



FACULTAD DE INGENIERÍA Y CIENCIAS AGROPECUARIAS

DISEÑO DE UN PROTOTIPO DE UN SISTEMA DE CONTROL DE
INVENTARIO PARA BODEGAS UTILIZANDO TECNOLOGÍA RFID Y
ADMINISTRADO POR UNA APLICACIÓN



AUTOR

Danny Leonardo Novillo Ortega

AÑO

2017



FACULTAD DE INGENIERÍA Y CIENCIAS AGROPECUARIAS

DISEÑO DE UN PROTOTIPO DE UN SISTEMA DE CONTROL DE
INVENTARIO PARA BODEGAS UTILIZANDO TECNOLOGÍA RFID Y
ADMINISTRADO POR UNA APLICACIÓN.

Trabajo de Titulación presentado en conformidad con los requisitos
establecidos para optar por el título de Ingeniero en Electrónica y Redes de
Información

Profesor guía

Ms. Carmen Alicia Carabalí Carabalí

Autor

Danny Leonardo Novillo Ortega

Año

2017

DECLARACIÓN DEL PROFESOR GUÍA

“Declaro haber dirigido este trabajo a través de reuniones periódicas con el estudiante, orientando sus conocimientos y competencias para un eficiente desarrollo del tema escogido y dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación”.

Carmen Alicia Carabalí Carabalí

Master en Ingeniería

CI: 1716369713

DECLARACIÓN DEL PROFESOR CORRECTOR

“Declaro (amos) haber revisado este trabajo, dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación”.

Jean-Michel Clairand
Master en Ingeniería
CI: 1714736681

DECLARACIÓN DE AUTORÍA DEL ESTUDIANTE

“Declaro que este trabajo es original, de mi autoría, que se han citado las fuentes correspondientes y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes.”

Danny Leonardo Novillo Ortega

CI: 0604761577

RESUMEN

El presente trabajo consiste en el diseño de un prototipo que utilizando la tecnología RFID permita llevar a cabo la administración de un inventario de una manera automatizada utilizando una aplicación sencilla e intuitiva.

El hardware y software para el desarrollo fue escogido en base a un análisis de requerimientos realizado de forma general ya que este prototipo puede ser aplicado en cualquier escenario de gestión de inventario.

El diseño de la aplicación se realizó pensando en usuarios que no tengan mucho conocimiento de informática por lo que la interfaz de usuario es amigable y de fácil manejo.

La metodología empleada para llevar a cabo el diseño del prototipo fue deductiva y experimental. El método deductivo se aplica en que partiendo de la gran cantidad de usos y aplicaciones que tiene la tecnología RFID, se enfoca en un uso específico que es la administración de un sistema de inventario automatizado; y se utiliza el método experimental por la necesidad de crear una aplicación centralizada que administre de forma fácil y eficiente la información de los artículos en existencia almacenándolos en una base de datos.

ABSTRACT

The present work consists in the design of a prototype that uses the RFID technology to carry out the management of an inventory in an automated way using a simple and intuitive application.

Hardware and software for development was chosen based on a requirement analysis carried out in a general way since this prototype can be applied in any inventory management scenario.

The design of the application was done thinking of users who do not have much knowledge of computers, so the user interface is friendly and easy to use.

The methodology used to carry out the prototype design was deductive and experimental. The deductive method is applied in that starting from the great amount of uses and applications that has RFID technology, it focuses on a specific use that is the administration of an automated inventory system; And the experimental method is used because of the need to create a centralized application that easily and efficiently manages the information of the articles in existence by storing them in a database.

ÍNDICE

INTRODUCCIÓN	1
ALCANCE	2
JUSTIFICACION	2
OBJETIVO GENERAL	3
OBJETIVOS ESPECÍFICOS	3
METODOLOGÍA	3
1. CAPÍTULO I: MARCO REFERENCIAL	4
1.1 Marco Teórico.....	4
1.1.1 Tecnología RFID.....	4
1.1.2 Base de Datos.....	16
1.1.3 Programación por Capas	24
1.1.4 Gestión de Inventarios.....	25
1.2 Análisis de Requerimientos.....	31
1.2.1 Análisis de los Lectores RFID.....	34
1.2.2 Análisis de los Lenguajes de Programación.....	40
1.2.3 Análisis del Sistema Gestor de base de Datos	46
1.3 Propiedades del Sistema.....	50
2. CAPÍTULO II: DISEÑO E IMPLEMENTACIÓN DEL SISTEMA	51
2.1 Desarrollo de la Aplicación.....	52
2.1.1 Capa de Datos	54
2.1.2 Capa de Negocio	60
2.1.3 Interfaz de Usuario.....	69
2.2 Desarrollo de la Base de Datos	77
2.2.1 Modelo Conceptual	78
2.2.2 Modelo Lógico.....	80
2.3 Integración del Sistema	80
3. CAPÍTULO III: PRUEBAS DE FUNCIONAMIENTO	85
3.1 Pruebas de Inicio de Sesión.....	85
3.1.1 Ingreso de un Usuario que no este Registrado	85
3.1.2 Validación de la Contraseña de un Usuario Registrado.....	86
3.1.3 Validación del Nivel de Usuario	87
3.2 Pruebas de Gestión RFID.....	89
3.2.1 Conexión al Lector RFID.....	89
3.2.2 Lectura de TAG's	90

3.3 Pruebas de Gestión de Usuarios.....	91
3.3.1 Creación de Usuarios	91
3.3.2 Actualización de Usuarios.....	92
3.3.3 Eliminación de Usuarios	94
3.3.4 Asignación de Permisos a Usuarios.....	95
3.4 Pruebas de Gestión de Productos.....	96
3.4.1 Ingreso de Productos.....	96
3.4.2 Actualización de Productos	98
3.4.3 Eliminación de Productos	99
3.4.4 Consulta de Productos.....	100
3.4.5 Asignación de TAG's a Productos.....	100
3.5 Pruebas de Gestión de Inventario	102
3.5.1 Escaneo de Productos.....	103
3.5.2 Productos Existentes	104
3.5.3 Productos sin Asignar	104
3.5.4 Productos Fuera de Bodega.....	105
3.6 Pruebas de Cobertura del Lector RFID	106
4. ANÁLISIS COSTO/BENEFICIO.....	108
4.1 Punto de Equilibrio	111
4.2 Período de Devolución	112
4.3 Valor Actual Neto (VAN).....	113
4.4 Tasa Interna de Retorno (TIR).....	114
4.5 Análisis de resultados del Estudio Costo/Beneficio.....	115
5. CONCLUSIONES Y RECOMENDACIONES.....	117
5.1 Conclusiones	117
5.2 Recomendaciones.....	118
REFERENCIAS.....	119
ANEXOS.....	121

INTRODUCCIÓN

El presente trabajo de titulación tiene por objetivo el diseño de un prototipo de un sistema de control de inventario para bodegas utilizando tecnología RFID y administrado por una aplicación, su principal objetivo es la administración eficiente de un inventario al centralizarlo en un sistema de gestión con una interfaz amigable con el usuario.

En la actualidad las pequeñas y medianas empresas (PYMES) del país llevan un inventario de forma manual, es decir que la persona encargada de la bodega tiene que revisar y tomar datos de cada producto existente y registrarlo. Con esta aplicación se trata en gran medida de automatizar este proceso realizando una sola vez el ingreso del producto al sistema y monitoreando en tiempo real si ha salido o se encuentra en stock. Para lograr realizar esto utilizamos la tecnología RFID, el lector para poder identificar cada producto a través de tags asignados, información que será procesada por la aplicación diseñada y comparada con la base de datos de los productos para poder identificar que producto está disponible o fuera de stock.

El diseño del prototipo fue en base a un análisis de requerimientos, el cual permite identificar las necesidades principales y entorno en el cual es funcional. La información es almacenada en una base de datos para lo cual se hace su diseño en función del modelo relacional.

El desarrollo de esta aplicación estuvo enfocado en usuarios que no tengan mucho conocimiento de informática ya que su diseño por módulos e interfaz amigable le ayuda a que de una manera intuitiva navegue por todo el sistema y pueda administrarlo.

ALCANCE

El alcance de este proyecto es diseñar un prototipo de un sistema que al utilizar una tecnología RFID nos permita llevar una administración más eficiente de un inventario al centralizarlo a través de un software.

Para lo cual las actividades a realizar que se contemplan son:

- Hacer un análisis de los requerimientos necesarios tanto a nivel físico como a nivel de software para el desarrollo del sistema.
- Establecer un diseño relacional eficiente para la creación de la base de datos en función de las necesidades requeridas.
- Diseño e implementación de una interfaz amigable para que las personas puedan administrar el sistema de una manera sencilla e intuitiva.
- Integración del software diseñado con el hardware establecido para el sistema.

JUSTIFICACION

La importancia de desarrollar este diseño de un prototipo de un sistema de control de inventario para bodegas utilizando tecnología RFID y administrado por una aplicación radica en que las empresas cada vez exigen mayores y mejores resultados a la hora del manejo de inventarios que se encuentran en bodegas.

Ya que el gran inconveniente de un sistema de inventario en pequeñas y medianas empresas se ve reflejado en que dependen en su mayoría de la supervisión de una persona para mantenerlo actualizado y de la forma en que se realiza este proceso.

OBJETIVO GENERAL

Diseñar un prototipo de un sistema que al utilizar una tecnología RFID nos permita llevar una administración más eficiente de un inventario al centralizarlo en un sistema de gestión con una interfaz amigable con el usuario.

OBJETIVOS ESPECÍFICOS

- Establecer los requerimientos mínimos necesarios para la implementación del sistema.
- Diseñar un modelo entidad-relación en base a las necesidades del sistema para la creación de la base de datos.
- Crear una interfaz amigable, sencilla e intuitiva para facilitar su administración.
- Integrar el software con el hardware de una manera eficiente para mejorar su desempeño.

METODOLOGÍA

En el presente anteproyecto se utilizarán los métodos deductivo y experimental para su desarrollo, ya que se basa en el diseño de un prototipo de un sistema de control de inventario. El método deductivo se utilizará ya que se pretende implementar una aplicación particular (específica) de la tecnología RFID para la realización de este sistema. El método experimental se utilizará para el diseño e implementación de un prototipo que enlace adecuadamente software y hardware.

El método deductivo establece que partiendo de la gran cantidad de usos y aplicaciones que tiene la tecnología RFID en la actualidad, centrarnos en un

uso específico que es la administración de un sistema de inventario automatizado.

En el método experimental se plantea la necesidad de crear una aplicación centralizada que administre de forma fácil y eficiente los artículos en existencia y su ubicación en una bodega almacenándolos en una base de datos.

1. CAPÍTULO I: MARCO REFERENCIAL

1.1 Marco Teórico.

1.1.1 Tecnología RFID

Las comunicaciones de radiofrecuencia donde se intercambia información sobre la identificación de objetos son consideradas de tipo RFID. Radio Frequency Identification (RFID) es el nombre para tecnologías que usan ondas de radiofrecuencia para leer y capturar información almacenada en un tag que se encuentra asociado a un objeto (EPC-RFID INFO, s.f.), normalmente esta información se almacena en un microchip (el conjunto de la antena y chip se lo denomina transponder) el cual permite la transmisión de información a un lector, éste transforma las ondas de radio de la etiqueta RFID en información digital que es interpretada por los ordenadores para hacer uso de ella.(RFID Journal, s.f)

Esta tecnología RFID utiliza diferentes bandas de frecuencia (IMPINJ, s.f.) que se clasifican en:

- Baja frecuencia (125 - 134 kHz y 140 - 148.5 kHz)
- Alta frecuencia (13.56 MHz)

Las cuales se pueden utilizar libremente sin necesidad de obtener licencias para su uso.

- Y la frecuencia ultra alta (UHF: 868 - 928 MHz) no puede ser utilizada de forma general, debido a que depende de cada país el uso y restricciones que le den a ese rango de frecuencias.

1.1.1.1 Historia de RFID

La tecnología RFID fue desarrollada durante la segunda guerra mundial para aplicaciones militares. Una versión indica que esta tecnología nació con la intención de convertirse en una herramienta de espionaje es decir que su propósito era el de un dispositivo de escucha, el cual se desarrolló en el año de 1945 para el gobierno soviético. (Jerry Landt, Barbara Catlin, 2001)

Otras afirmaciones indican que esta tecnología habría existido hace mucho tiempo atrás en el año de 1920, utilizada para la identificación de aviones por parte del ejército británico durante la segunda guerra mundial y se afirma que esta tecnología fue la base para la creación de los actuales sistemas RFID activos (Jerry Landt, Barbara Catlin, 2001).

De la misma manera, el ejército alemán también durante la guerra, identificó que si sus aviones realizaban ciertos tipos de movimientos perturbarían la señal transmitida cambiándola de cierta forma para poder ser identificados como aviones alemanes de los aliados y de esta manera dio origen al primer dispositivo RFID pasivo (Jerry Landt, Barbara Catlin, 2001).

Para el año de 1973 se comenzaron a desarrollar las primeras patentes para el uso, ya no militar, de la tecnología RFID, la cual consistía en una etiqueta RFID activa la cual podía grabar o eliminar información debido a que portaba una memoria interna. En el mismo año también se realizaron patentes para el uso de esta tecnología para abrir puertas sin la necesidad de portar llaves ya que

esta tarjeta emitía una señal de validación hacia un lector el cual verificaba y abría la puerta a través de una cerradura. (Jerry Landt, Barbara Catlin, 2001).

Con la creciente evolución de estos sistemas el gobierno americano también fue partícipe de su uso, en un sistema de control de acceso de puertas en las centrales nucleares permitiendo solo el ingreso a camiones autorizados. También se incursionó con este sistema en la agronomía usándolos con la finalidad de saber si el ganado había sido vacunado o no mediante etiquetas RFID pasivas colocadas debajo de la piel del animal. (Jerry Landt, Barbara Catlin, 2001).

Con el crecimiento de la electrónica y la creación de los semiconductores las etiquetas RFID fueron ganando popularidad y pasaron a ser un producto de consumo masivo al bajar sus costos.

Las investigaciones y continuos avances en esta tecnología han hecho que ganen mercado siendo utilizadas en numerosas aplicaciones tales como por ejemplo: en la detección de contrabando de artículos y otros que actualmente siguen en desarrollo o que incluso ya se encuentran en uso. (Jerry Landt, Barbara Catlin, 2001),

1.1.1.2 Componentes del sistema RFID

Los sistemas RFID requieren de un mínimo de componentes para poder funcionar correctamente (nControl, s.f.) como los que se muestran en la Figura 1.

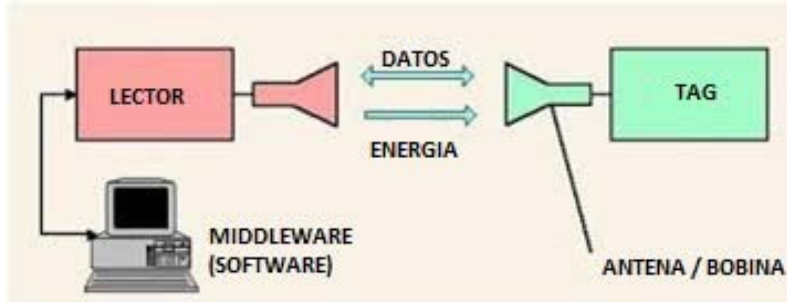


Figura 1. Componentes básicos de un sistema RFID.
Tomado de nControl, s.f

A continuación detallaremos la función de cada uno:

Transponder (Tag o Etiqueta)

Los transponders o etiquetas (nControl, s.f.), Figura 2 son dispositivos electrónicos que internamente están compuestos por:

- **Memoria:** En la cual se pueden almacenar datos e instrucciones básicas propias de la tarjeta que van a ser interpretados por el lector.
- **Antena:** Es el dispositivo mediante el cual la etiqueta o tag detecta el campo magnético creado por el lector y transmite la información almacenada en la memoria para después ser interpretada por el middleware.
- **Componentes electrónicos:** Estos componentes son utilizados para el procesamiento de la señal a través de la antena y de los datos a transmitirse hacia el lector.

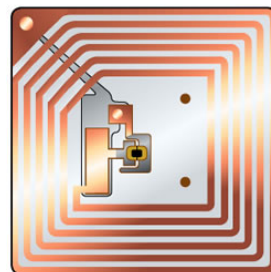


Figura 2. Etiqueta RFID.
Adaptado de Murimar, 2011

Los transponders tienen la función escuchar una señal de radio específica en base a sus elementos internos y enviar una respuesta, dependiendo de la complejidad del sistema la respuesta puede ser con un solo carácter o con una cadena de caracteres. (Tecnología RFID, s.f.).

Las etiquetas en función de sus requerimientos de energía se clasifican en:

- **Etiquetas Activas:** Este tipo de etiquetas necesitan de una fuente de energía propia para poder funcionar, la que generalmente suele ser una batería. Al tener una batería, estos sistemas pueden tener un amplio alcance por lo que se los utiliza en aplicaciones donde se manejen grandes distancias para el envío /recepción de información (Tecnología RFID, s.f.). La figura 3 muestra un ejemplo.



Figura 3. Etiquetas RFID activas.
Tomado de Tecnología RFID, s.f.

- **Etiquetas Pasivas:** Estas etiquetas en cambio al no tener una batería como fuente de alimentación necesitan esperar a una señal emitida por un lector. La etiqueta contiene un círculo resonante que le permite absorber la energía emitida por la antena del lector para poder así entrar en funcionamiento (Figura 4). Debido a que la energía proviene del lector las etiquetas no deben estar demasiado distantes, deben permanecer dentro del denominado “Near Field” o campo cercano donde puede recibir energía del lector para poder trabajar. (Tecnología RFID, s.f.)

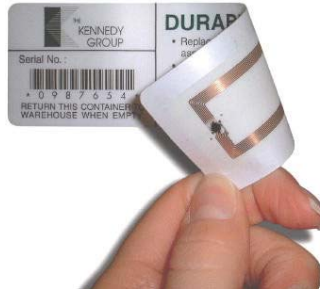


Figura 4. Etiqueta RFID pasiva.
Tomado de Murimar, 2011

Las etiquetas en función de la información que pueden almacenar se clasifican en:

- **Etiquetas de solo lectura:** En estas etiquetas la información debe ser previamente almacenada por los lectores cuando son fabricadas, normalmente lo que se almacena es un número de serie único para su identificación. (Tecnología RFID, s.f.).
- **Etiquetas de Lectura /Escritura:** En este tipo de etiquetas se puede grabar o borrar información dependiendo de lo que se necesite en base a la aplicación en que se utilice este sistema, con estas etiquetas es muy sencillo actualizar o modificar la información según la capacidad del equipo. (Tecnología RFID, s.f.)

Lector

Un lector RFID es un dispositivo que permite transmitir señales de radio hacia los transponders o tags, así como también recibir información de ellos para procesarla (Figura 5).

La forma básica en que funciona un lector es a través de la producción de electricidad que viaja a través del dispositivo y cables hasta llegar a la antena que irradia la misma señal en el espacio a una frecuencia determinada para que otros dispositivos lo escuchen. El trabajo del lector no solo es generar y transmitir señales a través de la antena sino que también escucha la respuesta

de dispositivos compatibles que en este caso serían las etiquetas RFID. (nControl, s.f.)

Cada lector puede tener una o más antenas dependiendo del modelo y de la potencia en la que va a irradiar la señal. Respecto a su interfaz de comunicación, la mayoría de lectores posee interfaces de comunicación RS-232 o Ethernet para poderse comunicar con un computador o servidor y de esa manera poder administrar el dispositivo. (nControl, s.f.)

Hay una gran cantidad de lectores disponibles en el mercado según sus capacidades por ejemplo: lectores simples, multifrecuencias, multiregionales, multiprotocolo, etc.



