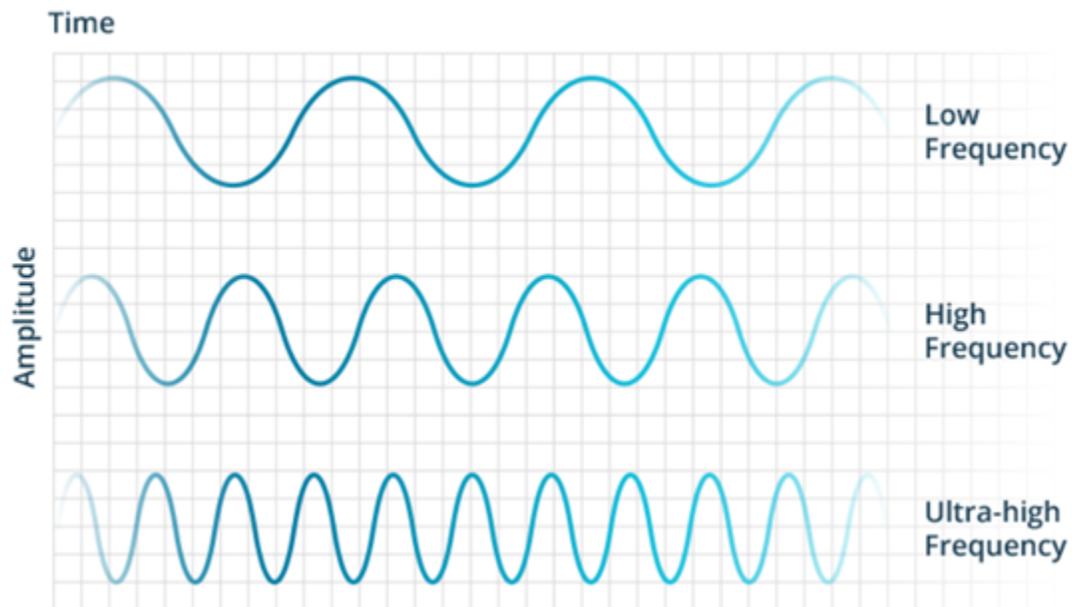


Imagen 9. (Frecuencias RFID)

La frecuencia hace referencia al tamaño de onda usado para comunicarse entre los componentes. Los sistemas RFID que existen en el mundo operan en baja frecuencia, alta frecuencia o hiper alta frecuencia. Las ondas de radio son diferentes en estas frecuencias y hay ventajas o desventajas al utilizar estos anchos de banda.

Por ejemplo, un Sistema RFID de baja frecuencia, tiene menos capacidad de transmisión de datos, pero aumenta la capacidad de ser leído cerca del metal o líquidos. Si un sistema opera en una frecuencia más alta, generalmente transmiten datos de manera más rápida y a más distancia de detección, pero las ondas de radio son más sensibles a interferencias causadas por líquidos y metales en el ambiente.

Sin embargo, las últimas innovaciones tecnológicas en los últimos años permiten que los sistemas RFID UHF (ultra-high frequency) sea posible utilizarlos en entornos con líquidos y con metales.

2.3.2.5 Baja frecuencia (lf) rfid

La banda LF cubre frecuencias entre 30 KHz a 300 KHz. Los sistemas típicos de LF RFID funcionan con 125 KHz o 134 KHz. Esta frecuencia proporciona un rango de lectura corto, unos 10 cm, y la velocidad de lectura es lenta. Resiste mucho a las interferencias externas. Las aplicaciones RFID típicas de LF son el control de accesos y el control animales. Las

normas estándar para sistemas de trazabilidad para animales están definidas en la ISO 14223, y ISO/IEC 18000-2. El espectro LF no es considerado una frecuencia para aplicaciones globales debido a las diferentes frecuencias y potencias de lectura en las que se trabaja alrededor del mundo.

2.3.2.6 Alta frecuencia (hf) rfid

Los rangos de las frecuencias HF van de 3 a 30 Mhz. La mayoría de sistemas RFID HF funcionan con 13,56 Mhz con rangos de lectura entre 10 cm y 1m. Las interferencias afectan de manera moderada a los sistemas HF. Los sistemas HF son comúnmente usados para ticketing, pagos y aplicaciones de transferencia de datos.

Hay unos cuantos estándar para HF RFID, la ISO 15693 es el estándar para la trazabilidad de objetos, el ECMA-340 y ISO/IEC 18092 son para el NFC (Near Field communication), una tecnología con ratio de lectura corto usado para el cambio de datos entre aparatos. Los estándares MIFARE son la ISO/IEC 14443 A y ISO/IEC 14443, que se utiliza en los smart cards, y los JIS X 6319-4 para FeliCa que se utiliza normalmente en las tarjetas con sistemas de pago.

2.3.2.7 Ultra-alta frecuencia (uhf) rfid

Los sistemas UHF cubren rangos de frecuencia desde 300Mhz a 3Ghz. Los sistemas RAIN RFID cumplen con la norma UHF Gen2 estándar que usa las frecuencias 860 a 960 Mhz. Hay diferencias de variación entre regiones, la mayoría de ellas operan entre 900 y 915 Mhz.

Los sistemas de lectura RFID UHF pueden llegar a más de 12 metros, tienen una transmisión de datos muy rápida y son muy sensibles a interferencias. Pero hoy en día, la mayoría de fabricantes de productos RFID como Dipole, hemos encontrado la manera de diseñar tags, antenas y lectores que dan un alto rendimiento en entornos complejos. Los tags UHF son más fáciles y económicos de fabricar comparados con los LF y HF.

Los sistemas RAIN RFID UHF son utilizados en una gran variedad de aplicaciones. Desde inventarios en tiendas hasta la identificación de medicamentos para su protección. La mayoría de proyectos RFID actualmente utilizan la tecnología UHF (RAIN RFID), convirtiendo está en el segmento de mercado que más crece.

La frecuencia UHF es regulada por un estándar global llamado EPCglobal Gen2 (ISO 1800-63) estándar UHF. Impinj y Smartrac partners principales de Dipole RFID, son dos de los grandes propulsores del RFID a nivel mundial desarrollando soluciones universales para que pueda adaptarse el RFID en la mayoría de sectores.

	Active RFID	Passive RFID	Battery-Assisted Passive (BAP)
Tag Power Source	Internal to tag	Energy transfer from the reader via RF	Tag uses internal power source to power on, and energy transferred from the reader via RF to backscatter
Tag Battery	Yes	No	Yes
Availability of Tag Power	Continuous	Only within field of reader	Only within field of reader
Required Signal Strength from Reader to Tag	Very Low	Very high (must power the tag)	Moderate (does not need to power tag, but must power backscatter)
Available Signal Strength from Tag to Reader	High	Very Low	Moderate
Communication Range	Long Range (100m or more)	Short range (up to 10m)	Moderate range (up to 100m)
Sensor Capability	Ability to continuously monitor and record sensor input	Ability to read and transfer sensor values only when tag is powered by reader	Ability to read and transfer sensor values only when tag receives RF signal from reader

Imagen 10. (SISTEMAS RFID PASIVOS, ACTIVOS Y BAP)

Los sistemas RFID activos los tags transmiten su propia señal con la información que tienen almacenada en el chip porque tienen potencia propia. Normalmente esta fuente de potencia son baterías. Usualmente los sistemas RFID activos operan en frecuencias UHF y ofrecen un rango de lectura de más de 100 metros. Suelen usarse en objetos muy grandes como vagones, contenedores o productos que tienen que estar controlados en grandes espacios. [29]

2.3.2.8 Funcionamiento de un sistema rfid

Como vimos en el apartado anterior, la tecnología RFID se basa en dos principios fundamentales distintos según sea la banda de frecuencia utilizada en la transmisión para la comunicación entre lector y antena:

El acoplamiento de inducción se usa tanto para comunicaciones a baja frecuencia (LF) como a alta (HF). La corriente eléctrica que circula por la antena del lector genera un campo magnético que, cuando llega a la antena de la etiqueta, induce en ésta una corriente que la alimenta. El tag conmuta entonces la impedancia de carga de su antena para crear una modulación que le permita la transmisión de datos. Es decir, el tag es capaz, con la energía inducida por el campo magnético generado por el lector, de modular una onda electromagnética con la señal que tiene almacenada y transmitirla.

El acoplamiento capacitivo o backscattering se usa para la comunicación en frecuencias UHF y microondas. En este caso, el lector transmite una señal de radiofrecuencia que la etiqueta recibe, modula y refleja de nuevo hacia el lector. Dependiendo del tipo de alimentación de

las etiquetas (pasivas o activas), éstas tomarán de la señal que les llega del lector su alimentación o no, antes de retransmitirla en respuesta. Es decir, a diferencia del acoplamiento inductivo, aquí no se genera una nueva onda electromagnética sino que se refleja la señal recibida por el lector. [30]

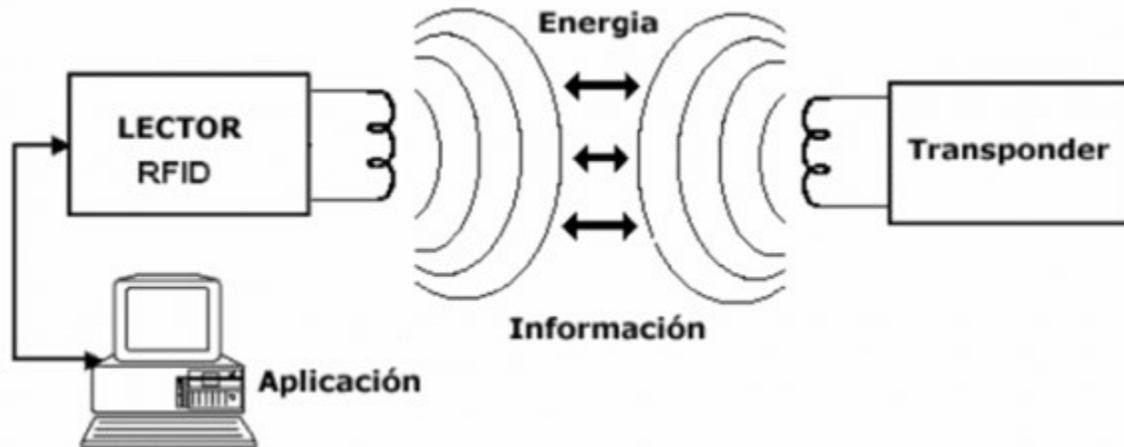


Imagen 11. (Funcionamiento de un sistema RFID)

2.3.3 Comunicación serial

Los puertos serie son la forma principal de comunicar una placa Arduino con un ordenador. Gracias al puerto serie podemos, por ejemplo, mover el ratón o simular la escritura de un usuario en el teclado, enviar correos con alertas, controlar un robot realizando los cálculos en el ordenador, encender o apagar un dispositivo desde una página Web a través de Internet, o desde una aplicación móvil a través de Bluetooth.