Figura 5. Lector RFID.
Tomado de GAO RFID, s.f.

Middleware

El middleware se encuentra entre el hardware RFID y las aplicaciones de software que utilizan los clientes. La principal función del middleware es administrar todo el sistema RFID a nivel físico, recibir toda la información de los

tags y filtrar para solo transmitir información útil a los sistemas de gestión o software de manejo (Figura 6). (nControl, s.f.)

En gran medida, el middleware puede ser también un software diseñado para transmitir la información recogida por lectores a la aplicación correspondiente para luego procesarla según las necesidades del software involucrado. (nControl, s.f.)



Figura 6. Diagrama de bloques de la interacción del middleware con el sistema RFID y el sistema de gestión.

Tomado de Muñoz Javier C., Camarasa Jordi E., 2011

1.1.1.3 Funcionamiento

El funcionamiento del sistema RFID está determinado por el lector que su función principal es emitir una señal de radio de baja potencia que va a ser transmitido por una antena que también forma parte del lector. Esta señal (campo electromagnético) en el caso de los tags pasivos es el encargado de proveerlos de energía y de transmitir información de identificación del lector hacia la etiqueta y viceversa, en los tags activos como poseen su propia fuente de energía servirá únicamente para la transmisión de información (Figura 7). (nControl, s.f.)

En la descripción del funcionamiento se realizara énfasis en el uso de tags pasivos ya que son los que se utilizarán en el desarrollo del prototipo, el tag también posee una antena la cual recibe la energía a través del campo magnético generado por el lector para poder energizar los elementos internos

que posee y principalmente el chip de memoria que posee la información que posteriormente será transmitida hacia el lector (Figura 7). (nControl, s.f.).

En el tag se establecen dos fases la primera es en la que recibe energía es decir está en fase de almacenamiento de energía y la segunda fase es en la que cada etiqueta envía su código utilizando la energía almacenada en el proceso anterior. La señal electromagnética que genera el tag es recibida por el lector el cual la procesa y decodifica para luego verificar si los datos transmitidos son correctos. Una vez que el lector verifica y valida que los datos estén sin errores los datos son reestructurados para su transmisión como información en el formato requerido por el sistema el cual está conectado al lector. (nControl, s.f.)

El rango de cobertura depende directamente de la potencia del lector así como también del tamaño y número de antenas que posea, por parte de los tags influye si son activos o pasivos, siendo los activos los que soporten mayores distancias.

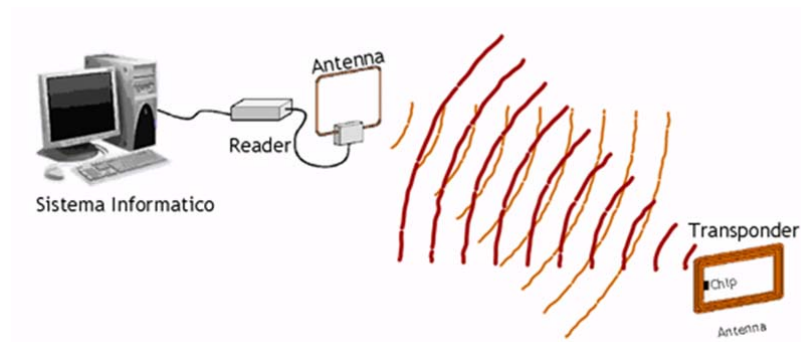


Figura 7. Esquema completo de funcionamiento de un sistema RFID. Tomado de ERPlan, s.f.

1.1.1.4 Aplicaciones

La tecnología RFID conforme el paso del tiempo ha ido evolucionando, entrando cada vez más en el mercado tecnológico con un sin número de aplicaciones que para dar un ejemplo citamos algunas.

- Controles de Acceso



Figura 8. La tarjeta RFID se identifica con el lector para garantizar el acceso. Tomado de RAHID, s.f.

Las aplicaciones en este campo han tenido una gran demanda debido a la necesidad de tener un control de seguridad de puertas según los diferentes niveles de acceso para los usuarios autorizados a través de una simple tarjeta (Figura 8). Estas nuevas formas de acceder a numerosos lugares son cada vez más comunes que incluso en edificios residenciales ya poseen este sistema. (AB&R, s.f.).

- Identificación de equipajes en aeropuertos



Figura 9. Etiquetas RFID ubicadas en el equipaje para poder ser identificadas Tomado de DELTA, s.f.

Ya hace varios años en los aeropuertos se está utilizando un sistema RFID para facilitar el control de equipaje en la línea de traslado reduciendo de esta manera costos y tiempo a las compañías aéreas. Ya que el equipaje es trasladado por toda la cadena de distribución mediante varios sensores en base a la información que guarda en su etiqueta (Figura 9) que incluso le permite saber a qué avión va dirigido dicho equipaje. (AB&R, s.f.).

- Industria Automotriz



Figura 10. Apertura de puertas de un vehículo en base a un transponder RFID de seguridad sin la necesidad de llaves.

Tomado de RAHID, s.f.

Se introdujeron los sistemas RFID en el segmento automotriz para incrementar la seguridad de los vehículos en la que un transponder se encontraba en la llave del auto el cual tiene un código único que al ser validado por el lector al girar la llave permite el arranque del motor (*Figura 10*). (AB&R, s.f.).

- Identificación de animales

En la actualidad es muy común que propietarios de mascotas implanten etiquetas RFID pasivas en ellos. En dicha etiqueta se guarda un código único así como también información relacionada al propietario como: dirección, número telefónico, nombre del propietario; para que en caso de que se pierda una mascota se pueda leer los datos del tag (*Figura 11*). (AB&R, s.f.)

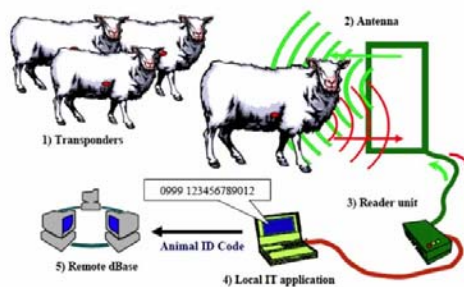


Figura 11. Implantación de chips (tags RFID) para la identificación de animales. Tomado de *Understanding RFID System-Basics*, s.f.

1.1.1.5 Ventajas y desventajas de RFID.

En el mercado la mayor competencia que tiene esta tecnología RFID son los códigos de barra por lo que se realizará una comparación entre estas dos tecnologías resaltando los beneficios que presenta RFID.

- No se necesita contacto visual para poder leer los Tags.

- Un sistema de código de barra no es capaz de identificar solo un objeto ya que si son productos iguales llevan la misma información evitando de esta manera poderlos diferenciar, lo que RFID lo puede hacer de una manera sencilla.

- RFID permite la lectura de múltiples etiquetas o Tags simultáneamente, los sistemas de códigos de barras no poseen esta capacidad por lo que deben ser leídos secuencialmente.

- Los Tags pueden almacenar más información ya que poseen una memoria interna.

- En los códigos de barras no se permite más que una impresión de código para su identificación mientras que con etiquetas RFID se puede sobrescribir las veces que sean necesarias.

- En un sistema RFID se evita en mayor medida las suplantaciones y falsificaciones ya que en los sistemas basados en códigos de barras se puede fotocopiar y reproducir un código. Una etiqueta RFID sobre un artículo u objeto garantiza su procedencia y de esta manera su autenticidad.

- Reduce costos en su administración.

1.1.2 Base de Datos

Generalmente se lo llama sistema de base de datos a un sistema informático (computarizado) que tiene como objetivo almacenar registros y permite que los usuarios puedan hacer consultas y modificaciones a dichos registros. (DATE, C.J., 2001, p. 33).

1.1.2.1 Arquitectura de los sistemas de base de datos.

Un sistema de base de datos tiene tres niveles de arquitectura según la ANSI/SPARC que son:

- Nivel Interno

Es el nivel que se encuentra más cerca de la parte física, describe como lo datos se almacenan físicamente en los discos duros (DATE, C.J., 2001, p. 34).

Define tipos de registros y especifica los índices que existen, es decir que establece como están almacenados los campos, en que secuencia, entre otras funciones. (DATE, C.J., 2001, p.35).

- Nivel Externo

Es el nivel que interactúa directamente con el usuario final, es decir que puede ser interpretado como orientado hacia el usuario y como este percibe la información para su trabajo. (DATE, C.J., 2001, p. 37).

Existen dos tipos de usuarios que pueden tener acceso a una base de datos, pero tienen una distinta percepción de la misma, estos usuarios son:

- Usuario programador es aquel que tiene mayor conocimiento en informática y tiene la capacidad de trabajar con lenguajes de programación

(PL/I, C++, java, etc.) que le permiten interactuar directamente con las base de datos a nivel de líneas de código y consultas. (DATE, C.J., 2001, p. 38).

- Usuario final es aquel que no posee mucho conocimiento de informática y necesita interactuar con una base de datos de una manera más sencilla, es decir este usuario trabaja con interfaces graficas previamente diseñadas según requerimientos específicos. (DATE, C.J., 2001, p. 37).

Independientemente del usuario que esté utilizando la base de datos, su medio de acceso incluirá un sublenguaje de datos encargado de los objetos y las operaciones (SLD) como parte de su lenguaje anfitrión. En general cualquier sublenguaje tiene por lo menos dos lenguajes subordinados, un DDL (lenguaje de definición de datos) que permite la definición o declaración de objetos y un DML (lenguaje de manipulación de datos) que permite la manipulación o procesamiento de dichos objetos. El sublenguaje más común soportado por la mayoría de lenguajes actuales es SQL el cual permite interactuar con la base de datos como un lenguaje de consultas. (DATE, C.J., 2001, p. 38).

En concreto la parte del DDL maneja las construcciones declarativas necesarias para expresar los objetos en la base de datos, extensiones especiales, para manejar objetos nuevos, etc. mientras que la parte DML consiste de instrucciones ejecutables que transfieren información desde y hasta la base de datos. (DATE, C.J., 2001, p. 38).

- **Nivel Conceptual**

Llamado también nivel lógico de la comunidad, consiste en una representación abstracta de una base de datos, a diferencia del nivel externo que representa una parte limitada de la base de datos. (DATE, C.J., 2001, p. 39).

Este esquema incluye características adicionales como por ejemplo la integridad de los datos, su seguridad. Se plantea que el objetivo final de este esquema es describir toda la base de datos, no solo los datos sino también como son éstos utilizados, la forma de cómo fluyen de un punto a otro, su función específica. (DATE, C.J., 2001, p. 39).

Estos tres niveles están representados en la figura 12.

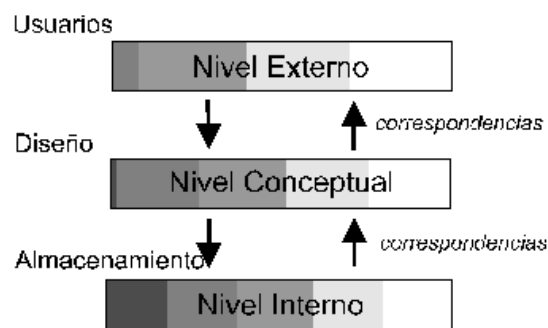


Figura 12. Los tres niveles de arquitectura.
Tomado de DATE, C.J., 2001, p. 34

1.1.2.2 El sistema de Administración de base de Datos.

Llamado también solo DBMS es el software que maneja todo acceso a la base de datos (DATE, C.J., 2001, p. 41), que tiene la siguiente estructura:

- Petición de acceso, utilizando cualquier sublenguaje de datos (por ejemplo SQL).
- Recibe la petición por el usuario y la analiza.
- Inspecciona el esquema externo del objeto, lo transforma de externa/conceptual luego la transforma en conceptual/interna según la arquitectura previamente analizada para finalmente llegar a la base de datos.
- Finalmente ejecuta las operaciones necesarias para que la instrucción del usuario se procese y ejecute.

Analizando las funciones del DBMS a más detalle tenemos:

- ***Definición de datos***

Debe ser capaz de aceptar las definiciones de los datos de toda la arquitectura y sus transformaciones en la forma fuente para convertirlos en la forma de objeto correspondiente. (DATE, C.J., 2001),

- ***Manipulación de datos***

Maneja las peticiones básicas de recuperar (mostrar), actualizar o eliminar datos existentes en la base de datos o agregar nuevos datos a esta. Es decir que para que pueda realizar todas estas acciones posee un procesador DML o compilador DML. (DATE, C.J., 2001, pp. 42-43).

- ***Seguridad e integridad de los datos***

Los usuarios al tener acceso a la base de datos el administrador se ve en la tarea de establecer reglas o restricciones de seguridad para verificar las peticiones y transacciones hacia la base de datos. (DATE, C.J., 2001)

La integridad de una base garantiza que los datos ingresados sean los correctos, es decir que no ocurra una inconsistencia al tratar de representar dos entradas diferentes asignadas a un mismo hecho, este problema puede ocurrir solo si existe redundancia en los datos almacenados. La manera más eficiente para evitar este problema es tener un control centralizado para que el administrador pueda definir las reglas del negocio o restricciones de integridad. (DATE, C.J., 2001, pp. 42-43).

- ***Diccionario de Datos***

Es interpretado como una recopilación o base de datos propia del sistema que generalmente contiene metadatos o descriptores, es decir que se utiliza este diccionario para definir otros objetos del sistema. Todos los diversos esquemas y transformaciones, las restricciones de seguridad e integridad serán también almacenadas en este diccionario. (DATE, C.J., 2001, p. 43).

1.1.2.3 Arquitectura Cliente-Servidor

Este modelo es muy utilizado en los sistemas de información que manejan procesos divididos que cooperan entre sí para el intercambio de información, intercambio de servicios o recursos como se muestra en la figura 13. (DATE, C.J., 2001, p. 48).

Se le denomina cliente a quien realiza las peticiones mientras que servidor es el que responde a las solicitudes. (DATE, C.J., 2001, p. 48). Sus principales características son:

- El servidor tiene una interfaz única y definida para sus clientes.
- El cliente solo conoce su interfaz externa.
- La ubicación física del servidor no es relevante para los clientes.
- Cualquier cambio en el servidor los clientes no lo perciben.

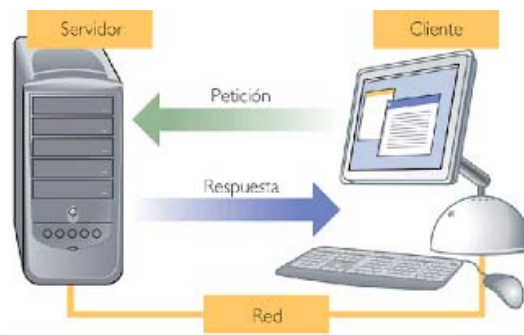


Figura 13. Modelo cliente-servidor.
Tomado de Rodríguez Dairo, 2011

El cliente normalmente tiene contacto para manipular y consultar datos, por lo que manejan interfaces gráficas (GUI) para su acceso. (DATE, C.J., 2001, p. 49).

El servidor en cambio procesa las peticiones de los clientes, maneja las reglas del negocio y los recursos de datos. (DATE, C.J., 2001, p. 49).

1.1.2.4 Modelo Relacional

Para que un sistema sea llamado relacional o que utilice un modelo relacional, solo requiere que el usuario tenga una percepción de la base de datos en forma de tablas. La lógica en un sistema relacional son las tablas y el nivel físico no tiene gran importancia ya que puede ser cualquiera que desee (utilizar archivos secuenciales, indexados, dispersión, etc.). (DATE, C.J., 2001, p. 109).

Es decir la forma en que los datos están almacenados es representado por tablas y omite detalles como su ubicación, la secuencia, las estructuras de acceso almacenadas, etc. que no son de relevancia para el usuario final. (DATE, C.J., 2001, p. 109).

En un modelo relacional los niveles conceptual y externo serán relacionados a diferencia del nivel físico o interno que se maneja de forma separada.

Las principales operaciones que se pueden realizar en este modelo relacional son:

- **Restringir** (también conocido como seleccionar) operación que permite extraer filas específicas de una tabla.

- **Proyectar** esta operación nos permite extraer columnas específicas de una tabla.

- **Juntar** esta operación permite unir dos tablas en base a valores de columnas comunes.

1.1.2.5 Introducción a SQL

SQL es un lenguaje normalizado ANSI que garantiza que cualquier motor de base de datos pueda interpretarlo. (DATE, C.J., 2001, pp. 83-85).

Los comandos que maneja este lenguaje son de dos tipos los DDL que permiten crear y definir nuevas bases de datos, campos e índices, como se muestra en la tabla 1 y los DML que permiten generar consultas para ordenar, filtrar y extraer datos, como se muestra en la tabla 2. (DATE, C.J., 2001).

Tabla 1.
Comandos DDL

Comando	Descripción
CREATE	Para crear nuevas tablas, campos e índices
DROP	Eliminar tablas e índices
ALTER	Modifica las tablas agregando campos o cambiando la definición de los campos.

Tabla 2.
Comandos DML

Comando	Descripción
SELECT	Consulta registros de la base de datos que satisfagan un criterio determinado
INSERT	Carga lotes de datos en la base de datos en una única operación.
UPDATE	Modifica los valores de los campos y registros especificados
DELETE	Elimina registros de una tabla de una base de datos

Las clausulas son los parámetros o las condiciones de modificación, como se muestra en la tabla 3.

Tabla 3.
Principales cláusulas

Cláusula	Descripción
FROM	Especifica la tabla de la cual se van a seleccionar los registros
WHERE	Especifica las condiciones que deben reunir los registros que se van a seleccionar
GROUP BY	Separa los registros seleccionados en grupos específicos
HAVING	Expresa la condición que debe satisfacer cada grupo
ORDER BY	Ordena los registros seleccionados de acuerdo con un orden específico

Tabla 4.
Operadores Lógicos

Operador	Uso
AND	Evalúa dos condiciones y devuelve un valor de verdad sólo si ambas son ciertas.
OR	Evalúa dos condiciones y devuelve un valor de verdad si alguna de las dos es cierta.
NOT	Negación lógica

Tabla 5.
Operadores de Comparación

Operador	Uso
<	Menor que
>	Mayor que
<>	Distinto de
<=	Menor o Igual que
>=	Mayor o Igual que
=	Igual que
BETWEEN	Especificar un intervalo de valores.
LIKE	Comparación de un modelo
In	Especifica registros de una base de datos

Las funciones de agregado son usadas para devolver un único valor de un grupo de registros, como se muestra en la tabla 1.6. (DATE, C.J., 2001, p. 89),

Tabla 6.
Funciones de agregado

Función	Descripción
AVG	Calcula el promedio de los valores de un campo determinado
COUNT	Devuelve el número de registros de la selección
SUM	Devuelve la suma de todos los valores de un campo determinado
MAX	Devuelve el valor más alto de un campo especificado
MIN	Devuelve el valor más bajo de un campo especificado

1.1.3 Programación por Capas

La programación por capas es muy útil para separar la aplicación en componentes que se encuentren en diferentes niveles funcionales, para de esta manera hacer más sencilla la corrección y detección de errores durante el desarrollo del aplicativo. Adicionalmente un modelo en capas ayuda a dar escalabilidad a una aplicación, pues las capas de bajo nivel suelen mantenerse inalterables y el cambio se evidencia en la capa que interactúa directamente con el usuario. (Microsoft, 2016).

- Capa de datos

Aquí se encuentran todos los datos cuya función son de recuperar, almacenar o eliminar la información con la cual va a trabajar la aplicación. Esta capa tiene conexión directa a la base de datos, la misma que se realiza con la importación de un cliente SQL utilizado para la apertura y cierre de dicha conexión. La base de datos contiene todas las tablas que relacionan la funcionalidad de la aplicación y almacenan toda la información de los productos. (Microsoft, 2016).