Existen un sin fin de posibilidades en las que se requiere el empleo del puerto serie. Por tanto el puerto serie es un componente fundamental de una gran cantidad de proyectos de Arduino, y es uno de los elementos básicos que debemos aprender para poder sacar todo el potencial de Arduino.

En esta entrada aprenderemos el funcionamiento básico de los puertos serie en Arduino. Al final de la entrada se adjuntan varios códigos de ejemplo, pero antes conviene explicar brevemente algo de teoría sobre qué es un puerto serie, y algunos términos que necesitaremos para entender correctamente el funcionamiento del puerto serie.

2.3.3.1 Arduino y puerto serie

Prácticamente todas las placas Arduino disponen al menos de una unidad UART. Las placas Arduino UNO y Mini Pro disponen de una unidad UART que operan a nivel TTL 0V / 5V, por lo que son directamente compatibles con la conexión USB. Por su parte, Arduino Mega y Arduino Due disponen de 4 unidades UART TTL 0V / 5V.

Los puertos serie están físicamente unidos a distintos pines de la placa Arduino. Lógicamente, mientras usamos los puertos de serie no podemos usar como entradas o salidas digitales los pines asociados con el puerto serie en uso. En Arduino UNO y Mini Pro los pines empleados son 0 (RX) y 1 (TX). En el caso de Arduino Mega y Arduino Due, que tienen cuatro puertos de serie, el puerto serie 0 está conectado a los pines 0 (RX) y 1 (TX), el puerto serie 1 a los pines 19 (RX) y 18 (TX) el puerto serie 2 a los pines 17 (RX) y 16 (TX), y el puerto serie 3 a los pines 15 (RX) y 14 (TX).

Muchos modelos de placas Arduino disponen de un conector USB o Micro USB conectado a uno de los puertos de serie, lo que simplifica el proceso de conexión con un ordenador. Sin embargo algunas placas, como por ejemplo la Mini Pro, prescinden de este conector por lo que la única forma de conectarse a las mismas es directamente a través de los pines correspondientes.

Un puerto Serial es un módulo de comunicación digital para un sistema embebido. Es decir, permite la comunicación entre dos dispositivos digitales. Cuenta con dos conexiones, RX y TX. Lo que nos indica los modos de comunicación que puede manejar, Full-duplex, Duplex y Simplex. Además podemos considerar como su principal ventaja a la sencillez de su protocolo de comunicación. Sin embargo también tiene desventajas como que sólo se puede comunicar a un puerto dos dispositivos.

- Full duplex. Significa que puede recibir y enviar información digital simultáneamente.
- Duplex o Half-duplex. Es cuando sólo podemos transmitir o recibir información, una cosa a la vez.
- Simplex. Cuando sólo podemos ya sea recibir o transmitir.

La función principal de un puerto serial, es la de empaquetar y des-empaquetar paquetes de datos binarios seriales. Como resultado, la serialización significa convertir un dato paralelo (byte) a un conjunto de pulsos seriales que puedan ser recibidos y enviados por una línea de transmisión. En primer lugar, el protocolo serial opera mediante tres condiciones digitales básicas: inicio de transmisión (IT), paridad (P) y fin de transmisión (FT). Estas condiciones son sincronizadas mediante un oscilador interno. El generador permite controlar la velocidad del puerto serial. Por lo tanto, la velocidad se mide en BAUD 's. Al módulo serial también se le conoce como UART ó USART o EUSART.

- UART – Universal Asynchronous Receiver and Transmitter que en español se traduciría como Transceptor Asíncrono.
- USART – Universal Synchronous and Asynchronous Receiver and Transmitter, que significa en español Transceptor Síncrono y Asíncrono.
- EUART – Enhanced Universal Asynchronous Receiver and Transmitter o Transceptor Asíncrono Universal Mejorado.

2.3.3.2 Estructura interna y configuración de un puerto serial

Una UART contiene, en su estructura interna, un generador de paridad, registros de corrimiento, oscilador variable (para generar el BAUD), verificadores de las tres condiciones y lógica de control. Por consiguiente, la Figura, muestra un diagrama a bloques general para una UART. Un paquete de datos se transmite a través de un registro de corrimiento. Por lo tanto, la velocidad a la que se transmite, está controlada por el generador de BAUD. La lógica de control se encarga de agregar los bits de Inicio, Paridad y de Fin de transmisión. El proceso de recepción serial es lo opuesto.

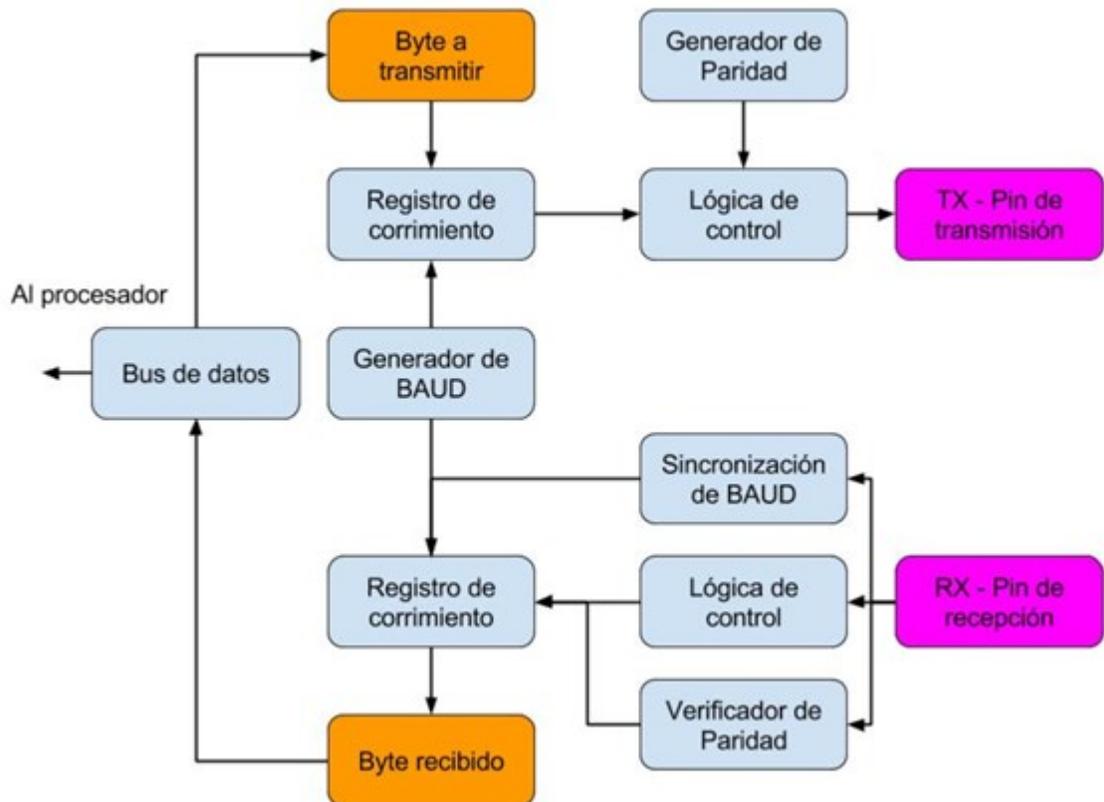


Imagen 16. Estructura interna y configuración de un puerto serial.

2.3.4 Arduino

Arduino es una plataforma de hardware libre, basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios.

Por otro lado Arduino nos proporciona un software consistente en un entorno de desarrollo (IDE) que implementa el lenguaje de programación de Arduino y el bootloader ejecutado en la placa. La principal característica del software de programación y del lenguaje de programación es su sencillez y facilidad de uso

¿Para qué sirve Arduino? Arduino se puede utilizar para desarrollar elementos autónomos, conectándose a dispositivos e interactuar tanto con el hardware como con el software. Nos sirve tanto para controlar un elemento, pongamos por ejemplo un motor que nos suba o baje una persiana basada en la luz existente es una habitación, gracias a un sensor de luz conectado

al Arduino, o bien para leer la información de una fuente, como puede ser un teclado, y convertir la información en una acción como puede ser encender una luz y pasar por un display lo teclado.

2.4 Visual C#

2.4.1 Programación Orientada a Objetos (POO)

2.4.1.1 Clases y Objetos

En este lenguaje se combina la estructura y el comportamiento de los datos en una única entidad, los objetos; que a su vez sólo tienen sentido en una aplicación. Los objetos contenidos en una clase poseen las mismas responsabilidades, así mismo, si varios objetos tienen las mismas responsabilidades, se pueden agrupar en una sola clase; y que las clases se utilizan para generalizar el dominio de los objetos.

Los objetos tienen al igual atributos como la identidad, que es inherente, para no mezclar objetos que puedan tener las mismas características pero no son el mismo. Ejemplo, dos peones de ajedrez.

2.4.1.2 Responsabilidades

Las responsabilidades podemos dividir las en dos grupos, los estados que son atributos y los métodos que determinan el comportamiento. Los atributos son valores que los objetos pertenecientes a una clase almacenan, mientras los métodos son pasos secuenciales que definen una acción o comportamiento; estos métodos solo se pueden llamar o ejecutar dentro de la misma clase.

2.4.1.3 Relaciones entre clases

Las relaciones son conexiones entre elementos como dependencias, asociaciones y las generalizaciones como se muestra en la imagen 17.

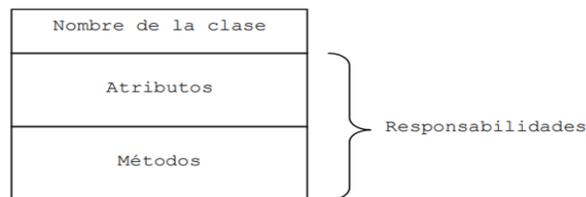


Imagen 17. Esquema general de clases.

2.4.1.4 Dependencia o uso

Las dependencias son aquellas que establecen una relación o uso de un elemento que puede afectar a otro elemento que la utilice, aunque cabe aclarar que esta relación no necesariamente tiene que funcionar a la inversa. Las dependencias se utilizan para definir elementos que utilizaran a otro para los métodos como se evidencia en la imagen 18.

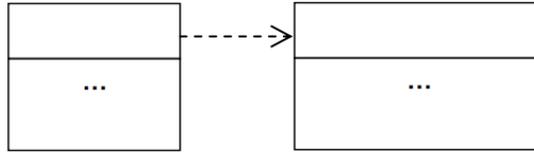


Imagen 18. Representación gráfica dependencia.

2.4.1.5 Asociación

Es la relación que se establece entre dos clases de forma estructural que especifica que los objeto de una clase están relacionados o conectados con los objeto de otra clase. Estas asociaciones son bidireccionales, por ende se les debe asignar un nombre y se leen en el sentido de la flecha como se muestra a continuación en la imagen 19.



Imagen 19. Representación asociación de clases.

Los números al lado de cada clase en la línea de la asociación representa la multiplicidad de la relación, siendo el 1 que significa que solo se le puede una vez a la otra clase, y la 1..* representa que la otra clase puede tener varias asociaciones.

2.4.1.6 Generalización

Esta relación determina las superclases, que igualmente que las clases hacen generalizaciones. Esto debido a que la generalización es base fundamental en los modelos orientados a objetos que interactúan entre sí para funcionar como un único sistema que es una habilidad lógica del pensamiento humano.

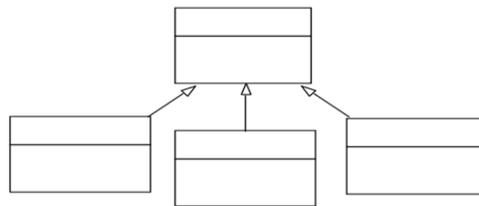


Imagen 20. Representación gráfica de relación de clases a superclases.

2.4.1.7 Clasificación atributos

Al mismo momento de creación y determinación de los objetos, se hizo una clasificación de los mismos implícita. De igual forma podemos clasificar atributos según los valores que ellos utilicen y sus operaciones definiendo el concepto de “tipo de dato” (cadena, numérico entero o real, lógico y conjunto de datos como clase).

2.4.1.8 Responsabilidades de objetos y Parametrización de métodos

Para que un objeto acceda a sus responsabilidades y métodos se utilizará la siguiente nomenclatura:

**<Objeto>.<responsabilidad>
<objeto>.<método>**

Para prevenir situaciones en donde la responsabilidad comparte el mismo nombre con un método, se utilizan los paréntesis al final del mismo “método ()” y de esta forma poder identificar y ser más específico si tiene más de un objeto asociado.

<Objeto x>.<método (objeto y)>

Llevando lo anterior al siguiente nivel se debe definir en cada clase la información de cada método para su correcta ejecución:

<Método> (<tipo de dato><identificado>)

2.4.1.9 Encapsulamiento

Tras un manejo de datos se hace evidente la necesidad de disponer un mecanismo que restrinja el acceso a componentes de los objetos de forma que garantice la integridad de los datos, esto se le llama encapsulamiento o encapsulación. Con lo anterior, se puede hacer un filtrado de información la cual se desea que el usuario pueda acceder o manipular.

2.4.1.10 Mensajes y Métodos

Como se ha reiterado anteriormente, la programación OO (Orientada a Objetos) es la interacción de objetos entre sí para solucionar un problema; la interacción entre los objetos se realiza mediante Mensajes, ya que estos especifican qué acciones se deben ejecutar, pero no el cómo.

Los métodos suelen entregar un resultado después de ejecutar su acción, así que también los podemos clasificar según su acción:

<Tipo retorno>.<identificador método>([<identificado>])

2.4.2 Tipos de Datos

Los datos utilizados en este lenguaje de programación se describen en la tabla a continuación:

Tipo	Descripción	Bits	Rango de Valores	Alias
System.SByte	Bytes con signo	8	[-128, 127]	sbyte
System.Byte	Bytes sin signo	8	[0, 255]	byte
System.Int16	Enteros cortos con signo	16	[-32.768, 32.767]	shor

System.UInt16	Enteros cortos sin signo	16	[0, 65.535]	ushort
System.Int32	Enteros normales	32	[-2.147.483.648, 2.147.483.647]	int
System.UInt32	Enteros normales sin signo	32	[0, 4.294.967.295]	uint
System.Int64	Enteros largos	64	[-9.223.372.036.854.775.808, 9.223.372.036.854.775.807]	ulong
System.UInt64	Enteros largos sin signo	64	[0, 18.446.744.073.709.551.615]	float
System.Single	Reales con 7 dígitos de precisión	32	[1.5×10^{-45} , 3.4×10^{38}]	double
System.Double	Reales de 15 a 16 dígitos de precisión	64	[5×10^{-324} , 1.7×10^{308}]	decimal
System.Decimal	Reales de 28 a 29 dígitos de precisión	128	[1×10^{-28} , 7.9×10^{28}]	bool
System.Boolean	Valores lógicos	32	True, False	char
System.Char	Caracteres Unicode	16	['\u0000', '\uFFFF']	string
System.String	Cadena de caracteres	Variable	Permitido por la memoria	

Tabla 2. Datos Básicos de C#

3. Metodología e implementación

La metodología e implementación del proyecto se basó en un seguimiento a los conceptos teóricos y prácticos mencionados anteriormente; es así como el hardware se compone de varios procesos los cuales se relacionan directamente con parte del software permitiendo obtener los datos exactos de los productos, alimentar la base de datos y obtener el control total del inventario generando reportes sobre la situación actual en que se encuentran, por medio de la información de la base de datos.

El primer paso fue el modelado y diseño de la arquitectura del presente software, se tuvieron en cuenta los requisitos funcionales, planteados como objetivos específicos y plasmados en el documento del anteproyecto, el cual fue aprobado. Los requisitos funcionales determinan el modelado de la base de datos, el del sistema de lectura RFID y el sistema de GUI (Interfaces Gráficas de Usuario). Para la implementación del proceso conocido y modelado se establecieron como un sistema en cascada definiendo el sistema con actores responsables de componentes que comprenden la arquitectura del software.

3.1 Modelamiento y diseño de la arquitectura del presente software

En el modelado de la base de datos se plantearon los actores principales de interacción con el software los cuales se identificaron como:

- Producto
- Usuarios
- Registros de Flujo

Después de identificados se implementaron los atributos según las características de cada uno. El Producto se estableció con los atributos de Nombre de Producto, ID, Ubicación, Cantidad y Familia, tal como se muestra en la *Imagen 21*.



Formulario de atributos Actor Producto desde la GUI Productos. El formulario contiene cinco campos de entrada de texto, cada uno con un label a la izquierda:

- NOMBRE
- ID
- UBICACION
- CANTIDAD
- FAMILIA

Imagen 21. Atributos Actor Producto desde la GUI Productos.

El Usuario se estableció con los atributos de Nombre de Usuario, Nombre de Login, Cédula, Contraseña y Edad, como se muestra en la *Imagen 22*.

Nombre	<input type="text"/>
Cedula	<input type="text"/>
Edad	<input type="text"/>
Usuario	<input type="text"/>
Contraseña	<input type="text"/>

Imagen 22. Atributos Actor Usuario desde la GUI usuario.

El Registro de Flujo como no contó con interfaz independiente para la adquisición de datos para la tabla en la base datos, se implementaron atributos desde la BD y estos fueron Número de Flujo (ID), Fecha de Flujo, Usuario que realizó el flujo, Tipo de Flujo, Id del producto, Nombre del Producto, Ubicación del producto, Cantidad del producto y Familia del Producto, en este último actor, no todos los atributos eran Not Null, es decir que dependiendo el tipo de registro de flujo existían algunos vacíos como se muestra en la *Imagen 23*.

idflujo	fecha	usuario2	tipo	id2	nombre2	ubicacion2	cantidad2	familia2
10	08-07-2018 11:11:06 a. m.	berbeoa	Consulta Flujo					
11	08-07-2018 01:11:57 p. m.	carmen	Consulta Flujo	388EC949				
12	08-07-2018 01:20:15 p. m.	berbeoa	Buscar Producto	50859275				
13	08-07-2018 01:21:16 p. m.	berbeoa	Buscar Producto	50859275				
14	08-07-2018 01:21:27 p. m.	berbeoa	Buscar Producto	50859275				
15	08-07-2018 01:22:18 p. m.	berbeoa	Eliminar Producto	50859275				
16	08-07-2018 01:24:19 p. m.	berbeoa	Adeorar Producto	50859275	rodillo ti	casillero 2	4	EPSON
17	08-07-2018 01:26:04 p. m.	berbeoa	Modificar Producto	50859275	rodillito	casillero 23	4	EPSON...

Imagen 23. Atributos Actor Flujo.

Con los atributos establecidos se generó el diagrama entidad relación, y las llaves principales y secundarias del modelamiento y normalización de la Base de Datos. Las llaves principales fueron ID para el Producto, esta fue la lectura Hexadecimal de las tarjetas RFID. Para el Usuario la llave primaria fue su Login, y para el Registro Flujo fue el Id registro, que fue auto incremental. Como llaves secundarias se aplicaron sólo dos para el actor Registró Flujo desde los actores sobrantes, la primera llave secundaria fue Login Usuaría, continuada de ID del producto, tal como se muestra en la *Imagen 24*.

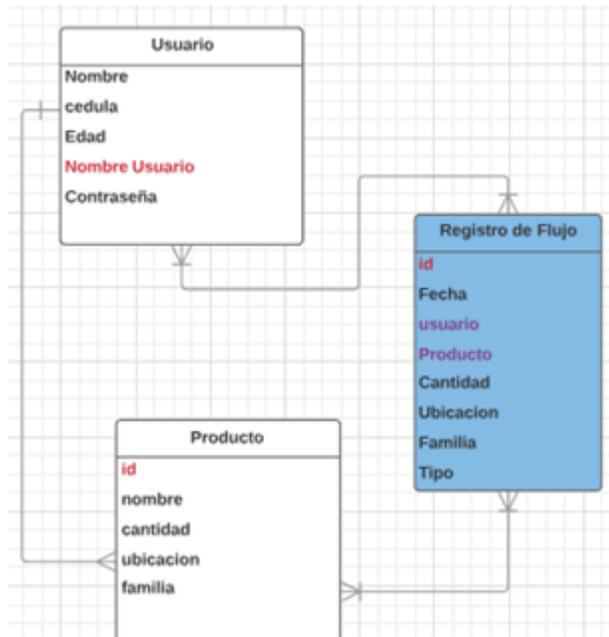


Imagen 24. Modelado BD, Diagrama entidad relación y Keys.