- Capa de Negocio

Aquí se realizan todas las validaciones de los datos de entrada y salida, es aquí donde se desarrolla toda la lógica de negocio, donde se realizan las funciones que manejan los datos provenientes de la capa conexión y se aplican

las reglas necesarias para la aplicación. Inclusive en esta capa se realizan las validaciones de formato. (Microsoft, 2016).

- **Interfaz de Usuario**

Aquí se desarrolla todo el entorno gráfico de la aplicación, es la capa que tiene relación directa con el usuario final, pues es a través de ella que se va a realizar toda la interacción entre la aplicación del negocio y el usuario final. La creación de esta capa está desarrollada en diferentes formularios que de manera intuitiva van guiando al usuario en el manejo de la aplicación. Esta capa es la que utiliza toda la funcionalidad del equipo escogido para el proyecto, dando al cliente un ambiente sencillo, amigable, de fácil manejo y a la vez robusto y confiable con la administración de los activos del negocio. (Microsoft, 2016).

1.1.4 Gestión de Inventarios.

La gestión de inventario provee información exacta acerca de la cantidad de existencias almacenadas, cual es el nivel mínimo que se debe mantener disponible, en el caso de que este por agotarse el producto indica cuando debe reponerse las existencias y el tamaño del pedido, toda esta información es realizada a través de políticas aplicadas al monitoreo de las existencias (Suárez, D.J., 2011).

La administración adecuada de un inventario optimiza los procesos operacionales y gerenciales de las empresas, ya que es de mucha importancia tener un control de flujo de información de inventario (actualizaciones o ajustes) para tener unos datos contables y financieros exactos, dentro de la dinámica de despacho y cadenas de suministros (Suárez, D.J., 2011).

- ***Determinación de existencias.***

Saber qué cantidad de productos se encuentran almacenados es muy crítico y se puede hacer de distintas formas tales como:

- Revisión física de las existencias, en la cual la persona responsable se encarga de revisar y contabilizar cada uno de los productos almacenados para generar un reporte de los mismos.
- Auditoria, en este proceso una persona designada para este fin se encarga a través de la información de la empresa de revisar y contabilizar toda la información de productos almacenados y si es necesario realizan una revisión física de los mismos.
- Revisión de los procesos de ingresos y egresos, todas las empresas llevan registros acerca de los nuevos productos que ingresan así como también las notas de entrega de los productos que salen de la empresa a los clientes, de esta manera se pueden tener registros de los productos.
- Conteos periódicos, dependiendo de la empresa y el giro de la misma tienen establecidas políticas de registros y conteo anuales de los productos que tienen almacenados.

- ***Análisis de inventarios***

Se realiza un análisis estadístico de las existencias para verificar que los procesos de adquisición y los tamaños de los pedidos estén valorizados correctamente o para modificarlos en caso de ser necesario (Suárez, D.J., 2011).

- ***Control de Producción.***

En este tipo de control se evalúa que tan óptimo es el proceso de transformación de la materia prima en productos terminados y listos para ser comercializados (Suárez, D.J., 2011).

1.1.4.1 Técnicas y Métodos de Gestión de Inventarios

- **Clasificación ABC.**

Cada empresa en su entorno de negocio utiliza diferentes tipos de productos que son manejados de una manera independiente en base a su importancia en los procesos de la empresa y sus posibilidades de adquisición. El análisis ABC es una manera de clasificar los productos haciendo referencia al valor de los inventarios y estableciendo porcentajes para los tres más significativos (Suárez, D.J., 2011).

Los productos indispensables para el funcionamiento o rentabilidad de la empresa son clasificados como productos A, siguiendo el mismo esquema de importancia para la empresa se establecen las demás zonas. La figura 14. indica una representación gráfica de este modelo.

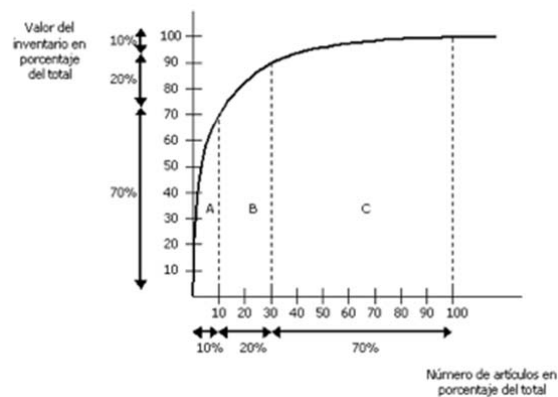


Figura 14. Olimpiadas Economía.
Tomado de Vázquez José, 2015

- **Modelo de Cantidad Económica del Pedido.**

Este modelo es ampliamente utilizado y parte de la suposición de varios factores tales como:

- La demanda es conocida y es constante.
- Los costos relacionados con el modelo son constantes.

- La cantidad de pedido por orden es la misma.
- El pedido se recibe en el momento que se ordena.
- El inventario se restablece en el momento en que se agota.
- El proveedor entrega las cantidades solicitadas en un solo lote.
- Se considera un horizonte infinito y continuo en el tiempo.

Estableciendo esas suposiciones se genera una gráfica como la que se muestra en la figura 15.

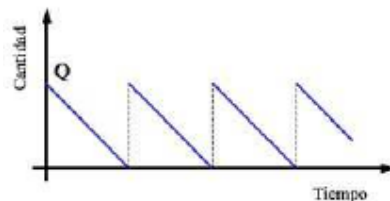


Figura 15. Gestión técnica de Inventarios.

Tomado de Suárez David, D.J., 2011.

Para establecer los parámetros de la gráfica se necesitan conocer los siguientes datos:

- Demanda, generalmente es anual o se puede calcular en base a los presupuestos de la empresa.
- Costo de los pedidos, este costo se genera cuando la empresa adquiere nuevos productos y cuenta desde los gastos administrativos que se derivan así como los gastos de transporte y recepción, se contabiliza el tiempo desde que se inicia el pedido.
- Costo de mantenimiento (conservación), este costo indica el gasto que genera tener un producto en almacenamiento, durante el tiempo en que se estime su demanda.

Relacionando todos los parámetros anteriores de una manera objetiva el modelo da unos resultados válidos (Suárez, D.J., 2011).

1.1.4.2 Modelos de Inventarios

Los modelos de inventarios tienen como objetivo satisfacer una demanda a tiempo, las empresas mantienen un inventario en función de sus ventas y con análisis a futuro (corto plazo) de su demanda (Suárez, D.J., 2011).

- Tipos de Stock

Dentro del stock de productos se subdividen en tres subtipos principales según la función que desempeñan que son:

- **Stocks en ciclo:** Es cuando se solicita una orden de compra superior a las necesidades del momento.
- **Stocks estacionales:** Este tipo de stock se genera cuando hay una demanda variable del producto, es decir aumenta o disminuye en determinadas épocas, lo que da lugar a una producción mayor que la demanda en determinados periodos de tiempo.
- **Stocks de seguridad:** Es cuando una empresa se protege frente a la demanda teniendo artículos adicionales disponibles.

- Características de la demanda.

Las principales características de la demanda son:

- **Determinista o probabilística,** ya que hay casos en los que la demanda futura puede ser conocida o totalmente aleatoria
- **Dependiente o independiente:** La demanda es dependiente cuando es parte de un producto final, los productos finales se consideran independientes.
- **Homogénea o heterogénea:** Se considera que la demanda se homogénea si su valor no cambia en el tiempo.
- **Diferida o perdida:** Es considerada como perdida si no satisface la demanda (ruptura de stock) y diferida cuando se puede solucionar el problema con otro proveedor.

- **Periodos de entrega y reaprovisionamiento.**

El periodo de entrega es el tiempo que transcurre entre que se realiza el pedido y la llegada al usuario final para su uso o consumo. En la figura 16. se observa claramente el punto A es el tiempo de detección del pedido, el C es cuando el producto ya está disponible para el usuario o consumidor, el punto B es cuando el tiempo de reaprovisionamiento fue mayor al de la demanda, que quiere decir que la demanda no fue cumplida y genera una ruptura de stock; en el caso del punto D el tiempo de la demanda es superior al tiempo de reaprovisionamiento por lo que la demanda está satisfecha pero genera costos de posesión de stocks (Suárez, D.J., 2011).

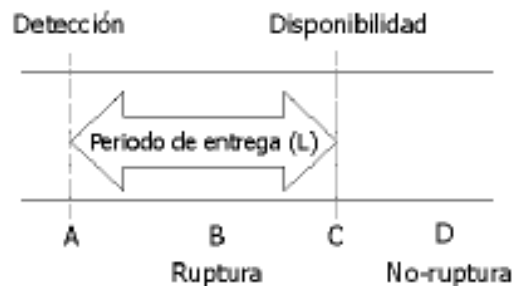


Figura 16. Gestión técnica de Inventarios.

Tomado de Suárez David, D.J., 2011.

- **Políticas de gestión de inventario.**

Las políticas de gestión de inventario tienen como objetivo dar respuesta a dos preguntas principales, la primera cuando se debe realizar un reaprovisionamiento de material, la segunda cuánto de este material debo pedir (Suárez, D.J., 2011).

Para dar respuesta a la primera interrogante una de las opciones es establecer valores referenciales para cuando el stock este por debajo de este valor se realice una orden de compra, otra alternativa es fijar un punto concreto de reabastecimiento en base a un período de revisión realizado previamente. En el caso de la segunda interrogante una opción es establecer una cantidad fija

para cada pedido, o la diferencia entre la cantidad necesaria para cubrir la demanda con lo que se encuentra en stock (Suárez, D.J., 2011).

Para describir una política de gestión es suficiente con responder este par de preguntas planteadas de la manera más adecuada en función de las necesidades de la empresa y la demanda.

1.2 Análisis de Requerimientos.

En la actualidad la cadena de suministros e inventarios tiene un rol muy importante dentro del funcionamiento de una empresa, ya que su uso y administración adecuada mejora la productividad y reduce los costos de operación, por lo que es necesario administrar los recursos de manera efectiva y eficiente.

Se han desarrollado sistemas de gestión para manejar los recursos y supervisar las operaciones, sin embargo aún en la actualidad es difícil actualizar las operaciones diarias de nivel de inventario, ubicaciones y unidades de mantenimiento de stock en tiempo real utilizando los sistemas de gestión tradicionales (códigos de barra o manuales). Por lo que se ha adoptado la tecnología RFID para facilitar la recolección y el intercambio de datos en la gestión de inventarios.

La bodega modelo utilizada para el desarrollo del prototipo es de 38m² de área y posee un solo acceso una puerta principal de 2m x 2m (4m² de área), esta bodega está destinada para el almacenamiento de productos electrónicos de un costo moderado tales como PC's, impresoras, monitores, tablets, celulares y más artículos de esa gama, organizados en diferentes perchas que cubren casi toda el área de la bodega.

Analizando las necesidades e inconvenientes generales que posee la bodega la cual se utilizará como modelo para el desarrollo del prototipo tenemos que su manejo de inventario siempre fue de forma manual, mediante órdenes de ingreso de manera escrita o verbal en muchos casos y de manera similar sus egresos de artículos.

Para revisar en su stock los productos faltantes la persona encargada de la bodega tiene que contar los artículos de uno en uno para de esta manera comprobar y hacer el pedido de algún faltante.

Para mitigar las necesidades de esta bodega se describen los principales requerimientos de control de inventario que son:

- Eficiente control y manejo de entradas - salidas de productos.
- Poder obtener productos y saldos actualizados a la fecha.
- Información de los productos, de la bodega en general y líneas existentes.

- Administración total de los productos, usuarios y líneas de bodega.

En función de estas necesidades se realizó un análisis de requerimientos que ayude a un manejo y administración eficiente del sistema de inventario.

Para el manejo de entradas y salidas de productos de una manera automatizada se hace uso de la tecnología RFID ya que ésta nos permite tener información en tiempo real del estado de los productos en la bodega asociado al desarrollo de una aplicación que con esta información almacene y administre de una manera ágil todos los productos en existencia, así como los que están fuera de bodega.

Para contabilizar el número de productos en existencia así como el de productos fuera de bodega el lector debe ubicarse en una posición donde pueda registrar la entrada y salida de productos, es decir que su mejor punto de ubicación sería donde tenga la completa cobertura en la puerta de acceso a la bodega. Como se muestra en la Figura 17.

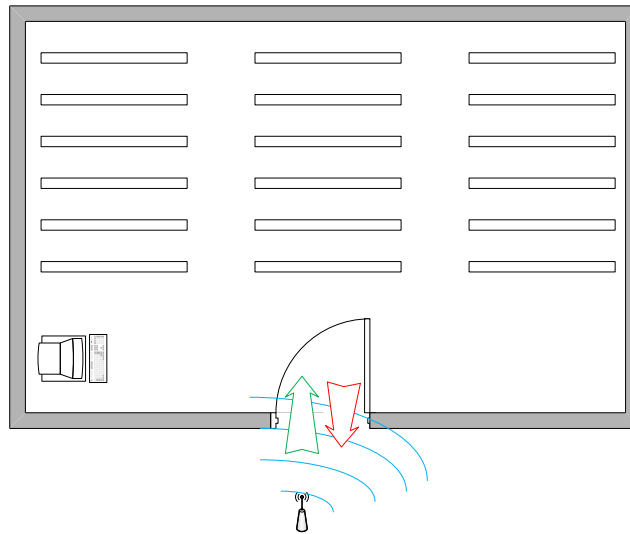


Figura 17. Esquema de ubicación del prototipo en la bodega

Para el análisis de los productos con mayor salida y las advertencias cuando estén por agotarse se las tomará en cuenta en el desarrollo del software de gestión de inventario para realizar ese análisis.

1.2.1 Análisis de los Lectores RFID

Para escoger el tipo de lector apropiado para el desarrollo del prototipo se evaluaron los diferentes tipos de lectores RFID (Dipole, s.f.) que existen que son:

- Lectores fijos.
- Lectores portátiles o manuales.
- Lectores de sobremesa USB.

- Lectores de montacargas.

Estableciendo una comparativa entre los cuatro tipos de lectores y sus respectivas funcionalidades se establece que el más apropiado para el desarrollo del prototipo es el lector fijo, para que los tags se desplacen hasta el área de cobertura del lector y sean leídos.

Los principales lectores de éste tipo en el mercado son lector Kimaldi, lector RFID UHF SL130, lector UHF-RW-MP-232-V1-7dbi; se han escogido estos lectores ya que son los que se encuentran en venta en Ecuador y cumplen con los requerimientos de la bodega analizada anteriormente.

Para poder dimensionar la potencia (capacidad) y el tipo de tags del lector se establece su área de cobertura la cual como se explica anteriormente debe cubrir el ingreso principal a la bodega como se muestra en la figura 1.17. Por lo que la potencia del equipo debe ser baja ya que va a cubrir solo una puerta de acceso, los tags a utilizarse en este caso serían pasivos debido a que la distancia hacia el lector RFID al atravesar el acceso (entrada/salida) de la bodega es corto.

En base a estas premisas se realiza un análisis comparativo entre los lectores previamente indicados para saber cuál es el más idóneo para la elaboración de este prototipo.

- **Lector Kimaldi**

El lector Kimaldi utiliza la banda de frecuencia UHF para su transmisión de largo alcance, es compatible con multi-protocolo, permite lectura y escritura de tags, lectura rápida, lectura multi-tag, el método de conexión para su administración es serial a través de RS 232.

Sus principales aplicaciones son: en logística, gestión de aparcamientos inteligentes, gestión de líneas de productividad entre otras. (Kimaldi, s.f.). Para mayor información acerca de este lector RFID revisar el anexo N° 2.

Características

- Frecuencia operacional: 902 - 928MHz, 865 - 868MHz, RF banda personalizable (opcional)
- Frecuencia Hopping FHSS o frecuencia fija.
- Potencia de salida: +20dBm ~ +30dBm@50Ω en carga, 1dBm en cada paso.
- Antena integrada de 12dBi de polarización lineal (personalizable horizontal/vertical).
- Interfaces de comunicación: RS-232, Wiegand 26/34
- Interfaces opcionales: RS-485, 802.11 Modulo y 10M/100M interfaces Ethernet
- Interfaces de I/O de 2 canales (Relay Output), 2 canales Trigger Input, 2 canales level Output (opcional)
- Actualización de firmware
- Plataforma de software de aplicación: API(C++, C#) kit de desarrollo
- Protocolos Tag: ISO18000-6B, EPC Class 1 Gen 2(ISO18000-6C)
- Rango de lectura: 0~15m
- Rango de escritura: 0~5m
- Indice de protección IP: IP63
- Indicador: LED y zumbador (Potencia, Antena, Lectura/Escritura)
- Potencia: DC 12V entrada, adaptador de potencia

- Consumo de energía: 2W(Standby), 10W(Trabajo)
- Dimensiones del equipo: 445mm x 445mm x 78mm (LxAxP)
- Dimensiones del paquete: 600mm x 500mm x 110mm (LxAxP)
- Peso neto: 3.5kg
- Peso bruto: 6.23kg
- Temperatura de operación: -40°C ~ +60°C
- Temperatura de almacenamiento: -60°C ~ +80°C
- Humedad: 5% ~ 95%(non-condensing)

- **RFID UHF SL130**

Este es un tipo de lector UHF integrado que puede leer etiquetas con el protocolo ISO 18000-6B/C, permite lectura y escritura, su método de conexión es serial (RS 232, RS 485).

Sus principales aplicaciones son: en la gerencia del vehículo, control de acceso de personal, anti-falsificación electrónica, monitor logístico, entre otras. (StrongLink, s.f). Para mayor información acerca de este lector RFID revisar el anexo N° 2.

Características

- Etiqueta de Lectura/Escritura: ISO18000-6B, EPC CLASS1 G2 tag
- Frecuencia:865~956MHz(ligeramente diferente en los distintos países y regiones)
- Tipo de trabajo: FHSS o frecuencia fija de emisión
- Potencia de salida:20~30dBm(ligeramente diferente en los distintos países y regiones)
- Distancia de lectura/escritura: distancia de lectura>8m,distancia de escritura >1m(dependiendo de las etiquetas)
- Velocidad de lectura: 32bits/6ms.

- Velocidad de escritura:32bits/50ms
- Fuente de energía: +9/12VDC.
- Potencia: potencia promedio<5W.
- Temperatura de trabajo:-10°C~+55°C

- **UHF-RW-MP-232-V1-7dbi**

Este lector utiliza la banda de frecuencia UHF para transmitir las señal a través de sus antenas, soporta lectura y escritura de tags, tag ID, lectura multi-tag, soporta protocolos ISO 18000-6C EPC G2 o ISO18000-6B UHF tag, su método de conexión es serial (RS 232), posee un paquete de software libre.

Sus principales aplicaciones son: en logística, sistema automático de gestión de producción, líneas de producción, entre otros. (RFID Shop, sf). Para mayor información acerca de este lector RFID revisar el anexo N° 2.

Características

- Modo de operación: lectura /escritura UHF Etiqueta ID y memoria
- Soporta lectura multi etiqueta
- Antena: integrada de polarización circular
- Frecuencia Estándar: 902-928Mhz (puede ser definida por el usuario)
- Frecuencia Europea: 865-868Mhz (puede ser definida por el usuario)
- Interfaces opcionales: RS232 / wiegand
- Potencia de Transmisión: < 30dbm (programable por software)
- Protocolos soportados: ISO18000-6C EPC G2 o ISO18000-6B etiquetas UHF
- Fuente de energía: +9V DC ,3A
- Paquete de software libre (VC, C#,VB.Net y Delphi, programa demo con código fuente)
- Antena integrada de 7 dbi
- Rango de lectura: ~3-5 metros (distancia efectiva que depende de las características de las etiquetas y del ambiente de instalación)

- Pico de potencia de transmisión: < 1 watt
- Dimensiones: (220x220x30)mm
- Peso: 2kg

Después de analizar las características de cada lector RFID se realizó una comparativa para poder determinar el más idóneo, el cual se muestra en la tabla 7.