Para los actores de diseñaron métodos especializados, según correspondan a las actuaciones respectivas. El Producto podría usar los métodos Agregar, Eliminar, Buscar, Modificar, Salida de unidades e Ingreso de unidades, esto se puede visualizar en la *Imagen 25*.

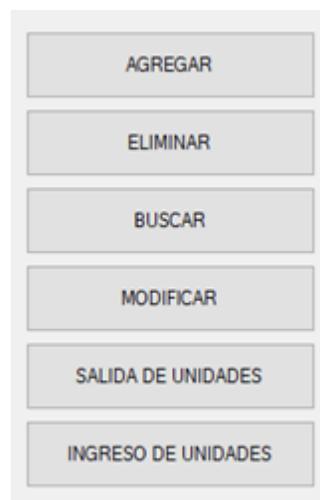


Imagen 25. Métodos Actor Producto desde la GUI Producto.

El Actor Usuario tenía a su funcionalidad dos métodos, Registrar Usuario y eliminar usuario, la *Imagen 26* lo muestra.



Imagen 26. Métodos de Actor Usuario.

El Actor Registro de Flujo solo posee un método llamado Consulta de flujo y se emplea con el siguiente botón

3.1.1 Sistema de Lectura RFID

En el inicio se debe realizar son los registros de los TAG, pasándose por el lector RFID, cada TAG tiene un código hexadecimal único para cada etiqueta, luego cada TAG es asignado a un elemento del inventario para identificarlo, especificando los datos de cada elemento como nombre, ubicación, familia a la que pertenece y referencia por catálogo, cada TAG es ideal para cada elemento del inventario, lo anterior significa que no pueden existir 2 asignaciones para un mismo TAG.

Luego concurre un proceso informático interactivo que está ejecutando permanentemente, a la espera de un dato con identificación RFID y pasarlo por el servidor. Al momento que el TAG pasa por el lector, se analiza la validez de la información. Si no es válido, la información es completamente desechada y se continúa a la espera de un dato válido. Si el código que pasa es validado, el código recibido es derivado a un canal y llevado a la base de datos, para poder liberar el proceso y poder capturar otros TAGs hay que pasar las etiquetas nuevamente sobre el lector RFID al momento de inicializar la interfaz gráfica. Si este paso no se realiza, se corre el riesgo de que el proceso central quedé esperando alguna respuesta de otros procesos, como conexiones a base de datos, momento en el cual se pueden perder otros TAGs que pasen por el portal hasta que llegue un Timeout, o el sistema completo quede congelado a la espera de una respuesta que nunca llegue.

El motivo de utilización de la tecnología RFID con las etiquetas (TAGS), reside en la necesidad de hacer un control de inventario ergonómico y didáctico para los diferentes usuarios, la calidad, eficiencia y rapidez al momento de adquirir los datos es la ventaja que tiene esta tecnología. Está soportada por un sistema de gestión de información mucho más avanzada a diferencia de los tags con código de barras, que puede generar fallas cuando se hace la lectura y es más complejo la captura de datos, generando conflictos en la adquisición de datos.

Las características por las que se toma la decisión tecnología RFID, es la facilidad de operar y diferentes aplicativos que se le puede dar a esta tecnología, haciendo algunos cambios al proyecto tales aplicativos como, control de acceso a personal, control de acceso vehicular en parqueaderos, control en bibliotecas, entre otros.

Para la lectura de las tarjetas se implementó con un microcontrolador común denominado Arduino Mega y un lector RFID RC522, con conexión tal como se muestra a continuación:

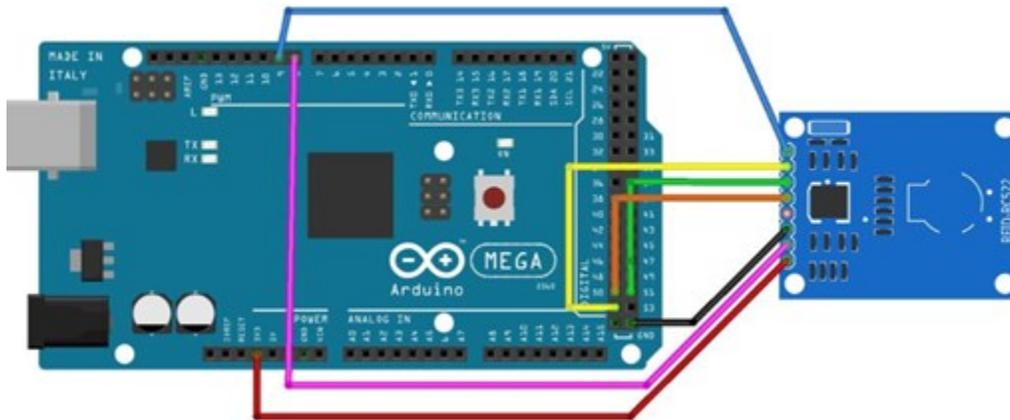


Imagen 27. Conexión Arduino Mega y lector RFID.

Se empleó el siguiente código, importando la librería establecida.

```
#include <SPI.h>
#include <MFRC522.h>
#define RST_PIN 5
#define SS_PIN 53
MFRC522 mfc522(SS_PIN, RST_PIN);
MFRC522::MIFARE_Key key;
void setup() {
  Serial.begin(9600);
  SPI.begin();
  mfc522.PCD_Init();
}
void loop() {
  if (! mfc522.PICC_IsNewCardPresent())
    return;
  if (! mfc522.PICC_ReadCardSerial())
    return;
  dump_byte_array(mfc522.uid.uidByte, mfc522.uid.size);
  mfc522.PICC_HaltA();
  mfc522.PCD_StopCrypto1();
}
void dump_byte_array(byte *buffer, byte bufferSize) {
  for (byte i = 0; i < bufferSize; i++) {
    Serial.print(buffer[i] < 0x10 ? "0" : "");
    Serial.print(buffer[i], HEX);
  }
  Serial.println();
}
```

3.1.2 Sistema de GUI'S (Interfaces Gráficas de Usuario)

En la parte de diseño de interfaces gráficas de usuario, se diseñó la menor cantidad posible para el prototipo funcional, pero se plantea ampliar estas con los requerimientos no funcionales en una segunda versión.

Se generaron dos interfaces, una diseñada para el acceso al inventario y otra que realiza el manejo de este. A estas se les denominó GUI de producto y GUI de usuario. Para el diseño de las interfaces se empleó el lenguaje C#, y se usaron todos los atributos planteados en la fase de modelamiento y arquitectura. En su primera versión las interfaces interactuantes del sistema son:



Imagen 28. GUI Usuario.

En la *Imagen 28* están los atributos y métodos del actor Usuario, adicional está el método Login para hacer la validación de acceso a usuarios registrados.

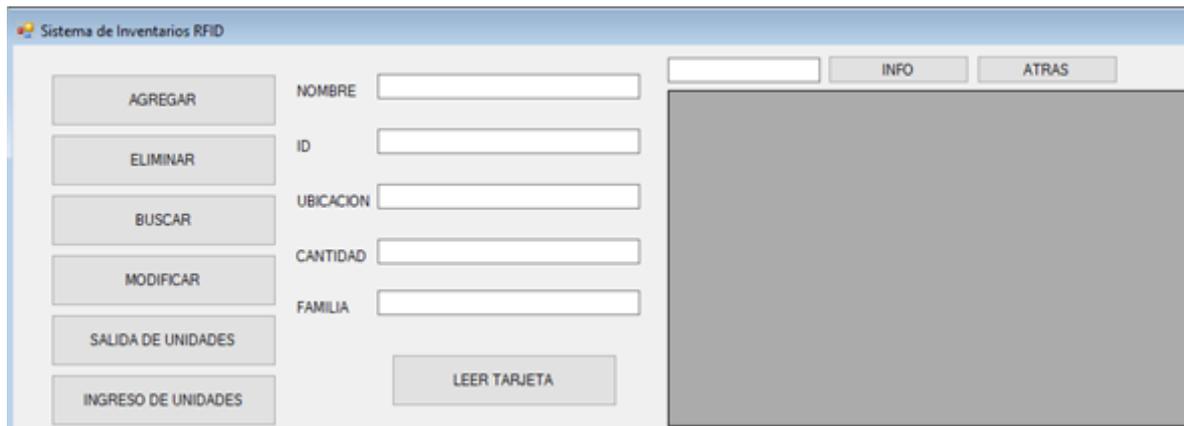


Imagen 29. GUI Producto.

En la *Imagen 29* están todos los atributos y métodos del actor Producto, adicionalmente está el método leer tarjeta que habita el sistema de lectura RFID y el método INFO, donde se genera el registro tipo consulta. Adicionalmente esta interfaz tiene un botón de ATRÁS, para salir de sección.

El siguiente paso se enfocó en el acople de las diferentes interfaces de usuarios, las validaciones de las GUI y la dinamización de estas dentro del sistema, paralelamente a este paso se generaron los procedimientos almacenados en la base de datos en la plataforma Workbench, las tablas y las relaciones entre las mismas, adicionalmente se generó la lectura del RFID por medio del Arduino al PC.

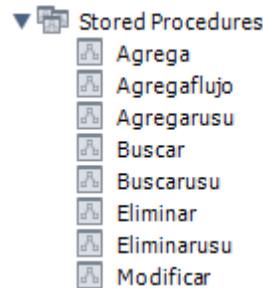


Imagen 30. Procedimientos Almacenados en Workbench.

El tercer paso fue el acople en cascada de los módulos planteados, el inicial fue la lectura del RFID con la GUI de productos del sistema de inventarios, continuamos con el acople de la BD (Base de Datos) en su tabla de usuarios y la GUI usuarios, eliminación y acceso de usuarios al sistema de inventarios. Para finalizar el tercer paso se documentó en la *Imagen 31*.

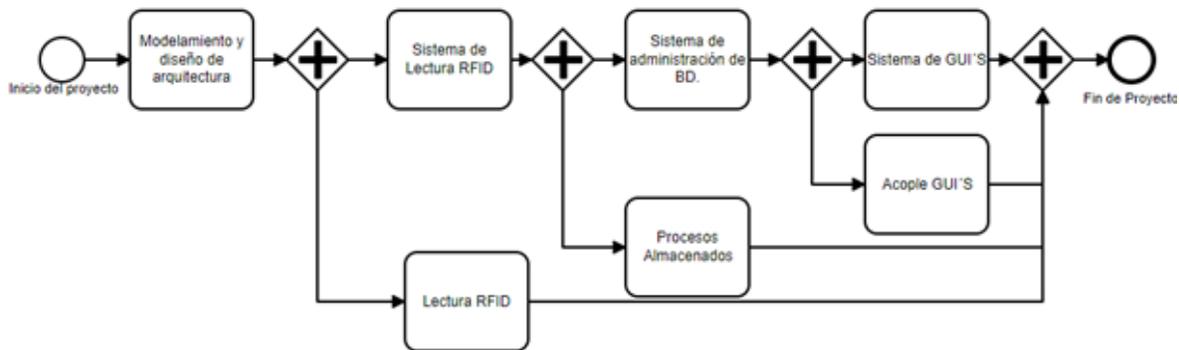


Imagen 31. Diagrama Sistema de Inventarios.

3.2 Requerimientos

En el desarrollo del proyecto se evidenciaron características y opciones de mejora significativas que se pueden implementar a corto, mediano y largo plazo del proyecto, por lo que se delimitó unas funciones específicas que apuntan directamente al cumplimiento de los objetivos y otras que mejoran el funcionamiento de las mismas que nombraremos como específicas.

3.2.1 Funcionales

En los requerimientos funcionales que son cruciales en el funcionamiento adecuado y eficiente del proyecto desarrollado podemos mencionar:

- Interfaz gráfica para el uso del aplicativo.
- Gestión y validación de usuarios.
- Administración de partes y equipos en la bodega.
- Lectura de ID en los diferentes productos.
- Administración de equipos en el aplicativo.
- Registro de transacciones para su análisis.
- Accesibilidad remota.

3.2.2 Específicas

Al ser un prototipo, se evidencian aspectos de mejora que añadirán características que pueden impactar de forma positiva el funcionamiento general del aplicativo:

- Sistema de cuentas de usuario por jerarquía.
- Separación de interfaces según funcionalidad requerida.
- Exportación de registros según filtros por el usuario.
- Visualización gráfica de los equipos y partes.

3.3 Documentación

Se realiza manual de usuario sobre el aplicativo, documento empleado para capacitación de personal por perfil; dependiendo su área y cargo (Nivel operativo y gerencial). Anexo.

4. Resultados, análisis y obtención de Datos

4.1 Manejo Interfaz Gráfica

A continuación se explica las funciones y campos de la interfaz que utilizarán los usuarios para interactuar con el aplicativo:

Imagen 32. Campos de registro y eliminación de usuarios.

Para realizar un registro de un nuevo usuario o en su defecto efectuar una eliminación del mismo, se hace el registro completo de los campos solicitados en la Imagen 32 y dependiendo de la acción a ejecutar se selecciona el botón registrar o eliminar.

Imagen 33. Campos de validación de usuario y contraseña para el ingreso al aplicativo.

Para iniciar el ingreso al sistema de inventario hay que tener un usuario y una contraseña registrada en el paso anterior, sabiendo el usuario y la contraseña se llenan los campos seleccionados en la Imagen 33 y luego se procede a dar click en el botón LOGIN, así estamos accediendo a la siguiente interfaz donde se puede manipular el inventario y hacer más operaciones dependiendo de la necesidad de cada usuario.

Imagen 34. Interfaz de administración del aplicativo.

En la Imagen 34 se puede evidenciar que hay variedad de botón y de campos, cada uno con una función específica, diseñados para que sean de fácil interacción y uso para el usuario, generando mayor facilidad para ejecutar una acción.

Imagen 35. Botón Agregar, utilizado para añadir partes nuevas referenciadas en el inventario.



Imagen 36. El botón Leer Tarjeta.

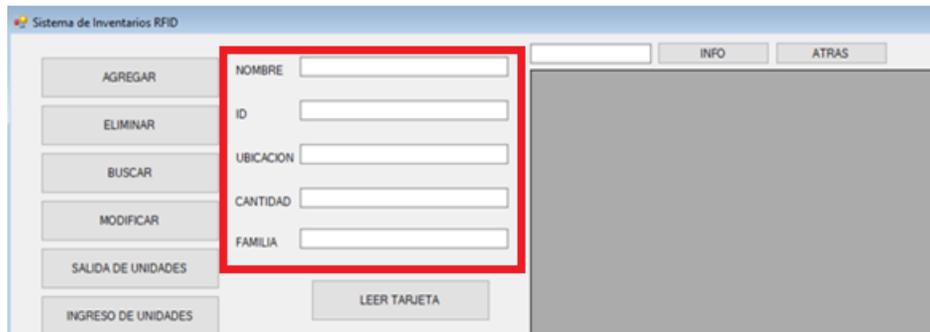


Imagen 37. Campos TextBox para visualizar información de componentes almacenados en el aplicativo y actualización de datos.

En la Imagen 35, encontramos señalado el primer botón, la acción para este botón es la de agregar, para agregar un producto damos click en el botón **LEER TARJETA** seleccionado en la Imagen 36, luego pasamos un TAG RFID sobre el lector RFID, donde este leerá un código hexadecimal y será asignado par alguno de los productos que se están almacenando en la base de datos, después de pasar el TAG POR el lector y evidenciar que fue leído, se llenan los datos requeridos en la interfaz, señalados en la Imagen 37, una vez llenados los campos con la información valida damos click en el botón **AGREGAR** para agregar un nuevo producto a la base de datos.

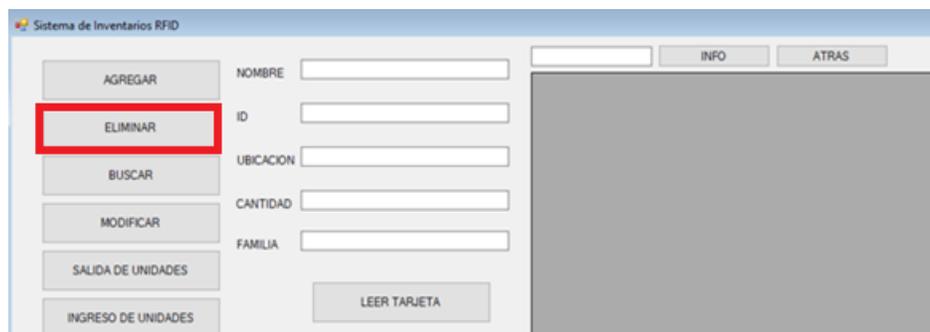


Imagen 38. Botón Eliminar, para borrar un componente relacionado con un ID.

Para realizar la eliminación de un producto, hay que dar click en el botón **LEER TARJETA** seleccionado en la Imagen 38 y luego pasar un TAG sobre el lector RFID, si el TAG ya fue

agregado luego de hacer la lectura en los campos seleccionado en la Imagen 37 aparecerá la información del producto que fue asignado para ese TAG, se le da click en el botón **ELIMINAR** y el producto que fue eliminado sale del listado de productos disponibles que están en la base de datos.



Imagen 39. Botón Buscar, para acceder a información relacionada con el ID una vez se ha leído la tarjeta RFID.

Para hacer la búsqueda de un producto en específico procedemos hacer el paso anterior, pero en vez de presionar el botón eliminar, se presiona el botón **BUSCAR** seleccionado en la Imagen 39, este botón nos va a permitir saber en qué posición se encuentra un producto o la cantidad de unidades que hay disponibles del producto en el stock de inventario.

Para realizar una modificación sobre un producto que ya fue añadido a la base de datos hacemos el paso anterior de **LEER TARJETA** y luego click en el botón **MODIFICAR**, este nos activar los campos para poder realizar las modificaciones que se necesitan sobre el producto, mejorando la confiabilidad de la base de datos.



Imagen 40. Salida e Ingreso de unidades para actualización de cantidades.

En la Imagen 40 están seleccionados 2 botón **SALIDA DE UNIDADES O INGRESO DE UNIDADES**, estos 2 botón me permiten el retiro de los productos o la entrada de productos, dependiendo de la actividad que quiera realizar se selecciona el botón adecuado, sabiendo las cantidades exactas.



Imagen 41. Botón Info, para la visualización de movimientos en la BD.

En el botón **INFO**, me permite saber todos los movimientos que he tenido en el inventario, teniendo conocimiento que usuarios ingresaron, que actividades realizaron mientras tuvieron la sección inicializada, actividades como ingresar productos, sacar productos, buscar cantidades disponibles o incluso modificaciones, como se ve en la Imagen 41.

Si se desea cerrar la sección damos click en el botón atrás y automáticamente se cerrara la sección que se tenía abierta y regresar a la anterior interfaz, con la disponibilidad de iniciar sección con otro usuario.

4.2 Validaciones de campos

Para el análisis del sistema se deben tener en cuenta correcto funcionamiento de hardware, la lectura del RFID desde el Pc, se realiza por medio de comunicación de puerto serial, por este motivo se debe verificar el funcionamiento de la comunicación con el puerto COM3, si no se hace la verificación de la comunicación el sistema no podrá hacer lectura del sistema RFID, adicionalmente se debe contar con espacio en el servidor bien sea parcial o total para el funcionamiento del sistema de inventarios RFID. Estos son los requisitos básicos para el funcionamiento del Sistema.

En la obtención de datos se provee procesos almacenados en MYSQL Workbench, que son específicos para el Sistema de inventarios por RFID, por tal motivo da seguridad que el sistema es confiable y no se puede clonar por personas externas al desarrollo de este. La adquisición de los datos se hace mediante una interfaz gráfica dependiendo la tabla de almacenamiento que se emplee, adicionalmente es automática la administración de datos en el sistema.

A continuación, se mostrará el funcionamiento del Sistema de inventarios por medio de RFID. El primer paso es conectar el lector RFID antes de iniciar el software, con esto garantizamos la comunicación de puerto COM3 con el microcontrolador y el subsistema lector RFID, tal como se muestra en la *Figura 32*. En próximas versiones se tendrá en cuenta el requerimiento no funcional referente a un dispositivo ergonómico para el subsistema de lectura RFID.