Tabla 7.

Cuadro comparativo de los tres modelos de equipos.

Lenguaje de Programación	KIMALDI	RFID UHF SL130	UHF-RW-MP-232-V1-7dbi
Sistema de evaluación(alto/medio/bajo)			
Costos	alto	alto	Bajo
Conectividad con drivers	medio	bajo	Alto
Rango de cobertura y de lectura	alto	medio	Alto
Características técnicas	alto	medio	Medio
Riesgos(alto/medio/bajo)			
Precisión en la captura de datos	bajo	bajo	medio
Seguridad de la señales	bajo	bajo	medio
Lectura rango de alcance	bajo	bajo	Medio
Velocidad de detección de tags	medio	medio	Medio
Problemas(alto/medio/bajo)			
Interferencia entre lectores 2 o mas	bajo	bajo	Bajo
Estabilidad en la señal	bajo	bajo	Bajo
Conectividad con otros equipos	bajo	bajo	Bajo

Como resultado de la comparación se establece que el lector RFID UHF READER/WRITER es el más apto para el desarrollo del prototipo considerando sus características, ventajas, desventajas y su costo es relativamente bajo y accesible.

Analizando las hoja de datos del lector RFID UHF READER/WRITER especifica que es compatible con los lenguajes de programación VC, C#, VB.Net y Delphi; por lo que se realizará una comparativa entre los lenguajes de programación para establecer el más apto para el desarrollo del prototipo.

1.2.2 Análisis de los Lenguajes de Programación

- Visual C

Es un compilador de C/C++ para proyectos elaborados con programación estructurada creado por Microsoft para facilitar la programación en Windows, posee un alcance de variables y recursividad. Este sistema estático impide muchas operaciones no intencionadas. Como se indica en la tabla 8.

Tabla 8.
Ventajas y desventajas de Microsoft VC.

Ventajas	Desventajas
Compatible con COM y con integración de código de plataformas.	Soporta solo programación estructurada
Proporciona seguridad por medio de mecanismos de confianza intrínsecos del código.	Es compatible solo en ambientes Microsoft
Soporta el uso de punteros	Por defecto permite la programación sin declaración de variables.
Compatible con componentes XML basados en Web y conceptos de metadatos extensibles.	Sólo permite el uso de funciones de librerías dinámicas (DLL)

Para su correcto funcionamiento Microsoft Visual C necesita los siguientes requerimientos de hardware

- Procesador Pentium III, 600 MHz o superior.

- Windows 2000 professional, Windows 2000 server o posteriores.
- 900 MB en la unidad del sistema y 4,1 GB en la unidad de instalación.

- **C#**

Es un lenguaje de programación diseñado para la construcción de gran variedad de aplicaciones que se ejecutan en .NET Framework, este lenguaje es orientado a objetos y de código abierto por lo que puede ser considerado como software libre.

La sintaxis y estructura es muy parecida a C++ o java pero más rápida ya que la intención de Microsoft es facilitar la migración de estos lenguajes de programación a C#.

Este lenguaje de programación es muy potente ya que está diseñado para el desarrollo de aplicaciones de escritorio, móviles, web y otros componentes (Microsoft, 2003). En la tabla 9 se puede apreciar las principales ventajas y desventajas de C#

Tabla 9.
Ventajas y desventajas de C#.

Ventajas	Desventajas
Amplio soporte en programación orientada a objetos	C # es menos flexible que los otros lenguajes
Control de errores	Se debe tener algunos requerimientos mínimos del sistema para poder trabajar adecuadamente.
Compiladores gratuitos para códigos ensamblados en C#	Carencia de soporte incorporado para números complejos.
Todas las nuevas funciones son adaptadas primero en C#	Dificulta la organización y programación en capas.
Ideal para desarrollo rápido de aplicaciones, por su costo de mantenimiento es definitivamente bajo.	Quien no está familiarizado con ningún lenguaje de programación, le costará más trabajo iniciar con C#

Para la correcta ejecución de este lenguaje de programación necesita unas características que las detallamos a continuación:

- Windows Vista con Service Pack 2, Windows 7 o posterior.
- Procesador de 32 o 64 bits a 800 megahercios.
- 512 megabytes de memoria del sistema.
- Tarjeta gráfica de clase de DirectX 9.
- 32 MB de memoria de gráficos.
- Disco duro de 20 gigabytes con al menos 15GB de espacio libre

- Visual Basic .Net

Visual Basic .Net es un lenguaje de programación orientado a objetos ya que tiene la posibilidad de trabajar con interfaces, sobrecarga y herencia. Este lenguaje incluye el subprocesamiento libre y el control de excepciones estructuradas y al utilizar .NET framework y CLR (Common Language Runtime)

que proporciona interoperabilidad, seguridad y compatibilidad con versiones mejoradas.

En la tabla 10 se indican las principales ventajas y desventajas de Visual Basic .Net.

Tabla 10.
Ventajas y desventajas de Visual Basic .Net

Ventajas	Desventajas
Soporta programación orientada a objetos	No es multiplataforma.
Control de excepciones estructurado	Es un lenguaje basado en objetos pero no aplica en su totalidad la filosofía de orientación a objetos.
Tiene características de herencia, interfaces y sobrecarga	No tiene instrucciones de preprocesamiento.
Pueden crearse aplicaciones multiproceso	Dificulta la organización y programación en capas.
Permite crear atributos personalizados.	No maneja adecuadamente los apuntadores de memoria

Los requerimientos mínimos para que Visual Basic .Net funcione de una manera fluida son:

- Procesador Pentium III, 600 MHz o superior.
- Windows 2000 professional, Windows 2000 server o posteriores.
- 900 MB en la unidad del sistema y 4,1 GB en la unidad de instalación.

- Delphi

Es un lenguaje de programación orientado a una forma de programación más visual y pretende agilizar la creación de software basándose en este esquema, el lenguaje de programación que utiliza Delphi internamente es Object Pascal que es una versión más actual del conocido lenguaje de programación pascal con la diferencia que este ya es orientado a objetos.(Ramírez J.,2008)

En la tabla 11 encontramos las principales ventajas y desventajas de Delphi.

Tabla 11.
Ventajas y desventajas de Delphi.

Ventajas	Desventajas
Soporta programación orientada a objetos.	Poco Soporte
Lenguaje multiplataforma	El funcionamiento de la web es a través de intranet
Fácil compatibilidad con hardware	Suele caer en incompatibilidad de versiones
Programación mas visual	No posee repositorios de componentes

El lenguaje de programación Delphi debe ser instalado en un sistema operativo con las siguientes características:

- Windows XP o posterior.
- Pentium 800 Mhz, o superior.
- Memoria interna de 256 MB
- Espacio de disco duro 250 MB (tamaño del programa).

- **Elección del lenguaje de programación.**

Después de analizar las características, las ventajas y desventajas de los tres lenguajes de programación preseleccionados se establece una comparativa para poder determinar el más idóneo, la cual se muestra en la tabla 12.

Tabla 12.

Calificación de factibilidad de los lenguajes de programación.

Lenguaje de Programación	VC	C#	VB .Net	Delphi
Sistema de evaluación(alto/medio/bajo)				
Costos	bajo	bajo	bajo	bajo
Características de diseño, interfaz y escalabilidad	bajo	alto	bajo	alto
Seguridad, garantía y conectividad	medio	alto	medio	medio
Capacidad a conexión a bases de datos	medio	alto	medio	alto
Riesgos(alto/medio/bajo)				
Estabilidad del Lenguaje	medio	alto	medio	Medio
Seguridad del Código	alto	alto	medio	Medio
Adaptación de controladores RFID	medio	alto	medio	Medio
Multiplataforma	bajo	alto	bajo	Alto
Problemas(alto/medio/bajo)				
Incompatibilidad con lenguajes de programación antiguos	medio	bajo	medio	Bajo
Disponibilidad de manejo de métodos	bajo	bajo	bajo	Bajo
Conexión con equipos RFID	medio	bajo	medio	Medio

Como resultado de la comparación se establece que el lenguaje de programación más adecuado para el desarrollo del sistema prototipo de Gestión de inventario es C# considerando sus características, ventajas y desventajas.

Para poder gestionar de una manera eficiente todos los datos e información del sistema, así como la obtención y consultas a la información en tiempo real es necesario el uso de una base de datos, por lo que a continuación se realizará una comparación entre las tres principales bases de datos (SQL Server, Oracle y MySQL Server) que existen en el mercado para elegir la más idónea en la gestión de información del proyecto.

1.2.3 Análisis del Sistema Gestor de base de Datos

- SQL Server

Es un sistema de gestión de base de datos relacionales que se basa en los lenguajes T-SQL y ANSI SQL desarrollado por Microsoft para la línea de negocio y almacenamiento de datos como también para las soluciones de comercio electrónico. (Microsoft, 2014). Sus principales características son:

- Instalación, distribución y uso de manera sencilla.
- Soporta transacciones simultáneas.
- Fácil integración con otros servicios de Microsoft.

Sus principales ventajas y desventajas se muestran en la tabla 13.

Tabla 13.

Ventajas y desventajas de SQL Server.

Ventajas	Desventajas
Tiene sentencias estándar, las que sirven para cualquier lenguaje de base de datos.	Tiene poca capacidad de almacenamiento la versión gratuita
Consta de escalabilidad, estabilidad y seguridad.	Utiliza considerable memoria RAM que utiliza para la instalación y uso del software.
Soporta procedimientos almacenados.	Tiene ciertas restricciones según la versión instalada
Permite la administración desde otros servidores	Hay una sola versión gratuita el resto es de paga.

Para instalar SQL server se necesita de un equipo de cómputo con las siguientes características:

- Procesador compatible con Pentium III o superior.
- Velocidad del procesador de 2GHz o más.
- Windows XP o posterior.
- Memoria RAM mínima 512 MB.

- Oracle

Es un sistema de gestión de base de datos relacional basado en una tecnología cliente/servidor, por lo que primero se instala y configura el servidor para que puedan acceder a la base de datos con herramientas de desarrollo como Oracle Designer y Oracle Developer que son herramientas básicas para la programación sobre esta base de datos. (Masip David, 2002).

Entre sus principales características están:

- Soporte de transacciones
- Permite crecer rápidamente.
- Multiplataforma
- Administración sencilla.

Sus ventajas y desventajas se muestran en la tabla 14

Tabla 14.
Ventajas y desventajas de Oracle.

Ventajas	Desventajas
Posee gran capacidad ilimitada de datos.	Elevado el coste de la formación y certificación en Oracle.
Puede ejecutarse en todas las plataformas	El precio. Incluso las licencias de Personal Oracle son excesivamente caras.
Permite el uso de particiones para la mejora de la eficiencia	Necesita permanecer siempre actualizada por los parches de seguridad
Oracle es la base de datos con más orientación hacia INTERNET	Oracle mal configurado puede ser lento y su rendimiento bajo.

Para instalar Oracle se necesita un equipo con las siguientes características:

- Sistema Operativo Windows XP o posterior, Linux.
- Memoria RAM 1GB.
- 500 MB de disco duro.

- JVM J2SE 5.0, JAVA SE 6

- **My SQL Server**

Es un sistema de gestión de base de datos relacional de código abierto que se basa en SQL, a pesar de que se puede utilizar en una amplia gama de aplicaciones se lo asocia más con aplicaciones basadas en la web (Rouse Margaret, 2015).

Entre sus principales características están:

- Facilidad de instalación y uso.
- Herramientas de conectividad con otros programas.
- Gran cantidad de librerías.
- Open Source.

Sus ventajas y desventajas se muestran en la tabla 15.

Tabla 15.
Ventajas y desventajas de MySQL Server.

Ventajas	Desventajas
MySql está escrito en C y C++.	No sincroniza los datos con otras bases de datos replicas.
Completo soporte para operadores y funciones	Un gran porcentaje de las utilidades de MySql no están documentadas.
Uso de multi-hilos mediante el uso del kernel	No tiene integridad referencial.
Soporta gran cantidad de datos	No es intuitiva, se necesita instrucción previa.
Los clientes se conectan al servidor MySql usando sockets TCP/IP	Función de conversión no soporta ciertos tipos de datos.

Para que se pueda instalar y ejecutar correctamente este sistema gestor de base de datos se necesita los siguientes requerimientos:

- Sistema Operativo Windows, Linux y Unix.
- 512 MB de memoria RAM.

- 1024 MB de máquina virtual.
 - 1GB de disco duro (libre para la aplicación).
- **Elección del Gestor de Base de Datos.**

Después de analizar las características, las ventajas y desventajas de los gestores de base de datos se realizarán una comparativa para poder determinar el más idóneo, el cual se muestra en la tabla 16.

Tabla 16.

Calificación de factibilidad de las bases de datos.

Lenguaje de Programación	SQL Server	Oracle	MySQL
Sistema de evaluación(alto/medio/bajo)			
Costos	medio	alto	Bajo
Capacidad de Almacenamiento	alto	alto	Alto
Riesgos(alto/medio/bajo)			
Estabilidad de la información	bajo	bajo	medio
Seguridad de la información	bajo	bajo	medio
Entornos gráficos	bajo	medio	medio
Multiplataforma	bajo	bajo	Bajo
Problemas(alto/medio/bajo)			
Incompatibilidad con lenguajes de programación antiguos	bajo	bajo	Bajo
Estabilidad en ejecuciones y búsquedas	bajo	bajo	Bajo
Seguridad en el almacenamiento de información	bajo	bajo	medio

Como resultado de la comparación de los tres gestores de base de datos preseleccionados se obtuvo que SQL Server es el más idóneo para llevar la gestión y administración de información del sistema prototipo.

Para un funcionamiento normal y fluido del software prototipo de gestión de inventario, su base de datos, enlazados al equipo RFID correspondiente según lo analizado anteriormente sobre cada elemento seleccionado los requisitos mínimos del PC son:

- Procesador Core 2 Duo 1.6GHz o superior.
- Disco duro 40Gb.
- Mínimo 1Gb de RAM o superior.
- Sistema Operativo Windows 7 de 32bits o superior.
- Monitor y adaptador de video.
- Conexión a puerto USB 2.0 o superior.

1.3 Propiedades del Sistema.

El prototipo de gestión de inventario está estructurado para facilitar la acción de realizar un inventario, ya que a nivel de hardware el lector RFID y Tags se enlazan a nuestro software a través de una PC para administrar cada producto que se encuentra en bodega.

Establecer una adecuada estructura para el diseño del prototipo es muy importante por lo que la detallaremos a continuación:

- Determinar el ambiente (tipo de bodega base) en el que se va implementar el prototipo para establecer los requerimientos para su desarrollo.
- Definir los requerimientos mínimos de hardware y software para el desarrollo de este prototipo.
- En base a los requerimientos establecer un diseño adecuado del sistema para que el usuario de una manera sencilla e intuitiva pueda realizar la gestión de inventario.
- Finalmente implementado el prototipo se necesita realizar pruebas de funcionamiento para verificar la funcionalidad y rendimiento en base al ambiente de desarrollo y requerimientos establecidos en los primeros puntos.

El desarrollo de este prototipo está estructurado en base a los métodos deductivo y experimental. El método deductivo se utilizará ya que se pretende implementar una aplicación particular (específica) de la tecnología RFID para la realización de este sistema. El método experimental se utilizará para el diseño e implementación de un prototipo que enlace adecuadamente software y hardware para la gestión de inventarios.

El método deductivo establece que partiendo de la gran cantidad de usos y aplicaciones que tiene la tecnología RFID en la actualidad, centrarnos en un uso específico que es la administración de un sistema de inventario automatizado. En el método experimental está claramente evidenciado en la creación y pruebas de funcionamiento de la aplicación que administre de forma fácil y eficiente los artículos en existencia.

2. CAPÍTULO II: DISEÑO E IMPLEMENTACIÓN DEL SISTEMA

En este capítulo se presentará el diseño e implementación del prototipo de gestión de inventario, para lo cual luego de realizar un análisis de requerimientos, se definió trabajar con una base de datos para almacenar la información de los productos que van a ser registrados y posteriormente consultados, y con una interfaz de usuario que permita de una manera sencilla llevar el control del inventario de una empresa.

La base de datos estará desarrollada en SQL Server y la interfaz de usuario en C# ya que este lenguaje de programación ofrece una interfaz amigable y facilidad de manejo de funciones para su programación. Una vez definidos los programas a utilizar se procederá con la creación de la base de datos que contiene las diferentes tablas para el ingreso de información de productos,

usuarios, permisos, seguridades y etiquetas, al igual que los procedimientos almacenados para realizar las diferentes consultas a la base de datos.

En la interfaz gráfica se tendrá principalmente los módulos que realicen el control primero de los usuarios, donde permitirá crear usuarios, modificar usuarios, eliminar usuarios, manejar diferentes permisos de usuarios y luego de los productos, donde de igual forma se pueda realizar la creación de nuevos productos, modificar un producto, eliminar un producto, buscar un producto y una sección principal que es la que permite la asignación de una etiqueta a un producto y a través de esto realizar el manejo del inventario, consultando la cantidad de productos totales en bodega, los productos que ya salieron y los productos nuevos.

De esta forma el usuario tendrá de manera fácil e intuitiva una interfaz que a través de módulos lo guíen en la administración y control de inventario, la cual vinculada a la tecnología RFID realiza un conteo de productos en bodega o que ya han salido, para facilitar el control de inventario.

2.1 Desarrollo de la Aplicación.

Para el desarrollo de este sistema prototipo se analizó el tipo de datos que se va a manejar y qué usuarios serían los que van a manipular el programa, una vez identificado esto se realizó el formato de la parte gráfica colocando todas y cada una de las funcionalidades que se desea brindar en la aplicación, como por ejemplo el manejo de usuarios, la aplicación debe tener la versatilidad para crear nuevos usuarios, modificarlos, eliminarlos, actualizarlos y sobre todo darles privilegios, de las vistas que van a acceder al momento de ingresar al aplicativo; con lo cual se sugiere que debe existir un inicio de sesión previo, para poder validar los niveles de acceso que tendrán los usuarios.

También se requiere tener un completo control de los productos que se van a inventariar como, ingreso de un producto nuevo, actualización de un producto existente, eliminación de un producto en caso de darlo de baja, asignación de un código al producto, asignación de una etiqueta a los productos, visibilidad gráfica de una imagen del producto, validaciones de qué hacer cuando ingresa un nuevo producto, contabilizar los productos existentes en inventario, contabilizar los productos que salieron de la bodega, para ello juega un papel muy importante el lector RFID, pues es a través de él que se va a llevar este control de ingreso, salida y asignación de una etiqueta a cada producto, haciendo mucho más fácil el administrar un inventario cuando la cantidad de productos no se puede manejar de manera manual.

Una vez establecido el modelo gráfico o apariencia gráfica que va a tener el aplicativo, se procede a analizar el tipo de programación que se va a aplicar para el desarrollo y se encuentra que es mucho más escalable y ordenado el uso de una programación por capas, las cuales ayudan a definir de manera más específica y diferenciada la funcionalidad de cada nivel en la programación.

La aplicación desarrollada responde a un modelo de programación en capas, él cual es muy útil cuando se necesita separar la aplicación en componentes que se encuentren en diferentes niveles funcionales, para de esta manera hacer más sencilla la corrección y detección de errores durante el desarrollo de la aplicación. Adicionalmente un modelo en capas ayuda a dar escalabilidad a una aplicación, pues las capas de bajo nivel suelen mantenerse inalterables y el cambio se evidencia en la capa que interactúa directamente con el usuario. Basado en este concepto se determinó el uso de 3 capas bien definidas: la capa de datos, capa de negocio y la interfaz de usuario como se muestra en la figura 18.

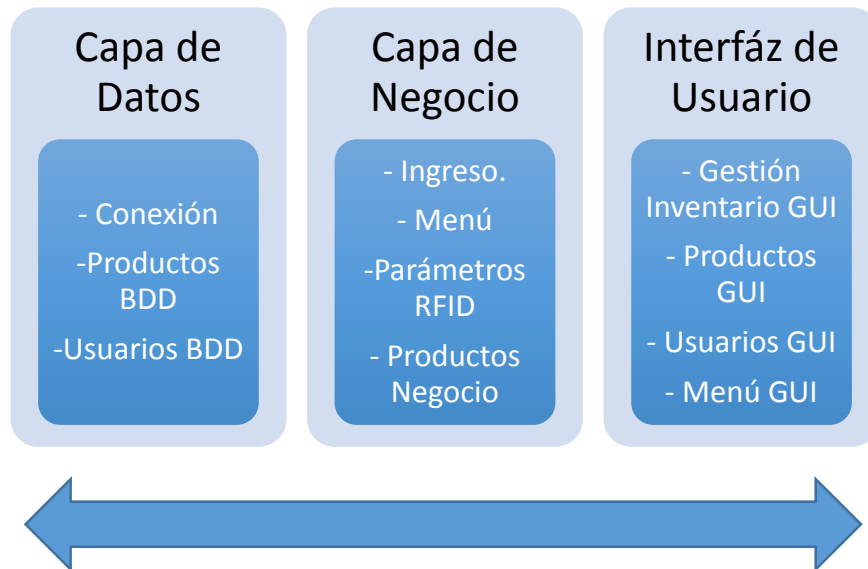


Figura 18 Esquema Programación por capas del prototipo

2.1.1 Capa de Datos

Aquí se encuentran todos los datos cuya función son de recuperar, almacenar o eliminar la información con la cual va a trabajar la aplicación. Esta capa tiene conexión directa a la base de datos, en la aplicación esta capa está definida en el proyecto Conexión, la misma que invoca a un cliente SQL utilizado para la apertura y cierre de la conexión. Adicionalmente están los proyectos productosBDD y usuariosBDD. El proyecto productosBDD contiene todos los datos de un producto y además los procedimientos almacenados de la base de datos relacionados con los productos. El proyecto usuariosBDD contiene todas las llamadas a los procedimientos almacenados que están en la base de datos, relacionados con los usuarios.

La base de datos contiene todas las tablas que relacionan la funcionalidad de la aplicación y almacenan toda la información de los productos. Como se muestra en la figura 2.1.

- **Proyecto Conexión**

Contiene una clase clsConexion la cual invoca a un SqlConnection para abrir y cerrar la conexión con la base de datos y las credenciales de acceso de seguridad a la misma. Como se muestra en la Figura 19

La configuración mostrada anteriormente se encarga de realizar la conexión con la base de datos, para lo cual se manejó dos clases de tipo público cuya función es abrir la conexión a la base de datos y la otra cierra dicha conexión. Como se muestra en la tabla 17.

Tabla 17.
Clases del proyecto Conexión

CLASE	DESCRIPCIÓN
Abrir	Función que abre la conexión a la base de datos.
Cerrar	Función que cierra la conexión a la base de datos.

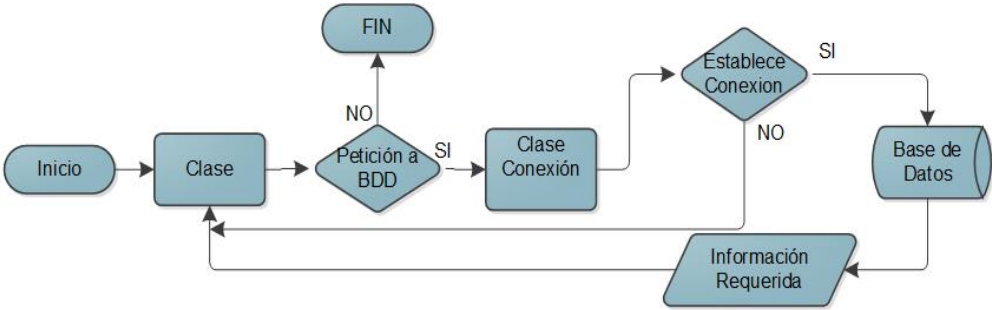


Figura 19. Clase Conexión.

- **Proyecto Productos BDD**

Contiene la programación para mostrar los datos de un producto determinado, al igual que la llamada a los procedimientos para ingresar, actualizar, eliminar y mostrar un registro de un producto específico, como se muestra en la Figura 20 y la tabla 18.

Tabla 18.
Clases del proyecto Productos BDD

CLASE	DESCRIPCIÓN
AlmacenarRegistros	Función que permite el ingreso de Código, Nombre, Descripción, Cantidad, Precio, Fecha, Imagen, Marca, Grupo, CódigoFábrica, Serie, Origen, Garantía y Estado, para almacenar un producto nuevo en la base de datos.
ActualizarRegistros	Función que permite el ingreso de Código, Nombre, Descripción, Cantidad, Precio, Fecha, Imagen, Marca, Grupo, CódigoFábrica, Serie, Origen, Garantía y Estado, para actualizar los datos de un producto existente y guardar los cambios en la base de datos.
MostrarRegistroporCodigo	Función que permite realizar la consulta a la base de datos, por medio del ingreso del código del producto y mostrar los datos del mismo.
MostrarRegistroporNombre	Función que permite realizar la consulta a la base de datos, por medio del ingreso del nombre del producto y mostrar los datos del mismo.
MostrarRegistroporMarca	Función que permite realizar la consulta a la base de datos, por medio del ingreso de la marca del producto y mostrar los datos del mismo.
MostrarRegistroporSerie	Función que permite realizar la consulta a la base de datos, por medio del ingreso de la serie del producto y mostrar los datos del mismo.
MostrarRegistro	Función que permite consultar y mostrar todos los registros almacenados en la base de datos.
EliminarRegistroporCodigo	Función que permite eliminar un producto de la base de datos, por medio del ingreso del código del mismo.
EliminarRegistroporNombre	Función que permite eliminar un producto de la base de datos, por medio del ingreso del nombre del mismo.

EliminarRegistroporSerie	Función que permite eliminar un producto de la base de datos, por medio del ingreso de la serie del mismo.
RecuperarProductoporCódigo	Función que permite mostrar todos los datos de un producto consultado por código.
RecuperarProductoporNombre	Función que permite mostrar todos los datos de un producto consultado por nombre.
RecuperarProductoporSerie	Función que permite mostrar todos los datos de un producto consultado por serie.
IngresarTag	Función que permite el ingreso de un nuevo tag o etiqueta a la base de datos, asociandolo a un código de producto e identificador de estado.
ActualizaTag	Función que permite la actualización de una etiqueta en la base de datos, asociandolo a un código de producto e identificador de estado.
RecuperarTagporCodigo	Función que permite mostrar todos los datos de un tag consultado por código.
AlmacenarTag	Función que permite asignar una etiqueta a un código de producto y este a su vez se va almacenando en la base de datos.
MostrarTags	Función que permite mostrar todas las etiquetas guardadas en la base de datos.
ActualizaEtiquetaporTag	

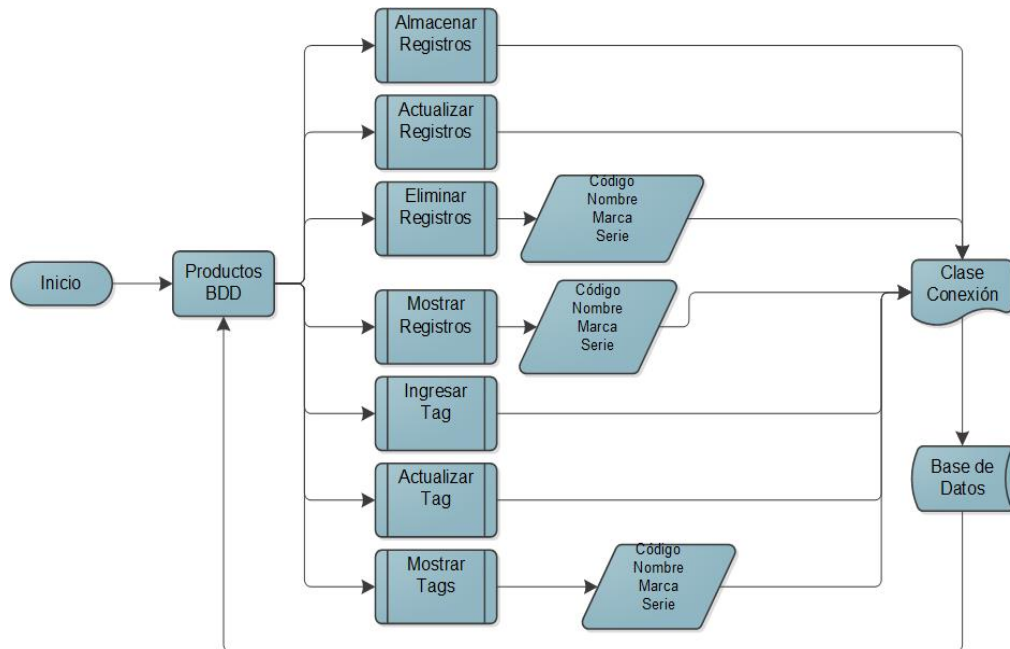


Figura 20. Clase Productos BDD

- Proyecto Usuarios BDD

Contiene todas las llamadas a los store procedures para almacenar, actualizar, eliminar y recuperar un usuario de la base de datos. Adicionalmente contiene las funciones para mostrar un usuario por nombre, por CI, actualizar permisos por CI y mostrar permisos por CI. Como se muestra en la Figura 21 y la tabla 19.

Tabla 19.
Clases del proyecto Usuario BDD

CLASE	DESCRIPCIÓN
AlmacenarUsuario	Función que permite el ingreso de cédula de identidad, apellidos y nombres, nombre de usuario, dirección, teléfono, contraseña, pregunta de seguridad, respuesta de seguridad y cargo, para almacenar un usuario nuevo en la base de datos.
ActualizarUsuariosCI	Función que permite el ingreso de cédula de identidad, apellidos y nombres, nombre de usuario, dirección, teléfono, contraseña, pregunta de seguridad, respuesta de seguridad y cargo, para actualizar los datos de un usuario existente y guardar los cambios en la base de datos.
EliminarUsuarioCI	Función que permite eliminar un usuario de la base de datos, por medio del ingreso de la cédula de ciudadanía.
RecuperarUsuario	Función que permite mostrar todos los datos de un usuario específico.
MostrarUsuarioporNombre	Función que permite realizar la consulta a la base de datos, por medio del ingreso del nombre del usuario y mostrar los datos del mismo.
RecuperarUsuarioporCI	Función que permite mostrar todos los datos de un usuario consultado por el número de cédula de identidad.
ActualizarPermisosCI	Función que permite actualizar los permisos de un usuario ingresando su número de cédula de identidad y los módulos a los que desea tener permisos.
MostrarPermisosporCI	Función que permite mostrar los permisos de un usuario, ingresando su número de cédula de identidad.

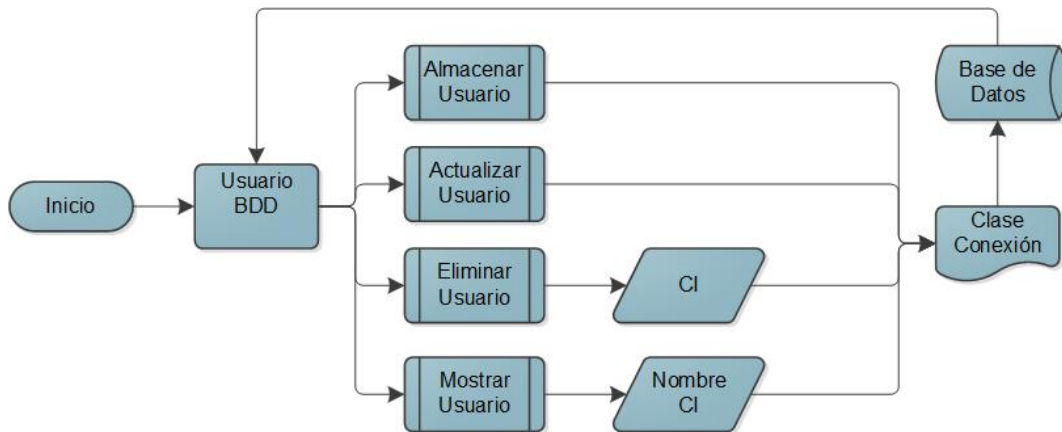


Figura 21. Clase Usuarios BDD

2.1.2 Capa de Negocio

Aquí se realizan todas las validaciones de los datos de entrada y salida, es donde se desarrolla toda la lógica de negocio, donde se realizan las funciones que manejan los datos provenientes de la capa conexión y se utilizan las reglas necesarias para la aplicación. Inclusive en esta capa se realizan las validaciones de formato.

En el diseño desarrollado la capa de negocio corresponde a la programación que va detrás de los formularios, debido a que es ahí donde se realizan todas las validaciones de los datos que ingresan o salen del inventario. Las clases utilizadas aquí son: Inicio de Sesión, menú, parámetros RFID (donde se realiza la conexión con el lector RFID previamente escogido) y productos negocio. Como se muestra en la figura 18.

- Inicio de Sesión

Método que contiene la validación de los usuarios que van a acceder a la aplicación. Esta validación se la realiza con un usuario y contraseña únicos por cada persona que necesite ingresar al sistema, considerando de forma particular los roles de cada uno, es decir si su rol va a ser de administrador

(quien tendrá acceso a todos los módulos), operador (quien tendrá acceso a crear y asignar productos) y monitoreo (quien podrá solo ver los productos almacenados). En la Figura 22 se muestra el proceso de inicio de sesión.

El proyecto inicio de sesión está compuesto de las clases descritas en la tabla 20.

Tabla 20.

Clases del proyecto Inicio de Sesión.

CLASES	DESCRIPCIÓN
btnIngresar_Click	Función que valida en la base de datos el usuario ingresado y según los permisos que este tenga le permite el ingreso al formulario de menú al hacer click en el botón INGRESAR.

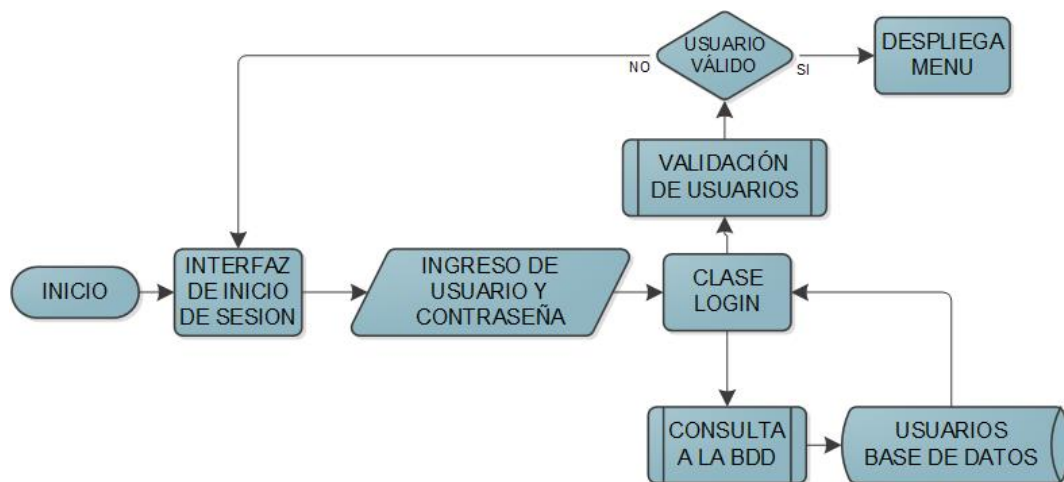


Figura 22. Clase Inicio de Sesión

- Menú

El desarrollo del menú principal guía de una manera sencilla al usuario dentro de sus cuatro principales módulos de administración que son: Parámetros RFID, Productos, Usuarios, Gestión de inventario y adicionalmente tiene la opción para poder cerrar sesión del sistema.

La programación del menú redirecciona a los diferentes módulos de la aplicación y la invocación a los diferentes formularios frmInventario, frmUsuario, frmProductos, frmGestionInventario. Como se muestra en la Figura 23 y la tabla 21.

Tabla 21.
Clases del proyecto Menú

CLASES	DESCRIPCIÓN
btnParametrosRfid_Click	Función que permite el ingreso al formulario de Parámetros RFID, al hacer click en el botón PARAMETROS RFID.
btnUsuarios_Click	Función que permite el ingreso al formulario de Usuarios, al hacer click en el botón USUARIOS.
btnProductos_Click	Función que permite el ingreso al formulario de Productos, al hacer click en el botón PRODUCTOS.
btnInventario_Click	Función que permite el ingreso al formulario de Inventario, al hacer click en el botón GESTION INVENTARIO.
btnCerrar_Click	Función que permite al usuario cerrar la sesión actual, al hacer click en el botón CERRAR SESION.

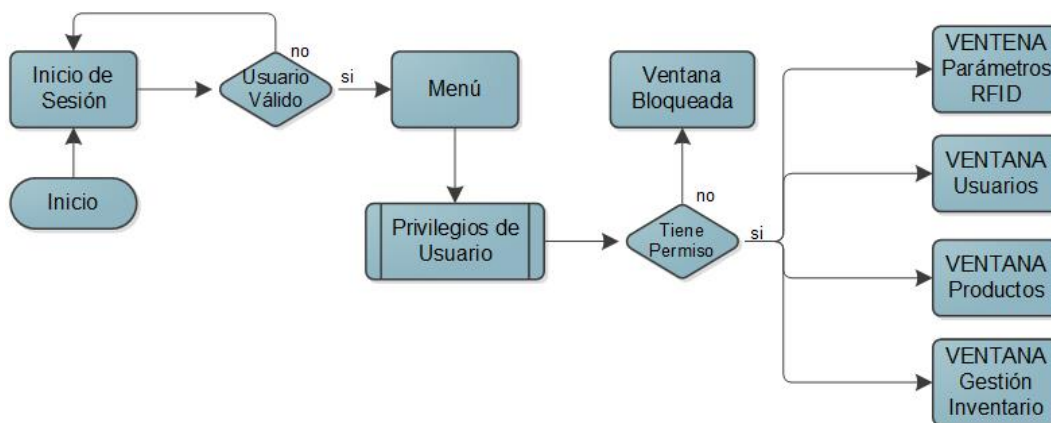


Figura 23. Clase Menú

- Parámetros RFID

En este método se definen los parámetros para la comunicación con el equipo lector UHF, aquí están detallados los puertos de comunicación y la información por defecto del lector como: tipo, dirección, frecuencia mínima y máxima, versión, potencia, protocolo a utilizar, tiempo de escaneo. Adicionalmente se tiene la opción de cargar los datos por defecto o cambiar los parámetros antes mencionados a los que se desee.

También tenemos la posibilidad de listar las etiquetas encontradas usando el protocolo EPCC1-G2, para ello se hace uso de un intervalo de lectura que está en el rango de 40 a 300ms.