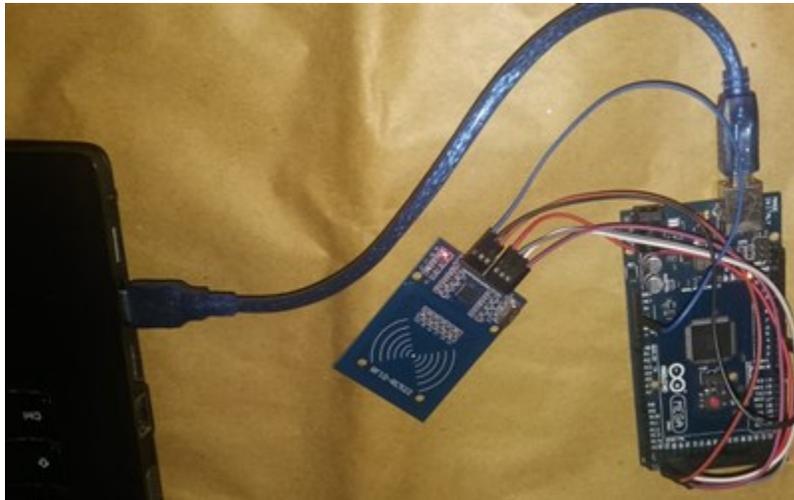


Imagen 42. Conexión puerto COM3, para inicio del sistema.

Después de entablar la conexión con el puerto COM3, se hace apertura del software para comprobar la comunicación del subsistema lector RFID con la GUI producto, esto se hace mediante el botón de “LEER TARJETA” mostrado en la Imagen 33, en caso de que la comunicación sea fallida, es porque no se reconoció la comunicación del puerto COM3 y se debe reiniciar el equipo al cual se le implementara el lector.

AGREGAR	NOMBRE	<input type="text"/>
ELIMINAR	ID	<input type="text" value="6260E62B"/>
BUSCAR	UBICACION	<input type="text"/>
MODIFICAR	CANTIDAD	<input type="text"/>
SALIDA DE UNIDADES	FAMILIA	<input type="text"/>
INGRESO DE UNIDADES		<input type="button" value="LEER TARJETA"/>

Imagen 43. Comprobación de comunicación con subsistema lectura RFID y GUI producto.

Para realizar el paso anterior se debe contar con acceso desde un usuario, todas las interfaces gráficas tienen un método de validación de campos para obtener los parámetros necesarios de los métodos característicos para cada actor. Si no se encuentran todos los campos Not Null diligenciados, el sistema generará una ventana emergente indicando de que falta diligenciar algún campo y no se puede continuar con la función seleccionada sin contar con estos. Igualmente, se generan ventanas emergentes si existen ingresos de datos no correspondientes al tipo de variables pertinentes de cada campo.

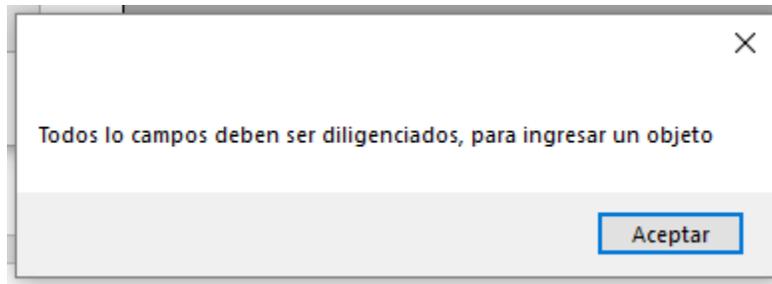


Imagen 44. Ejemplo ventana emergente, campos no diligenciados.

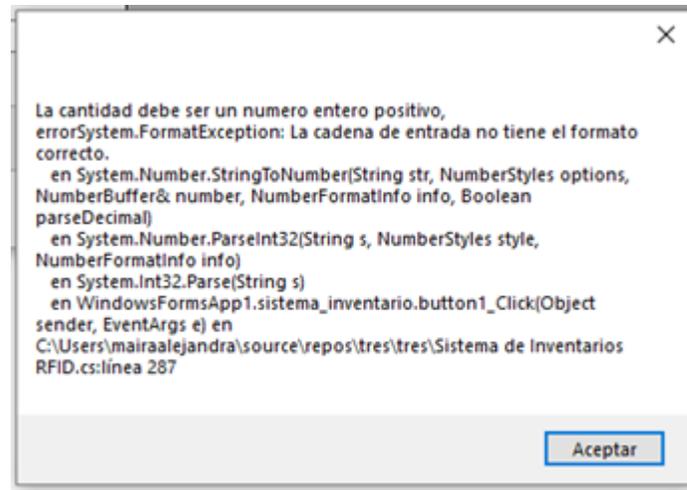


Imagen 45. Ejemplo ventana emergente datos no correspondientes.

De igual manera el sistema no deja duplicar las llaves primarias de las tablas en las BD, y genera alertas emergentes sobre la no ejecución cuando se emplean duplicidad.

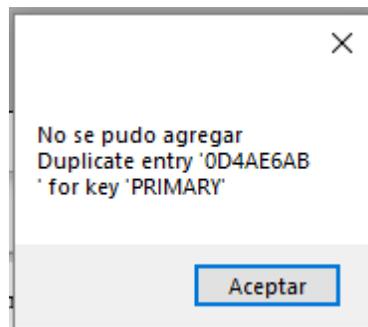


Imagen 46. Ejemplo ventana emergente, duplicidad de llave primaria.

Adicionalmente el Sistema de inventarios por RFID cuenta con una característica visual en tiempo real sobre la cual se miran los informes y el inventario actualizado, los flujos realizados y que usuarios lo realizaron, su puede ver esto en la Imagen 37.

carmen					
INFO					
ATRAS					
idflujo	fecha	usuario2	tipo	id2	
5	08-07-2018 10:16:11 a. m.	carmen	Consulta Flujo	3B8EC949	^
6	08-07-2018 10:16:41 a. m.	carmen	Agregar Producto	3B8EC949	
7	08-07-2018 10:16:52 a. m.	carmen	Eliminar Producto	3B8EC949	
8	08-07-2018 10:16:55 a. m.	carmen	Consulta Flujo		
9	08-07-2018 11:10:06 a. m.	carmen	Consulta Flujo		
10	08-07-2018 11:11:06 a. m.	berbeoa	Consulta Flujo		
11	08-07-2018 01:11:57 p. m.	carmen	Consulta Flujo	3B8EC949	
12	08-07-2018 01:20:15 p. m.	berbeoa	Buscar Producto	50859275	
13	08-07-2018 01:21:16 p. m.	berbeoa	Buscar Producto	50859275	
14	08-07-2018 01:21:27 p. m.	berbeoa	Buscar Producto	50859275	
15	08-07-2018 01:22:10 p. m.	berbeoa	Buscar Producto	50859275	v

Imagen 47. Parte Visual de la GUI sobre la BD.

4.3 Análisis y Obtención de Datos

Tras las distintas pruebas e implementación del proyecto en versión DEMO en las instalaciones de la empresa, se alimentaron registros de partes y equipos durante el mes de Julio con el fin de evidenciar los movimientos realizados para su posterior análisis como los siguientes aspectos:

4.3.1 Rotación de partes

En este módulo se evidenció que parte del inventario es la que más tiene rotación y al mismo tiempo las que no presentaron movimiento. Esto nos dio información partes y equipos que más están generando llamadas de servicio. Esto se considera importante ya que con un periodo mayor de estudio se lograría hacer un análisis en cuanto si es una falla de equipo-parte, o un procedimiento que pueda estar generando un cambio que no se prevé por el fabricante desde un principio; siendo una realimentación a la marca sobre sus productos.

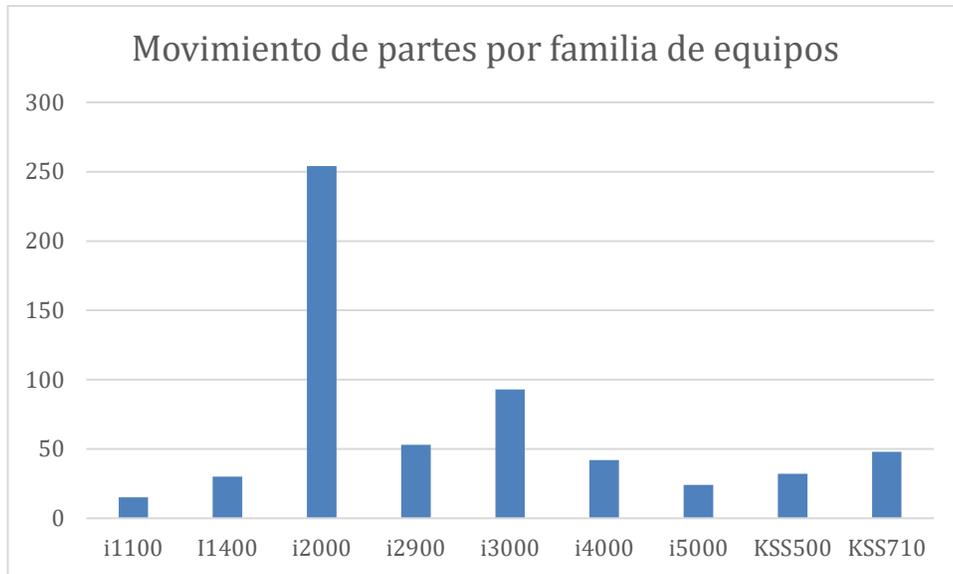


Imagen 48. Movimiento de partes por familia de equipos.

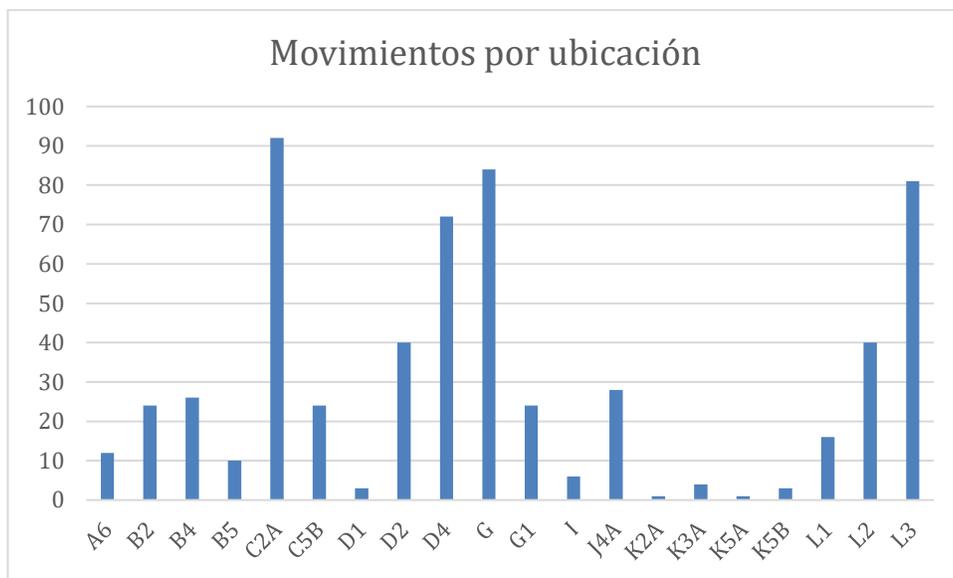


Imagen 49. Movimientos de partes por ubicación.

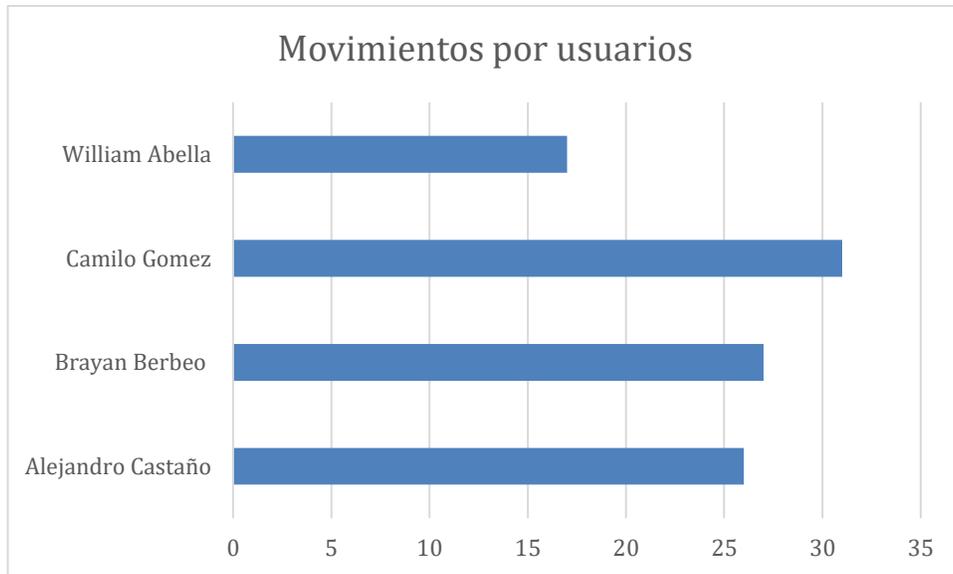


Imagen 50. Movimientos de partes por usuario.

4.3.2 Rotación Equipos

Al conocer cuales equipos son los que más movimientos presentan en el mercado y cuáles no, en un periodo mayor de muestra, se lograría evidenciar tendencias de clientes y sectores comerciales. Lo anterior se podría tomaría como una retroalimentación del mercado, tomando los datos y analizándolos para la creación de estrategias de marketing (campañas, promociones, etc.) que dan una ventaja competitiva si se llegan a aplicar de la forma adecuada y al sector indicado.

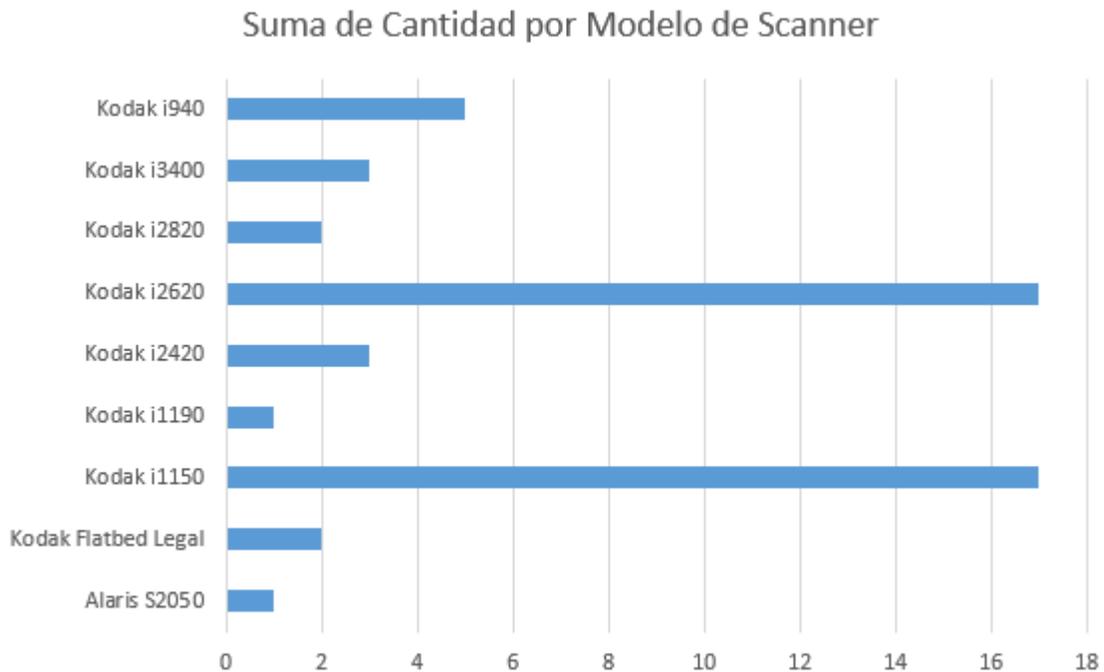


Imagen 51. Suma de equipos en venta.

4.3.3 Presupuesto

Una vez analizados los puntos anteriores ya se logró evidenciar que equipos y partes se necesitaban mantener en el inventario y en que posibles cantidades, previniendo escenarios como solicitud de partes agotadas, al igual que posesión varias partes de alto costo con rotación muy baja o nula; aplicando de igual medida a los equipos en venta y renta.

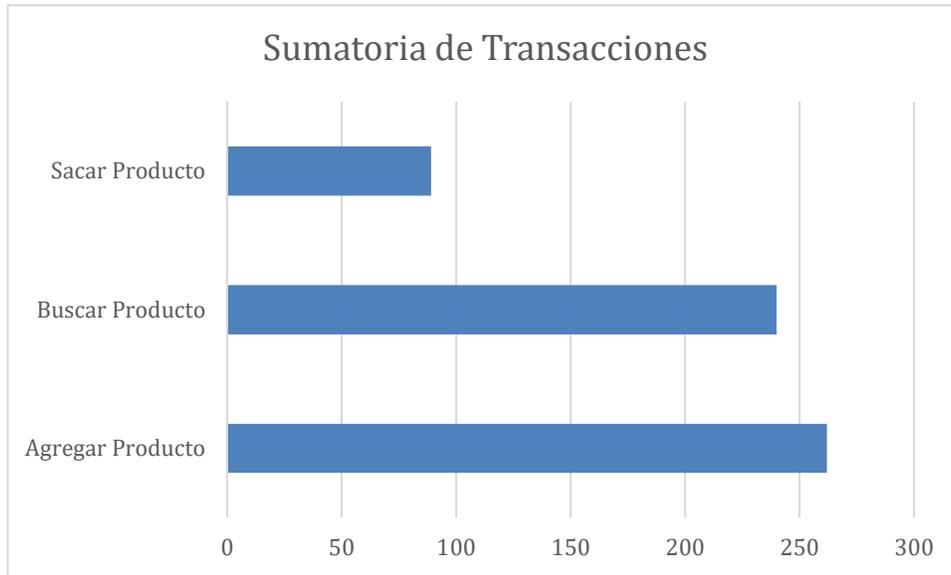


Imagen 52. Totalización de transacciones en el aplicativo

Esto afecta directamente el presupuesto que se maneja en la gerencia de servicio, mejorando el direccionamiento de los recursos en la empresa y a su vez moviendo el indicador de rotación de existencias que mide la eficiencia de una empresa en el manejo de los productos que vende o compra para revender.

5. Conclusiones

- Se logró implementar un sistema de posicionamiento que nos brinda información de los equipos y partes, con características como cantidades, ubicaciones, movimientos, etc., en el interior de la bodega.
- Se implementó un software de gestión que permite al usuario interactuar con la base de datos alimentando de información con validación de usuario y contraseña, generación de registros y validación de campos mediante tecnología RFID.
- Tras el análisis y rastreo de movimientos de los equipos y partes brindados por el aplicativo, sumando un periodo de tiempo mayor a 4 meses se lograrían evidenciar tendencias de movimientos con lo que la empresa estaría en posición de generar campañas comerciales para ciertos periodos de tiempo del año y realizar un análisis de presupuesto anual sobre esta área para el próximo año identificando en que momentos se deben incrementar inventarios o disminuirlos con el fin de mejorar el indicador de rotación.
- Con el control adecuado de inventario, el porcentaje de solicitudes de partes no disponibles disminuyó drásticamente gracias a la fiabilidad del software y practicidad al momento de realizar las transacciones con el aplicativo, ya que permite realizar una predicción de movimientos; sumándole la importancia del rastreo que se logra con los registros.
- Los costos de mantenimiento del software son muy bajos, ya que solo se contaría con horas ingeniero para soporte de la base de datos y la plataforma de desarrollo (Visual Studio); ambos software gratuitos.
- La comparativa de tiempos de consulta y actualización de datos de inventario se logra disminuir hasta en un 50% según repuesto. Esto debido a que cada parte ya tiene su identificador adherido, que se traduce en el reconocimiento de la misma por el aplicativo evitando acceder al manual y más archivos para completar una transacción.