En la tabla 22 se detallan las principales funciones utilizadas

Tabla 22.
Clases del proyecto Parámetros RFID

CLASES	DESCRIPCIÓN
GetReturnCodeDesc	Función que obtiene un código descendente
GetErrorCodeDesc	Función que abstrae el error del código
HexStringToByteArray	Función de cambio de tipo de dato string hexadecimal a un array de bytes.
ByteArrayToHexString	Función de cambio de un tipo de dato Byte Array a uno de tipo String hexadecimal.
AddCmdLog	Función que añade un log de comandos.
AddCmdLog	Función que añade un log de comandos.
ClearLastInfo	Función para borrar los últimos registros de información de la pantalla
InitComList	Función que despliega una

	lista de puertos COM para la comunicación serial.
InitReaderList	Función para leer la lista de puertos.
Form1_Load	Función que permite visualizar los botones, listas, grupos, barras y demás opciones que se encuentran en el formulario.
OpenPort_Click	Función que permite la apertura del puerto al hacer click en el botón Open COM Port.
ClosePort_Click	Función que permite el cierre del puerto al hacer click en el botón Close Port.
CheckBox_SameFre_CheckedChanged	Función que toma el valor seleccionado del check box SameFre
ComboBox_dfreSelect	Función que recoge el valor seleccionado en el combo box dfre
ChangeSubItem	Función que cambia de nombre los subitems
Timer_Test_Tick	Función de tiempo (Tick) programado para la lectura de tags.
SpeedButton_Read_G2_Click	Función mediante la cual lee tags cuando se de un click en el botón read.
Timer_G2_Read_Tick	Función de lectura a través de intervalos de tiempo de reloj.
Button_DataWrite_Click	Función que mediante el click en el botón DataWrite permite escribir datos en los tags.
Button_SetProtectState_Click	Función que mediante el click en el botón SerProtectState evita que cambien el estado del tag.
Button_DestroyCard_Click	Función que mediante el click en el botón DestroyCard permite eliminar una tarjeta del sistema.

- **Productos Negocio**

Contiene la programación para mostrar los datos de un producto determinado, al igual que la llamada a los procedimientos para ingresar, actualizar, eliminar y mostrar un registro de un producto específico. Como se muestra en la Figura 24.

Las principales clases que tiene este proyecto se muestran en la tabla 23.

Tabla 23.

Clases del proyecto Productos Negocio

CLASE	DESCRIPCIÓN
AlmacenarRegistros	Función que permite el ingreso de Código, Nombre, Descripción, Cantidad, Precio, Fecha, Imagen, Marca, Grupo, CódigoFábrica, Serie, Origen, Garantía y Estado, para almacenar un producto nuevo en la base de datos.
ActualizarRegistros	Función que permite el ingreso de Código, Nombre, Descripción, Cantidad, Precio, Fecha, Imagen, Marca, Grupo, CódigoFábrica, Serie, Origen, Garantía y Estado, para actualizar los datos de un producto existente y guardar los cambios en la base de datos.
MostrarRegistroporCodigo	Función que permite realizar la consulta a la base de datos, por medio del ingreso del código del producto y mostrar los datos del mismo.
MostrarRegistroporNombre	Función que permite realizar la consulta a la base de datos, por medio del ingreso del nombre del producto y mostrar los datos del mismo.
MostrarRegistroporMarca	Función que permite realizar la consulta a la base de datos, por medio del ingreso de la marca del producto y mostrar los datos del mismo.
MostrarRegistroporSerie	Función que permite realizar la consulta a la base de datos, por medio del ingreso de la serie del producto y mostrar los datos del mismo.
MostrarRegistro	Función que permite consultar y mostrar todos los registros almacenados en la

	base de datos.
EliminarRegistroporCodigo	Función que permite eliminar un producto de la base de datos, por medio del ingreso del código del mismo.
EliminarRegistroporNombre	Función que permite eliminar un producto de la base de datos, por medio del ingreso del nombre del mismo.
EliminarRegistroporSerie	Función que permite eliminar un producto de la base de datos, por medio del ingreso de la serie del mismo.
HexStringToByteArray	Función que permite la conversión de hexadecimal a byte.
ByteArrayToHexString	Función que permite la conversión de byte a hexadecimal.
Tags_asignacion	Función que permite la asignación de un nuevo tag o etiqueta a la base de datos, asociándolo a un código y a un Tagpass.
RecuperarProductoporCódigo	Función que permite mostrar todos los datos de un producto consultado por código.
RecuperarProductoporNombre	Función que permite mostrar todos los datos de un producto consultado por nombre.
RecuperarProductoporSerie	Función que permite mostrar todos los datos de un producto consultado por serie.
IngresarTag	Función que permite el ingreso de un nuevo tag o etiqueta a la base de datos, asociándolo a un código de producto e identificador de estado.
ActualizaTag	Función que permite la actualización de una etiqueta en la base de datos, asociándolo a un código de producto e identificador de estado.
RecuperarTagporCodigo	Función que permite mostrar todos los datos de un tag consultado por código.
AlmacenarTag	Función que permite asignar una etiqueta a un código de producto y este a su vez se va almacenando en la base de datos.
MostrarTags	Función que permite mostrar todas las etiquetas guardadas en la base de datos.
ActualizaEtiquetaporTag	

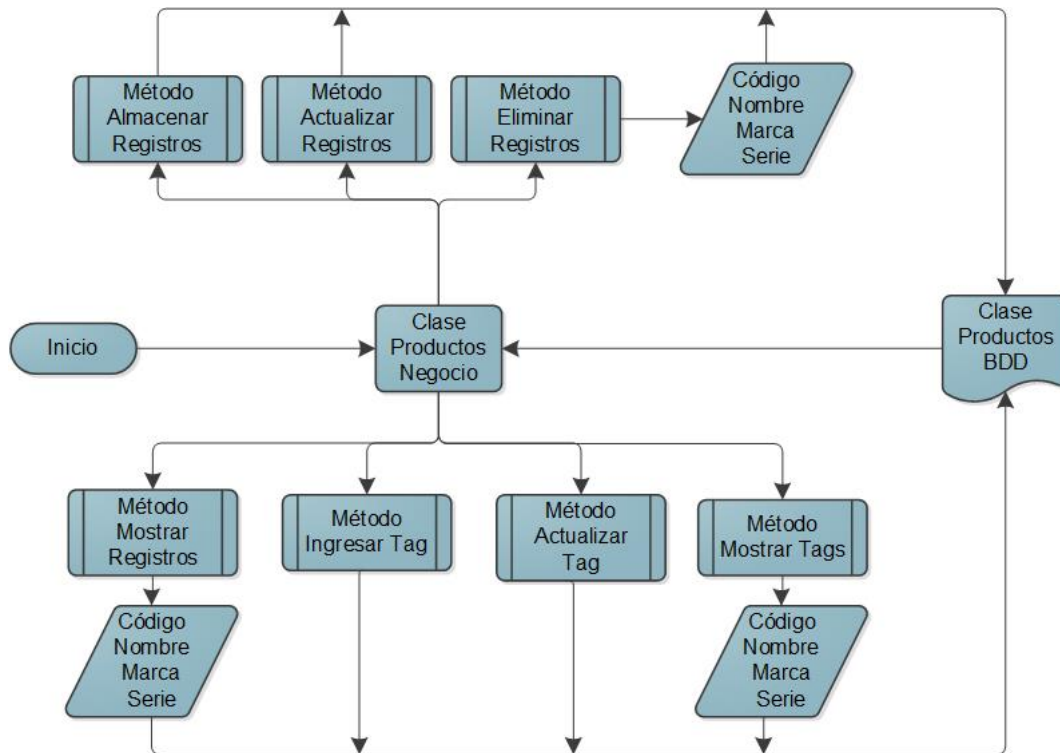


Figura 24. Clase Productos Negocio

- Usuario Negocio

Contiene los métodos para actualizar, eliminar, recuperar, actualizar permisos, mostrar permisos de un usuario por CI, recuperar un usuario y mostrar un usuario por nombre. Como se muestra en la Figura 25.

Las principales clases que se manejan en este proyecto son las descritas en la tabla 24.

Tabla 24.
Clases del proyecto Usuario Negocio

CLASE	DESCRIPCIÓN
NuevoUsuario	Función que permite el ingreso de cédula de identidad, apellidos y nombres, nombre de usuario, dirección, teléfono, contraseña, pregunta de seguridad, respuesta de seguridad y cargo, para almacenar un usuario nuevo en la base de datos.
ActualizarUsuariosCI	Función que permite el ingreso de cédula de identidad, apellidos y nombres, nombre de usuario, dirección, teléfono, contraseña, pregunta de seguridad, respuesta de seguridad y cargo, para actualizar los datos de un usuario existente y guardar los cambios en la base de datos.
EliminarUsuarioCI	Función que permite eliminar un usuario de la base de datos, por medio del ingreso de la cédula de ciudadanía.
RecuperarUsuario	
MostrarUsuarioporNombre	Función que permite realizar la consulta a la base de datos, por medio del ingreso del nombre del usuario y mostrar los datos del mismo.
RecuperarUsuarioporCI	Función que permite mostrar todos los datos de un usuario consultado por el número de cédula de identidad.
ActualizarPermisosCI	Función que permite actualizar los permisos de un usuario ingresando su número de cédula de identidad y los módulos a los que desea tener permisos.
MostrarPermisosporCI	Función que permite mostrar los permisos de un usuario, ingresando su número de cédula de identidad.

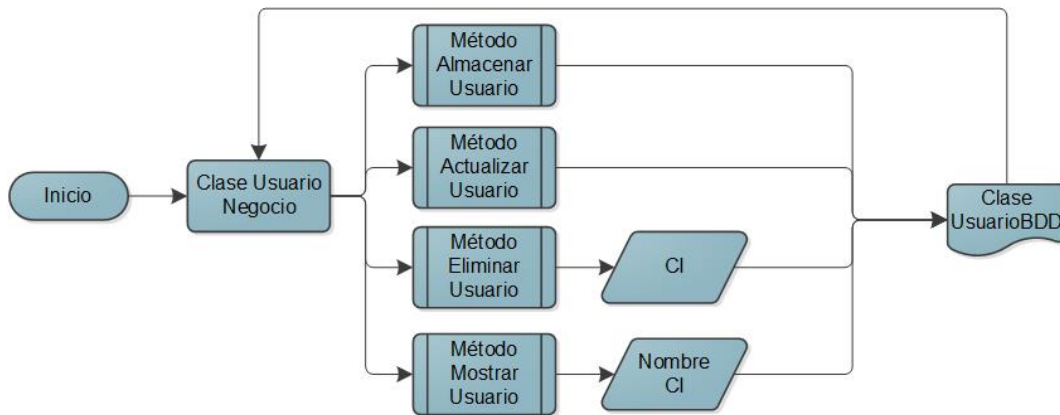


Figura 25. Clase Usuarios Negocio

2.1.3 Interfaz de Usuario

Aquí se desarrolla todo el entorno gráfico de la aplicación y es la capa que tiene relación directa con el usuario final, pues es a través de ella que se va a realizar toda la interacción con los datos e información del sistema. La creación de esta capa está desarrollada en diferentes formularios que de manera intuitiva van guiando al usuario en el manejo de la aplicación. Como se muestra en la figura 26.

En el prototipo la capa gráfica está desarrollada por los siguientes proyectos: gestión inventario GUI, productos GUI, usuario GUI y los siguientes módulos: parámetros RFID, usuarios, productos y gestión de inventario. Como se muestra en la figura 18.

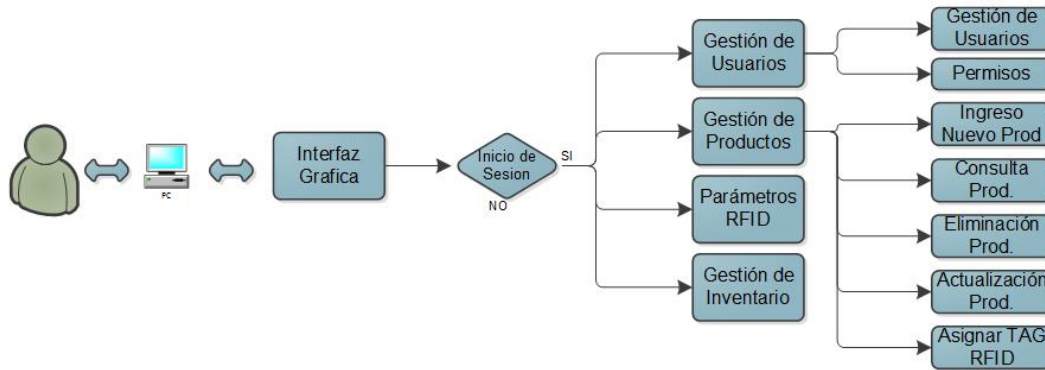


Figura 26. Capa Gráfica

2.1.3.1 Gestión de Inventario GUI

En este formulario se puede consultar el total de productos que existen actualmente en stock y han sido asignados una etiqueta, para consultar un producto en particular se tiene los criterios de búsqueda por código, nombre, marca, serie, todos. Como se muestra en la Figura 27.

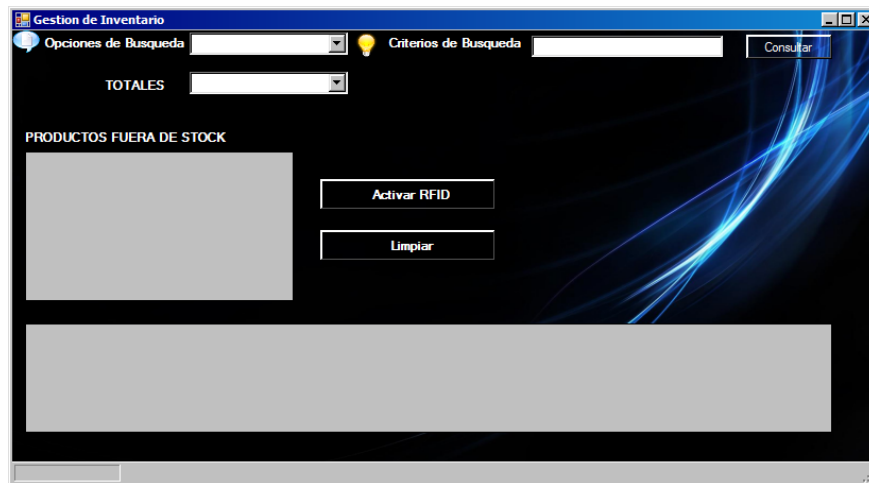


Figura 27. Pantalla de Gestión de Inventario

El proyecto contiene las principales clases que se muestra en la tabla 25.

Tabla 25.

Clases del proyecto Gestión Inventario GUI

CLASE	DESCRIPCIÓN
txtCriterioBusq_TextChanged	Función que permite la búsqueda de un producto según el criterio elegido, entre código, nombre, marca, serie, todos.

2.1.3.2 Productos GUI

Este formulario contiene las siguientes acciones: ingreso de un nuevo producto, consulta de productos, eliminación de productos, actualización de un producto, asignación de etiqueta RFID.

- Ingreso de un nuevo producto

En esta sección se realiza el registro de nuevos productos con su respectivo código, nombre, descripción, cantidad, precio, fecha de ingreso, imagen del producto, marca, grupo, código del fabricante, serie, origen y garantía. Como se muestra en la Figura 28

The screenshot shows a web application window titled 'PRODUCTOS'. The main heading is 'INGRESO DE PRODUCTOS'. Below the heading, there is a list of input fields for product information: 'Codigo del Producto', 'Nombre del Producto', 'Descripcion del Producto', 'Cantidad', 'Precio', 'Fecha de Ingreso' (with a dropdown menu showing '09/06/2011'), 'Imagen del Producto', 'Marca', 'Grupo' (with a dropdown menu), 'Codigo del Fabricante', 'Serie', 'Origen', and 'Garantia'. To the right of the 'Imagen del Producto' field is a button labeled 'Examinar'. At the bottom right of the form area is a button labeled 'Almacenar Registros'. The application has a menu bar with options: 'Ingreso', 'Consulta de Productos', 'Eliminacion de Productos', 'Actualizar Producto', and 'Asignar TAG RFID'.

Figura 28. Pantalla de Ingreso de un producto nuevo

- Consulta de productos

En esta sección se realiza la búsqueda de un producto en específico, para lo cual se presenta varias opciones de búsqueda como: Código, Nombre, Marca, Serie, Todos. Como se muestra en la Figura 29.

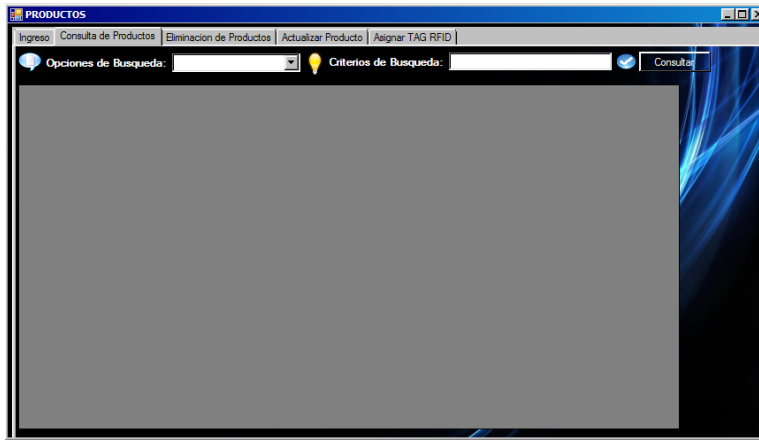


Figura 29. Pantalla de consulta de un producto existente

- Eliminación de productos

Se realiza la búsqueda del producto a eliminar, para lo cual se presenta varias opciones de eliminación que son por: Código, Nombre, Serie. Como se muestra en la Figura 30.

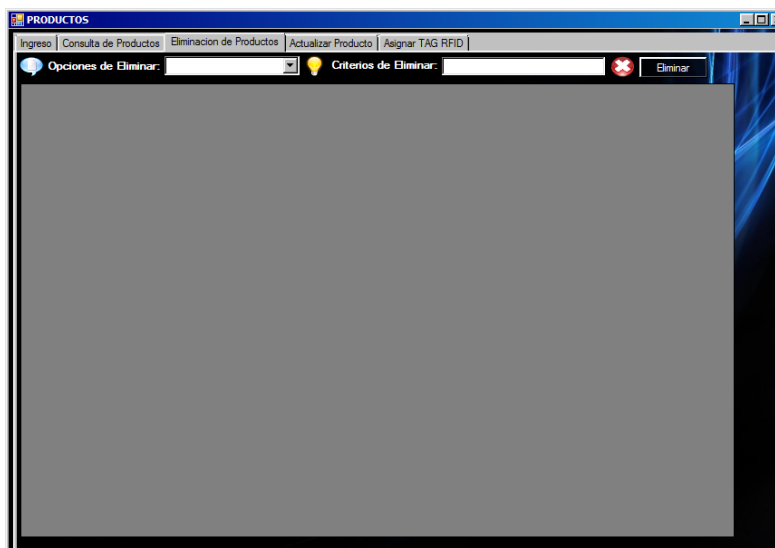


Figura 30. Pantalla de eliminación de un producto

- Actualizar producto

En esta opción se realiza una búsqueda del producto a actualizar, para lo cual se presenta varias opciones de actualización por: Código, Nombre o Serie. Una vez localizado el producto se procede con la actualización de cualquiera de los campos activos: código, nombre, descripción, cantidad, precio, fecha de ingreso, imagen del producto, marca, grupo, código del fabricante, serie, origen y garantía. Como se muestra en la Figura 31.

Figura 31. Pantalla de actualización de productos

- Asignar etiqueta RFID

En esta opción se busca un producto determinado, para lo cual se presenta varias opciones de búsqueda: código, nombre, marca, serie, todos. Una vez localizado el producto se le asocia una etiqueta que viene asignada a cada tarjeta utilizada para la aplicación, con esto queda registrado un código de producto con una etiqueta. Como se muestra en la Figura 32.