6. Bibliografía

- [1]. Efrén Darío Acevedo Cárdenas, Javier Eduardo Arias Osorio, Jorge Hernando Ramón Suárez. Universidad Industrial de Santander. “Análisis de los beneficios de la identificación por radiofrecuencia en un centro de distribución textil colombiano” 10 de Febrero de 2015, Colombia. Recuperado 6 de Octubre de 2017. <http://www.unilibre.edu.co/revistaavances/edicion-11-2/analisis-de-los-beneficios-de-la-identificacion-por-radiofrecuencia-en-un-centro-de-distribucion-textil-colombiano.pdf>
- [2]. David Chang Falconi, Alan Lozano Solís, Universidad Politécnica Salesiana-Sede Guayaquil. “DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA PARA EL CONTROL E INVENTARIO CONTINUO, UTILIZANDO TECNOLOGÍA RFID, PARA LA BIBLIOTECA DE LA UPS SEDE GUAYAQUIL” Agosto de 2013, Ecuador. Recuperado 6 de Octubre de 2017.
- [3]. Cristhian Peter Alejandro Meneses. Pontificia Universidad Católica de Perú, “Diseño de un sistema de control de activos para el almacén de electrónica de la pontificia universidad católica del Perú utilizando RFID” (Agosto de 2012). Lima Perú. Recuperado 6 de Octubre de 2017. <http://tesis.pucp.edu.pe/repositorio/handle/123456789/1507>

- [4]. Santa Rosa Corro García. Universidad Nacional Autónoma de México. “Diseño de una metodología para evaluar la implementación de un sistema RFID en el proceso de gestión de inventarios dentro del sector de servicios” (2012). México. Recuperado 6 de Octubre de 2017.
<http://www.ptolomeo.unam.mx:8080/xmlui/bitstream/handle/132.248.52.100/5385/tesis.pdf?sequence=1>
- [5]. Mary Catherine O'Connor. “Piloto en Grecia para recolectar basura” (2016). Grecia. Recuperado 7 de Octubre de 2017. <http://www.rfidjournal.com/articles/view?2973/2>
- [6]. Alejandro Arango. Pontificia Universidad Javeriana de Bogotá. “Implementación de un sistema RFID (radio frequency identification) para la identificación de los productos y sus características, en el proceso de manufactura flexible en el centro tecnológico de automatización industrial (ctai) de la facultad de ingeniería industrial de la pontificia universidad javeriana de Bogotá.”(2013). Colombia, Recuperado 7 de Octubre de 2017. <https://repository.javeriana.edu.co/handle/10554/10304>
- [7]. Residuos profesional (2016). Un proyecto piloto identificará las bolsas de basura por radiofrecuencia en la comarca de Txingudi (Gipuzkoa) <http://www.residuosprofesional.com/experiencia-piloto-conetiquetas-inteligentes-en-las-bolsas-de-basura-en-la-comarca-de-txingudi-gipuzkoa/>
- [8]. Antonio Abarca Álvarez. Universidad de Castilla-La Mancha. “Sistema de agentes para control de stock de almacén basado en identificación por radiofrecuencia” (2010). España. Recuperado 7 de Octubre de 2017 <https://ruidera.uclm.es/xmlui/bitstream/handle/10578/1497/TESIS%20DEF.pdf?sequence=1>
- [9].Cristián Maturana M. Departamento de Gestión de información, Escuela de Bibliotecología. “RFID: El código de barras inteligente para Bibliotecas.” (Octubre 2006). Chile. Recuperado 7 de Octubre de 2017. <http://eprints.rclis.org/8438/1/CCF70ADC.pdf>
- [10]V. ACEVEDO DURAN, A. GARCIA SANDOVAL and J. SANDINO ARIZA, "SISTEMA DE REGISTRO Y CONTROL DE SALIDA DE ELEMENTOS MEDIANTE DISPOSITIVOS RFID", Bogotá D.C., 2017.
- [11]C. Collao Vilches, "SISTEMA DE SOPORTE PARA CONTROL DE INVENTARIOS MEDIANTE RFID", Santiago de Chile, 2008.
- [12]E. Ramirez Cerpa and F. Meléndez Pertuz, "Sistemas RFID aplicados a grandes inventarios", Ing Juan, vol. 4, no. 8, 2014.
- [13]Escuela de Ingeniería de Antioquia, "USO DE LA TECNOLOGÍA DE IDENTIFICACIÓN POR RADIOFRECUENCIA RFID EN LA CADENA DE SUMINISTRO PARA PRODUCTOS DE CONSUMO EN MEDELLÍN", Envigado, 2010.
- [14]L. Ni, Y. Liu, Y. Lau and A. Patil, "LANDMARC: Indoor Location Sensing Using Active RFID", Wireless Networks, vol. 10, no. 6, pp. 701-710, 2004.

- [15]M. Casero, "Tecnología de identificación por radiofrecuencia. Lectura de pedidos RFID en un almacén.", Ingeniería en Informática, Universidad de la Rioja, Facultad de ciencias, estudios agroalimentarios e informática., 2013.
- [16]Fundación Iberoamericana de Altos Estudios Profesionales, "CONTROL Y MANEJO DE INVENTARIO Y ALMACÉN", FIAEP, Anzoátegui, Venezuela, 2014.
- [17]X. Han and D. Wang, "Optimization on RFID-Enabled CONWIP Control Strategy for Multi-Echelon Inventory of Supply Chain", World Congress on Intelligent Control and Automation, vol. 1, no. 1, 2016.
- [18]. W. G. B. deMatos, M. Ueda, T. R. de Camargo and A. Morais, "Inventory control with RFID integration," 2015 IEEE Brasil RFID, Sao Paulo, 2015, pp. 1-3. doi: 10.1109/BrasilRFID.2015.7523842
- [19]. Tianle Zhang, Zongwei Luo, Haifeng Wang and Yongliang Zhang, "A service platform for RFID technology adoption analysis in inventory control," 2008 IEEE International Conference on Service Operations and Logistics, and Informatics, Beijing, 2008, pp. 1922-1927.
- [20]. H. Dane, K. Michael and S. F. Wamba, "RFID-Enabled Inventory Control Optimization: A Proof of Concept in a Small-to-Medium Retailer," 2010 43rd Hawaii International Conference on System Sciences, Honolulu, HI, 2010, pp. 1-10.
- [21]. X. Jing and P. Tang, "Research and Design of the Intelligent Inventory Management System Based on RFID," 2013 Sixth International Symposium on Computational Intelligence and Design, Hangzhou, 2013, pp. 8-11
- [22]. M. Zohaib, S. M. Pasha, Z. Hassan and J. Iqbal, "A centralized architecture for inventory management using RFID," 2016 2nd International Conference on Robotics and Artificial Intelligence (ICRAI), Rawalpindi, 2016, pp. 118-123.
- [23]. M. Shahzad and A. X. Liu, "Fast and Accurate Tracking of Population Dynamics in RFID Systems," 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS), Atlanta, GA, 2017, pp. 836-846.
- [24]Q. Xiao, S. Chen, M. Chen, Y. Zhou, Z. Cai and J. Luo, "Adaptive Joint Estimation Protocol for Arbitrary Pair of Tag Sets in a Distributed RFID System", IEEE/ACM Transactions on Networking, pp. 1-16, 2017.
- [25]T. Zhang, Z. Luo, H. Wang and Y. Zhang, "A Service Platform for RFID Technology Adoption Analysis in Inventory Control", IEEE/ACM Transactions on Networking, vol. 1, no. 1, 2017.
- [26]. Raúl Serna Mulero, Santiago Depares Aguilar, Antonio B. Martínez Velasco. Universidad Politécnica de Catalunya. "Implantación de un sistema RFID para obtener trazabilidad en la cadena de suministros". (14 de Mayo de 2012), España. Recuperado 7 de Octubre de 2017. <http://upcommons.upc.edu/bitstream/handle/2099.1/15545/82439.pdf?sequence=1>

[27]. María Estefanía Casero, María Vico Pascual Martínez-Losa. Universidad de la Rioja. “Tecnología de identificación por radiofrecuencia. Lectura de pedidos RFID en un almacén”. (2012-2013), España. Recuperado 7 de Octubre de 2017. https://biblioteca.unirioja.es/tfe_e/TFE000285.pdf

[28]. Observatorio Regional de la Sociedad de la Información. (ORSI). Junta de Castilla y León. “Tecnología de identificación por Radiofrecuencia y sus principales aplicaciones”. (2007), España. Recuperado 8 de Octubre de 2017. https://www.google.com.co/url?sa=t&rct=j&q=&esrc=s&source=web&cd=4&cad=rja&uact=8&ved=0ahUKEwi-5ei7r_fWAhXH1CYKHV9LDM8QFgg1MAM&url=http%3A%2F%2Fwww.jcyl.es%2Fweb%2Fjcyll%2Fbinarios%2F211%2F716%2FRFID.pdf&usg=AOvVaw0wcS4rRXkarMkLI f46eYWB

[29]. Dipole. “Tipos de Sistemas RFID”. (01 de Junio de 2017), España. Recuperado 8 de Octubre de 2017. <http://www.dipolerfid.es/es/blog/Tipos-Sistemas-RFID>

[30] Egomexico. “¿Cómo funciona la tecnología de identificación por radiofrecuencia RFID?”. (s.f), México. Recuperado 9 de Octubre de 2017. http://www.egomexico.com/tecnologia_rfid.htm

[31] Andrea. "Teoría de la administración de inventarios". GestioPolis. Febrero 23, 2002. Consultado el 31 de Octubre de 2017. <https://www.gestiopolis.com/teoria-de-la-administracion-de-inventarios/>.

[32] 4D Doc center. “Conceptos básicos de bases de datos”. Consultado el 18 de Julio de 2018

[33] Techlandia.”¿Qué es una clave foránea en una base de datos relacional”. Diciembre 2001. Consultado el 18 de Julio de 2018. https://techlandia.com/clave-foranea-base-datos-relacional-hechos_353336/

[34] Josefina Vicenteño. “¿QUE ES UNA LLAVE PRIMARIA Y UNA FORÁNEA?”. Octubre 25, 2017. Consultado el 18 de Julio de 2018. <http://usodeoracle.blogspot.com/p/que-es-una-llave-principal-y-una-foranea.html>

[35] Consultado el 18 de Julio de 2018. https://www.campusmvp.es/recursos/image.axd?picture=Northwind_EF.png

[36] Prof. Dr. Manuel Blázquez Ochando Dpto. Biblioteconomía y Documentación Universidad Complutense de Madrid. “Fundamentos y Diseño de Bases de Datos

”. Febrero 20, 2014. Consultado el 18 de Julio de 2018. <http://ccdoc-basesdedatos.blogspot.com/2013/02/modelo-entidad-relacion-er.html>