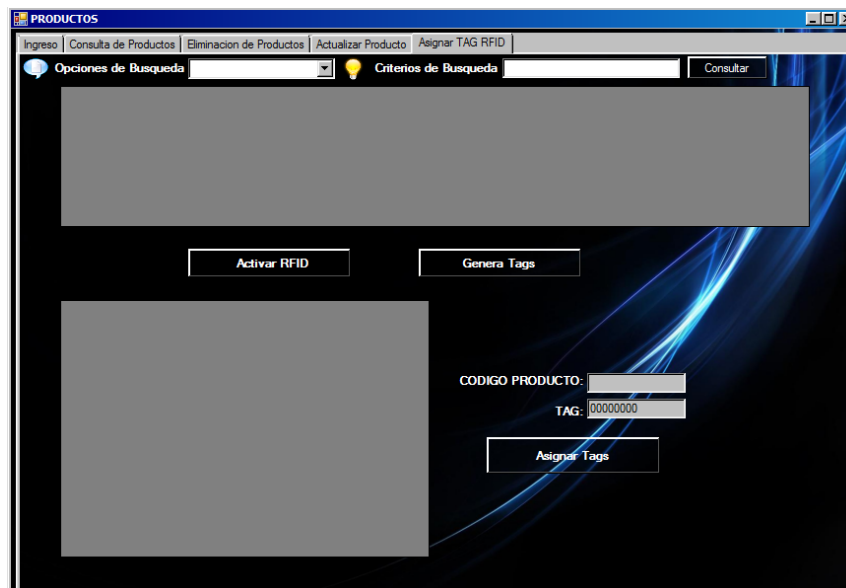


Figura 32. Pantalla de asignación de una etiqueta

Las principales clases que se utilizan para estos formularios están descritas en la tabla 26.

Tabla 26.
Clases del proyecto Productos GUI

CLASE	DESCRIPCIÓN
btnagregarprod_Click	Función que valida el ingreso de todos los campos de un producto nuevo e indica si el registro realizado fue exitoso o no.
btnConsultar_Click	Función que al dar click en el botón Consultar, valida el criterio de búsqueda seleccionado (código, nombre, marca, serie, todos) y muestra el producto requerido.
btnEliminar_Click	Función que al dar click en el botón Eliminar, valida el criterio de búsqueda seleccionado (código, nombre, serie) y elimina el producto requerido.
btnasignatags_Click_1	Función que al dar click en el botón Asignar Tags, valida que los campos CODIGO PRODUCTO y TAG estén bien completados, con el número correcto de dígitos y caracteres determinados.
btnActualizar_Click	Función que al dar click en el botón Actualizar, valida que todos los

	campos estén llenos y guarda la nueva información proporcionada en la base de datos.
btnExaminar3_Click	Función que al dar click en el botón Examinar, valida que se haya seleccionado una de las opción mostradas (Código, Nombre, Serie) y muestra los datos del producto que se tiene almacenado para que se realice el cambio requerido.
btnActivarRFID_Click	
btnGenerarTAG_Click	
btnExaminar_Click	Función que al dar click en el botón Examinar, permite escoger el archivo de tipo imagen que se va a asociar al producto que se está ingresando y se muestra la imagen en el formulario.
btnimagen_Click	
btnAsignarTAG_Click	Función que al dar click en el botón Asignar Tags, valida que se encuentre llenos los campos CODIGO PRODUCTO y TAG y lo almacena en la base de datos.
tmprogressbar_Tick	Función que permite visualizar en forma de barras, el tiempo que demora en almacenar el Tag asignado a un producto.
txtCriterioBusq_TextChanged	Función que valida que se haya seleccionado una de las opciones de criterio de búsqueda de un producto (código, nombre, marca, serie o todos) y muestra los datos del mismo.
dgvAsignacion_CellContentClick	
dgvMostrar_CellContentClick	

2.1.3.3 Usuario GUI

En este formulario se maneja el ingreso, eliminación y la modificación de los usuarios que tiene el sistema. También aquí se asignan los permisos que puede tener el usuario para el ingreso a los diferentes módulos del sistema como: Gestión de Usuarios, Configuración de RFID, Gestión de Productos y Gestión de Inventario. Como se muestran en las Figuras 33 y 34.

Figura 33. Pantalla de Gestión de Usuarios

Figura 34. Pantalla de Asignación de Permisos

Las principales clases que contienen los formularios son descritas en la tabla 27.

Tabla 27.
Clases del proyecto Usuario GUI

CLASE	DESCRIPCIÓN
comboBox1_SelectedIndexChanged	Función que permite escoger la acción que se va a realizar sobre el usuario, crear un nuevo usuario, actualizar o eliminar el usuario.
cmbUsuarios_SelectedIndexChanged	Función que permite seleccionar un usuario de un combo box.
btnAceptar_Click	Función que al dar click en el botón Aceptar, valida que se hayan llenado todos los campos solicitados y la acción a realizar con el usuario. Al final muestra si el registro fue almacenado exitosamente o no.
button1_Click	Función que al dar click en el botón Asignar Permisos, valida que se hayan escogido al menos uno de los cuatro formularios presentados para que el usuario pueda visualizar según los permisos otorgados.

2.2 Desarrollo de la Base de Datos

Para el manejo y organización de gran cantidad de datos en sistemas informáticos es necesario el uso de base de datos, ya que ésta simplifica la interacción con la información y nos permite acceder a los datos de una manera rápida y eficiente.

El diseño adecuado de la base de datos es muy importante ya que nos permite acceder de una forma rápida y eficiente a la información relevante que queremos que se almacene de nuestro prototipo, para que luego sea consultado a través del software de gestión de inventario.

Para el diseño de la base de datos se está tomando en cuenta el modelo conceptual y el modelo lógico.

2.2.1 Modelo Conceptual

En este modelo se establece la estructura que va a tener la base de datos de una manera genérica, es decir independiente del gestor de base de datos que se vaya a utilizar, se la representa en base a un modelo de datos de alto nivel, por lo cual para el desarrollo de la base de datos se utilizó el modelo entidad-relación.

Las principales entidades según los requerimientos del prototipo son:

- Usuario: la cual tendrá información personal del usuario, información de inicio de sesión y su rol (cargo) de usuario.
- Producto: la cual tendrá un código único que lo identifique, descripción general del producto.
- Permisos de Usuario: tendrá información acerca del usuario y según su cargo establecerá sus módulos de gestión en base a eso.
- Tags: Esta entidad poseerá un identificador único, indicará a que producto está asociado y respectivo código del TAG RFID.
- Total productos: Esta entidad se la utiliza para almacenar el total de productos con un respectivo identificador para realizar cálculos sobre los totales en la aplicación.
- Seguridad Producto: Esta entidad contiene información del producto y si se encuentra asignado para dar seguimiento y seguridad a cada producto.
- Inicio de sesión: En el inicio de sesión se almacena el usuario que ingresa y su cargo en función de los permisos de usuario que posea.
- Productos Fuera: Es un registro donde se almacenan todos los productos que salen de bodega para tener constancia y control de ello.

Todas estas entidades se ven reflejadas en la Figura 35.

Utilizando el modelo entidad-relación se identificaron y establecieron las

principales entidades y atributos necesarios para el almacenamiento de la información del prototipo que son:

- **Total productos**(ID, Cod_Pro, Cantidad)
- **Seguridad Producto**(IdSeg, Cod_Pro, IdEstado, Tag)
- **Tags**(Num, Cod_Pro, Tag)
- **Producto**(Código, Nombre, Descripción, Cantidad, Precio, Fecha_de_Ingreso, Imagen, Marca, Grupo, Código_Fabrica, Serie, Origen, Garantía, IdEstado)
- **Usuario**(CI, Apellidosyombres, NombreUsuario, Dirección, Teléfono, Contraseña, Pregunta_Seguridad, Respuesta_Seguridad, Cargo)
- **Permiso Usuario**(CI, GestionProductos, GestiónUsuarios, GestiónRFID, GestiónInventario)
- **Inicio_Sesión** (ID, Nombre de Usuario, Contraseña, Cargo)
- **Productos Fuera** (Código, Nombre, Descripción, Precio, Fecha de Salida, Imagen, Marca, Grupo, código Fabrica, Modelo, Origen, Garantía, IdEstado)

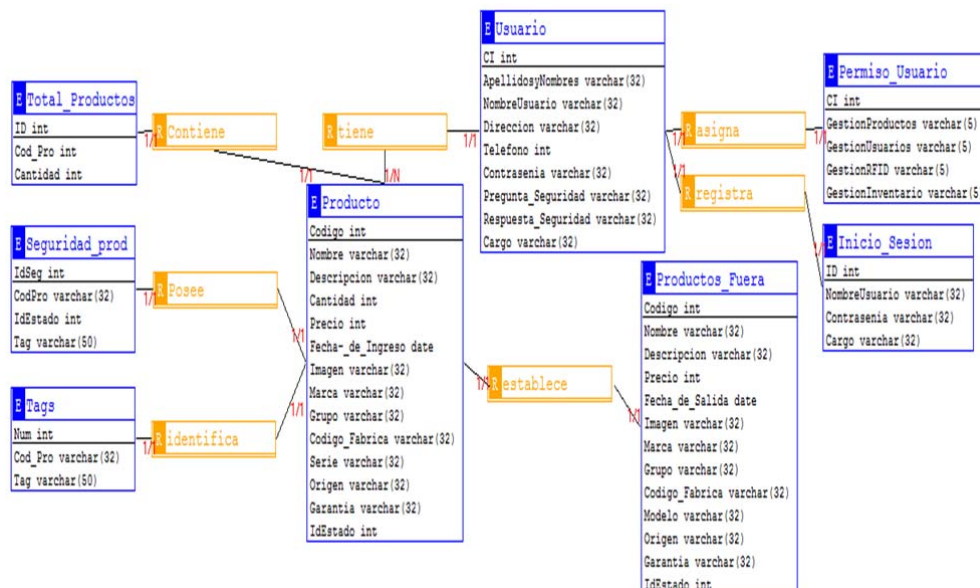


Figura 35. Diagrama entidad-relación.

2.2.2 Modelo Lógico

Este modelo parte del modelo conceptual en el que ya se establece claramente la estructura de los datos a almacenarse y se lo adapta al sistema gestor de base de datos elegido en este caso el desarrollo se lo lleva a cabo en SQL server 2008. Como se muestra en el diagrama de tablas de la Figura 36.

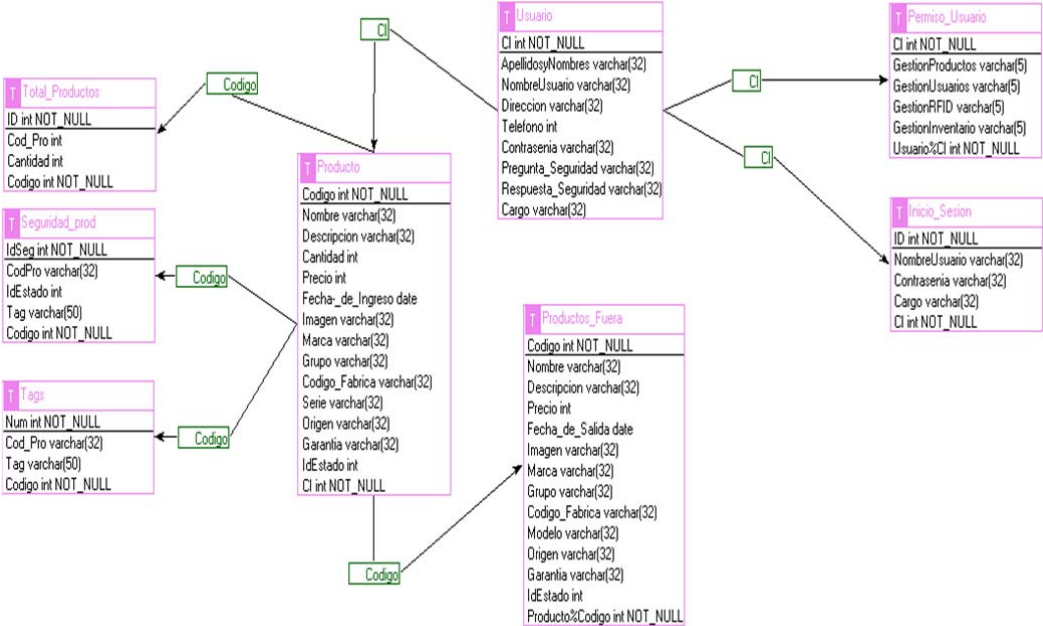


Figura 36. Diagrama de tablas

2.3 Integración del Sistema

Este sistema prototipo tiene tres puntos de integración el primero que es hacia la base de datos la cual se integra a la aplicación mediante la clase conexión que da la posibilidad de abrir o cerrar el flujo de información, el segundo punto es la integración con el lector RFID el cual se enlaza a la aplicación en la capa de datos y de negocio, en las cuales se agregan las librerías propias del lector para poder vincularse correctamente y configurar sus parámetros de conexión y lectura de tags, finalmente el tercer punto es la integración con el usuario a través de la interfaz gráfica fácil e intuitiva de usar, como se muestra en la figura 37.

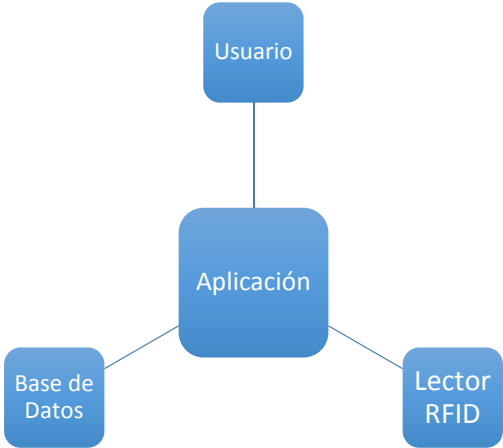


Figura 37. Integración del sistema

El proceso general de funcionamiento del prototipo de gestión de inventario utilizando la tecnología RFID comienza desde que el usuario inicia sesión ya que el sistema valida según sus credenciales si es un usuario administrador, operario o visor y según ello habilita o deshabilita módulos del sistema. Si es un usuario administrador tiene la posibilidad de ingresar a los cuatro principales módulos que posee el sistema que son: Parámetros RFID, Gestión de usuarios, Gestión de productos y Gestión de Inventario. Como se muestra en la Figura 38.

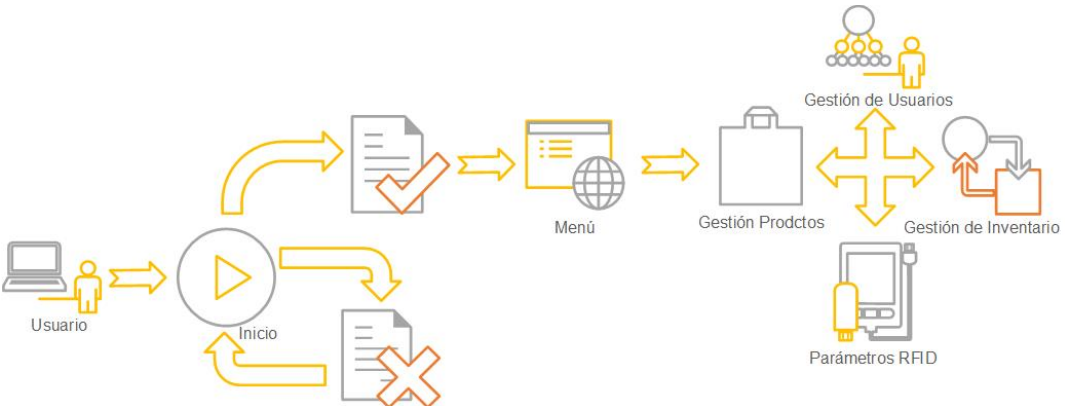


Figura 38. Diagrama General del Prototipo

En el módulo parámetros RFID el usuario podrá configurar la comunicación entre el lector RFID y el sistema, es decir se va a establecer las frecuencias de

transmisión del lector, potencia, intervalos de lectura entre otras características propias del lector.

En Gestión de usuarios podrá crear, actualizar, eliminar, consultar y cambiar de permisos a los usuarios que considere pertinentes para que puedan acceder a los diferentes módulos y configuración del sistema, además que toda esta información será almacenada en la base de datos.

En el módulo Gestión de productos podrá realizar el ingreso, actualizar, eliminar, consultar y asignar tags al producto para que pueda ser monitoreado con el lector RFID y la información detallada de los productos será almacenada en la base de datos.

En la gestión de inventario es donde se llevará a cabo el monitoreo de productos, así como los ingresos y egresos de los mismos en tiempo real para poder contabilizarlos e inventariarlos de una manera autónoma, esta información se verá reflejada en el sistema y almacenada en la base de datos.

Si el usuario es de tipo operario los módulos habilitados para ese perfil son: Parámetros RFID, Gestión de productos y gestión de inventario. Si el usuario es de tipo visor solo tendrá acceso al módulo de gestión de inventario pudiendo realizar las actividades descritas anteriormente. Todo este proceso se muestra en el diagrama de flujo de la figura 39.

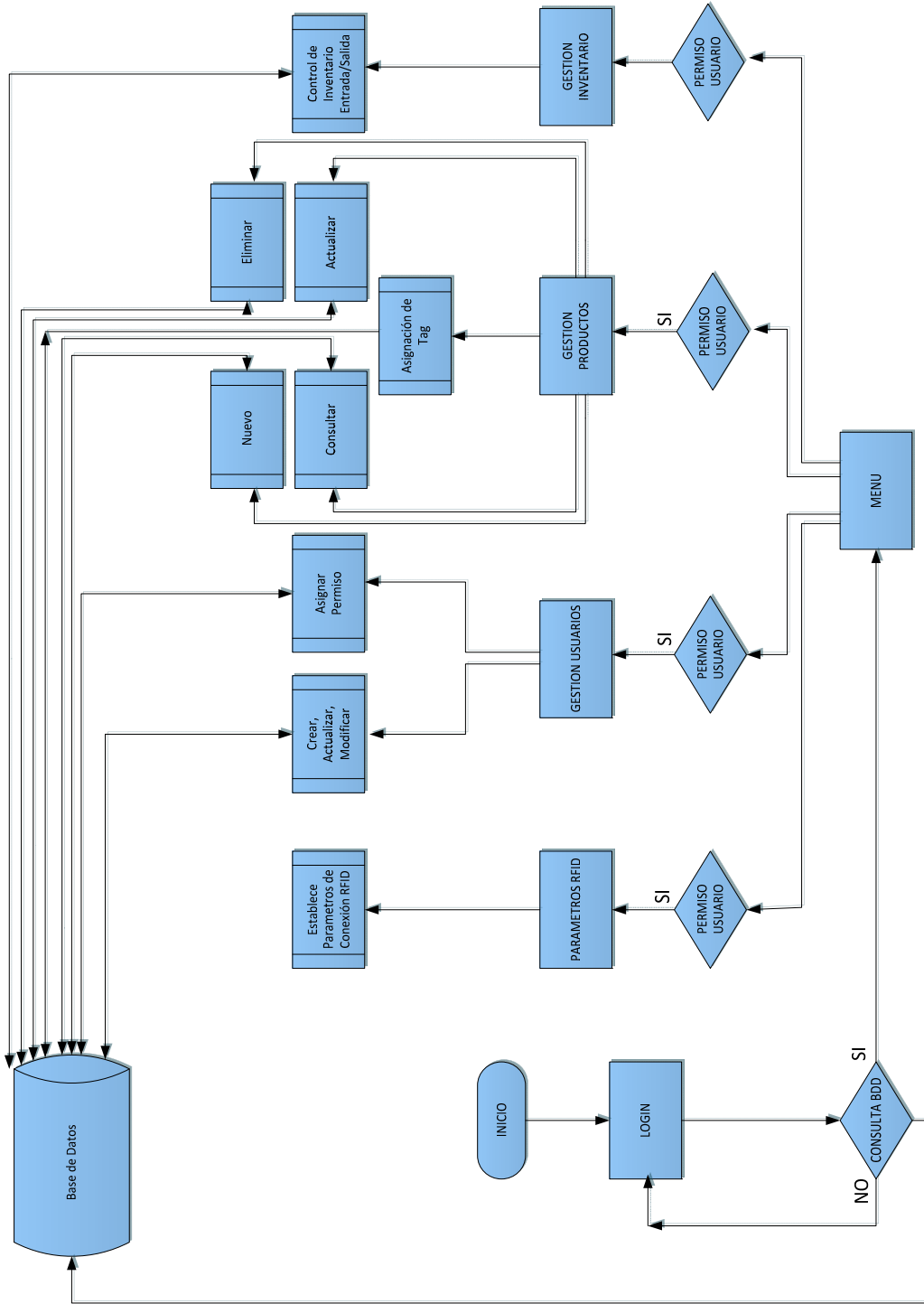


Figura 39. Desarrollo del sistema

El prototipo a nivel físico y electrónico se integra desde cuando el usuario inicia sesión y el computador establece conexión con el lector RFID a través de un cable serial RS-232, el lector por las peticiones del sistema (software prototipo) establece sus parámetros internamente y comienza a escanear TAG's a través de su antena emitiendo ondas de radio frecuencia UHF. Los TAG's internamente al recibir la señal emitida por el lector se energizan y establecen comunicación, su primera línea de transmisión es para informar sus parámetros de seguridad así como también de encriptación, si se autentican favorablemente ya transmiten información relevante de los productos asociados a ese TAG. Este proceso se ilustra en la Figura 40.

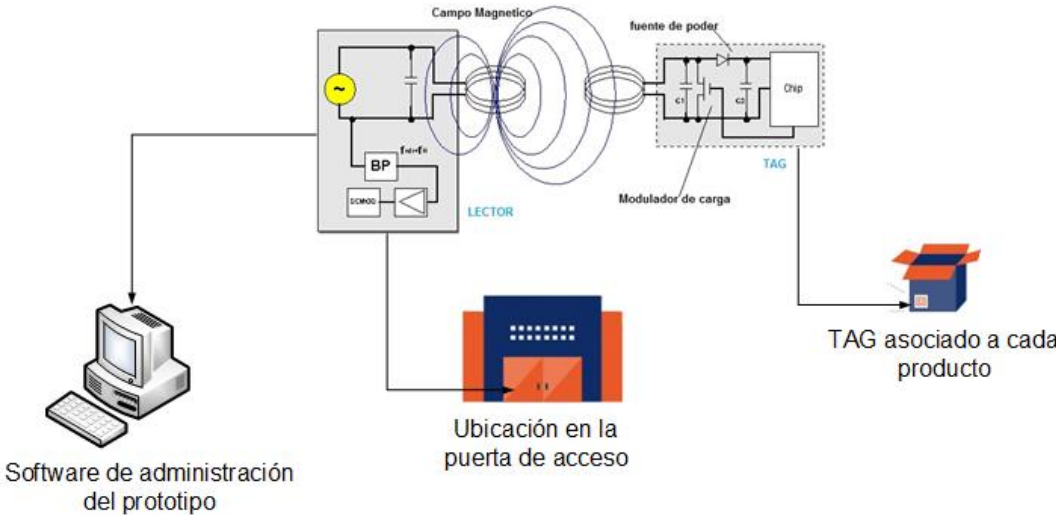


Figura 40. Diagrama del Prototipo a nivel Físico y Electrónico

Una vez que la transmisión de datos es exitosa esta información es procesada por el lector el cuál retransmite esta información al software prototipo donde es validada y procesada para llevar a cabo la administración de inventario.

3. CAPÍTULO III: PRUEBAS DE FUNCIONAMIENTO

En este capítulo se presentan las pruebas realizadas para verificar el correcto funcionamiento del prototipo, pruebas de verificación de datos, transacciones y usuarios autorizados.

Las pruebas se realizaron de una manera organizada, se las separaron en función de los módulos que posee el prototipo que son: inicio de sesión, gestión RFID, gestión de usuarios, gestión de productos y gestión de inventario.

3.1 Pruebas de Inicio de Sesión

Estas pruebas se realizaron para validar a los usuarios que tienen acceso al sistema y sus privilegios dentro de la aplicación.

3.1.1 Ingreso de un Usuario que no esté Registrado

Para realizar esta prueba primero se revisaron los usuarios que existen directamente en la base de datos, como se muestra en la figura 41.

	CI	ApellidosyNombres	NombreUsuario	Direccion	Telefono	Contraseria	Pregunta_Seguridad	Respuesta_Seguridad	Cargo
1	603598756	Ortega Catherine	caortega	Todedo y Madrid	2550632	*****	Auto Favortó?	Audi	Administrador
2	604761577	Novillo Danny	dnovillo	Toledo y Madrid	2550632	12345	Numeros primeros	cinco primeros numeros	Administrador
3	1111111111	Usuario1	User1	usuario	9999999	4321	Numeros inversos	Cuatro inversos	Minitoreo
4	1111111112	qqqqqqq	sssssssss	qqqqqq	1111111	1232	aaaaaaaaaaaa	aaaaaaaaaaaa	Administrador

Figura 41. Consulta usuarios existentes

Según esta información se intentó ingresar al sistema con un nombre de usuario que no estaba registrado para validar el usuario, como se muestra en la figura 42.



Figura 42. Inicio de sesión con un usuario no registrado

El sistema indica con un mensaje que el usuario o su contraseña son incorrectos ya que éste no se encuentra registrado, como se indica en la figura 43.

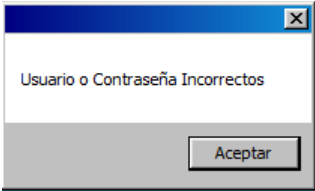


Figura 43. Error de inicio de sesión

3.1.2 Validación de la Contraseña de un Usuario Registrado.

En el caso de que el usuario este registrado pero su contraseña este mal digitada en el inicio de sesión, el sistema muestra el mensaje Usuario o Contraseña Incorrectos, como se puede ver en la figura 43 y si las credenciales son correctas muestra el inicio de sesión exitoso con un mensaje de bienvenida, como se muestra en la figura 45.

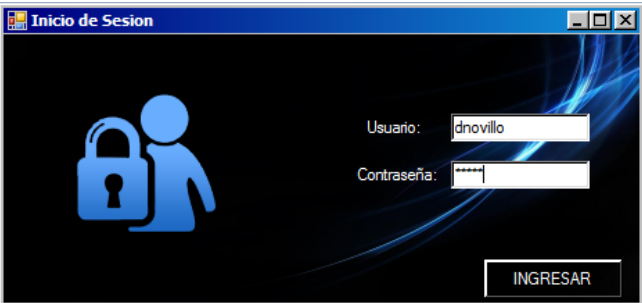


Figura 44 Inicio de sesión con un contraseña errónea

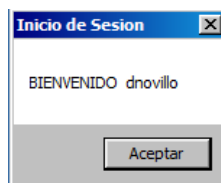


Figura 45 Mensaje de bienvenida

3.1.3 Validación del Nivel de Usuario

Esta validación se realizó en base a los privilegios asignados a cada usuario (administrador, operario o monitoreo) y su posibilidad de acceder a los diferentes módulos del sistema. Para la ejecución de esta prueba primero se crearon varios usuarios con niveles de acceso diferentes e ingresaremos como administrador el cual permite ingresar a todos los módulos del sistema, como se muestra en la figura 46.



Figura 46 Menú para usuario administrador

Ingresamos al sistema con un usuario con privilegios de operario el cual solo permite el ingreso a parámetros RFID, Productos y Gestión de Inventario, como se muestra en la figura 47.



Figura 47 Menú para usuario operador

Para un usuario monitor solo le permite el ingreso a Parámetros RFID y Gestión de productos, como se muestra en la figura 48.

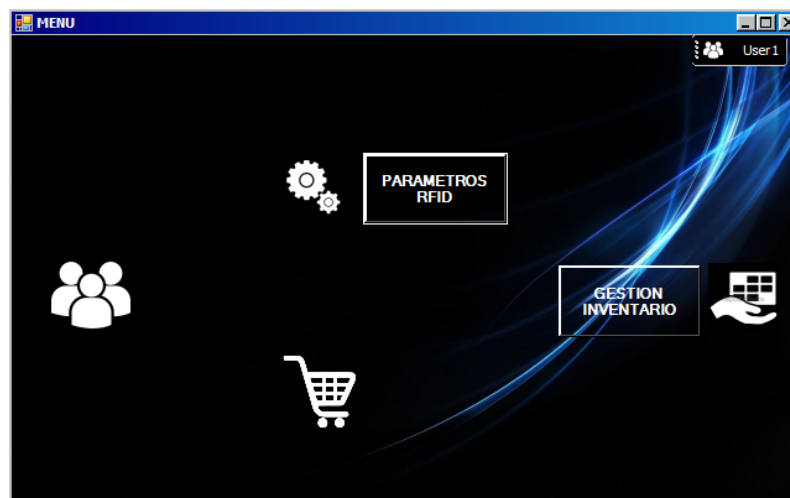


Figura 48. Menú para usuario monitor

Para validar los tres tipos de usuarios ingresados se revisó sus permisos directamente desde una consulta a la base de datos, como se muestra en la figura 49.

	CI	ApellidosNombres	NombreUsuario	Direccion	Telefono	Contrasenia	Pregunta_Seguridad	Respuesta_Seguridad	Cargo
1	603598756	Ortega Catherine	caortega	Todedo y Madrid	2550632	98765	Auto Favorito?	Audi	Operador
2	604761577	Novillo Danny	dnovillo	Toledo y Madrid	2550632	12345	Numeros primeros	cinco primeros numeros	Administrador
3	1111111111	Usuario1	User1	usuario	9999999	4321	Numeros inversos	Cuatro inversos	Monitoreo
4	1111111112	qqqqqqq	sssssssss	qqqqqq	1111111	1232	aaaaaaaaaaaa	aaaaaaaaaaaa	Administrador
5	1732497387	Cardenas Juan	jcardenas	Quito	2342312	cardenas	cual es su apellido?	Cardenas	Operador

Figura 49 Consulta de permisos de usuarios

3.2 Pruebas de Gestión RFID

Las pruebas de este módulo del prototipo consistieron en validar la conexión entre el lector RFID y la aplicación, también la veracidad con la que el lector identifica los tags.

3.2.1 Conexión al Lector RFID

Para verificar que el lector RFID se conecte exitosamente con el sistema se realizó una prueba desde la ventana Parámetros RFID, como se muestra en la figura 50, donde se seleccionó el puerto COM (puerto serial al que se conecta físicamente el lector con la PC) o se lo deja para que lo busque automáticamente el sistema.

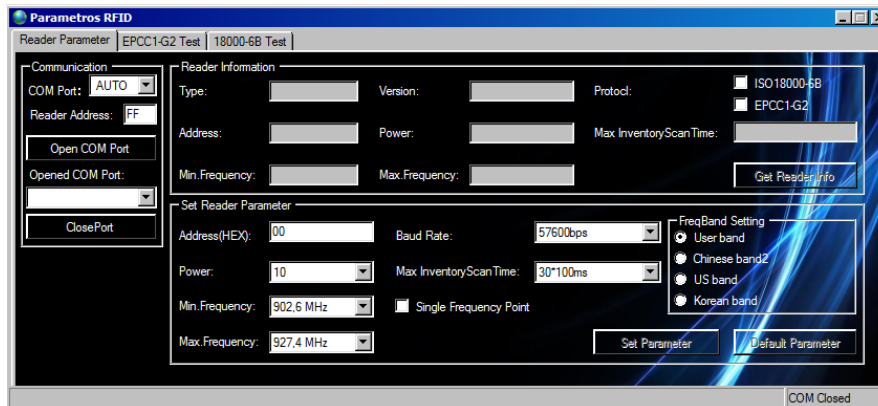


Figura 50. Ventana parámetros RFID

Una vez seleccionado el puerto COM correcto el lector se comunica con la aplicación y se puede ver las principales características y parámetros del lector en el formulario, como se muestra en la figura 51, lo cual indica que su conexión fue exitosa.

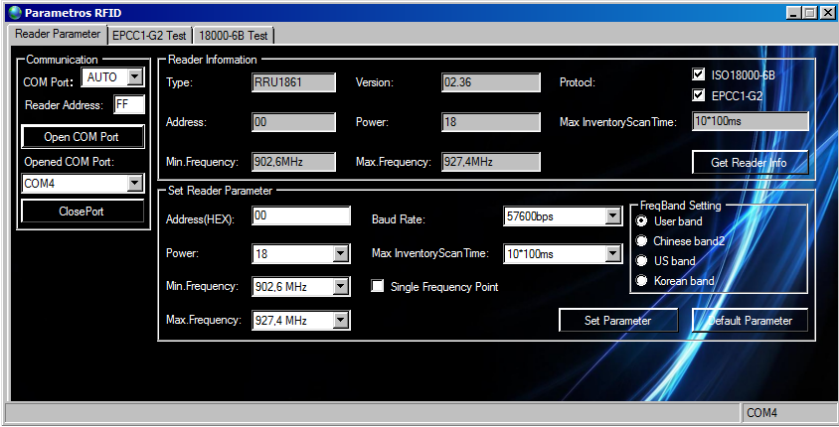


Figura 51 Características y parámetros del lector RFID

3.2.2 Lectura de TAG's

Ya establecida una conexión exitosa entre el lector y la aplicación se llevó a cabo la prueba de lectura de tags, para lo cual se seleccionó la pestaña EPCC1-G2 Test dentro del formulario Parámetros RFID y una vez seleccionado el intervalo de lectura el dispositivo se queda a la espera de que los tags se acerquen a su radio de cobertura para poder ser leídos, el lector los reconoce e identifica, como se muestra en la figura 52. Comprobando de esta manera la correcta lectura de tags.

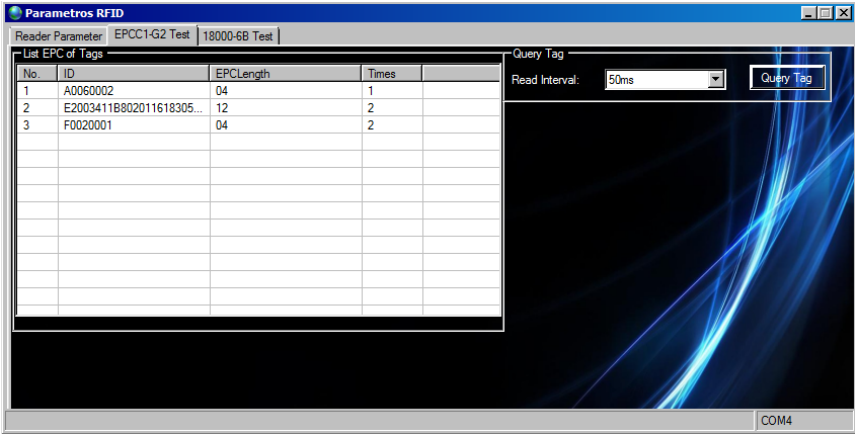


Figura 52. Lectura de Tags

3.3 Pruebas de Gestión de Usuarios

En las pruebas de gestión de usuarios se validó la creación, actualización, eliminación de usuarios del sistema, así como los diferentes permisos que puede tener dicho usuario para realizar cambios en la aplicación.

3.3.1 Creación de Usuarios

Aquí se realizaron las pruebas de creación de un nuevo usuario y validación si el usuario ya existe en el sistema. Para lo cual primero se realizó una consulta a la base de datos para revisar los usuarios existentes, como se muestra en la figura 53.

	CI	Apellidos y Nombres	Nombre Usuario	Dirección	Teléfono	Contraseña	Pregunta_Seguridad	Respuesta_Seguridad	Cargo
1	603598756	Ortega Catherine	caortega	Toledo y Madrid	2550632	98765	Auto Favorito?	Audi	Operador
2	604761577	Novillo Danny	dnovillo	Toledo y Madrid	2550632	12345	Numeros primeros	cinco primeros numeros	Administrador
3	1111111111	Usuario1	User1	usuario	9999999	4321	Numeros inversos	Cuatro inversos	Monitoreo
4	1111111112	qqqqqqq	sssssssss	qqqqqq	1111111	1232	aaaaaaaaaaaa	aaaaaaaaaaaa	Administrador

Figura 53 Consulta usuarios existentes

Una vez que se revisó que el usuario no exista, se creó un nuevo usuario desde la ventana de gestión de usuarios, como podemos ver en la figura 54.

The screenshot shows a window titled 'USUARIOS' with a sub-tab 'Gestion de Usuarios'. The main heading is 'GESTION DE USUARIOS'. On the left, there is an icon of three people. The form contains the following fields:

- N° Cédula: 17324973872
- Escoga una Opcion: Nuevo (dropdown menu)
- Apellidos y Nombres: Cardenas Juan
- Direccion: Quito
- Telefono: 2342312
- Nombre de Usuario: jcardenas
- Contraseña: jcardenas
- Pregunta de Seguridad: ¿cual es su apellido?
- Respuesta de Seguridad: Cardenas

A 'Crear' button is located at the bottom right of the form.

Figura 54 Gestión de usuarios

El sistema mediante una ventana indicó que el registro fue creado exitosamente, como se muestra en la figura 55.

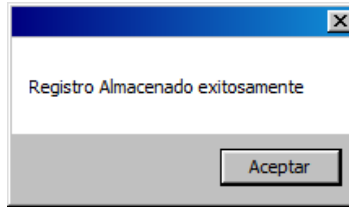


Figura 55. Almacenado exitoso

Para verificar que el usuario fue creado se revisó desde la base de datos, como se muestra en la figura 56.

	CI	ApellidosyNombres	NombreUsuario	Direccion	Telefono	Contrasenia	Pregunta_Seguridad	Respuesta_Seguridad	Cargo
1	603598756	Ortega Catherine	caortega	Todedo y Madrid	2550632	98765	Auto Favorito?	Audi	Operador
2	604761577	Novillo Danny	dnovillo	Toledo y Madrid	2550632	12345	Numeros primeros	cinco primeros numeros	Administrador
3	1111111111	Usuario1	User1	usuario	9999999	4321	Numeros inversos	Cuatro inversos	Monitoreo
4	1111111112	qqqqqq	ssssssss	qqqqqq	1111111	1232	aaaaaaaaaaaa	aaaaaaaaaaaa	Administrador
5	1732497387	Cardenas Juan	jcardenas	Quito	2342312	cardenas	cual es su apellido?	Cardenas	Operador

Figura 56. Consulta usuarios existentes

3.3.2 Actualización de Usuarios

Se llevó a cabo la prueba de actualizar un usuario en forma general es decir sus datos personales y sus privilegios de acceso, en este caso tomaremos como ejemplo el usuario Juan Cárdenas que fue ingresado en la sección anterior para realizar las pruebas, se ingresó al formulario gestión de productos y se digitó la CI (cédula de identidad) del usuario a ser actualizado, se desplegó la información actual del usuario, como se muestra en la figura 57 y se pudo confirmar haciendo una consulta a la base de datos como se muestra en la figura 56 de la sección anterior.



The screenshot shows a window titled 'USUARIOS' with a sub-header 'Gestion de Usuarios | Permisos'. The main title is 'GESTION DE USUARIOS'. On the left, there is an icon of three people. The form contains the following fields:

Nº Cédula :	1732497387
Escoga una Opcion:	Actualizar
Apellidos y Nombres :	Cardenas Juan
Direccion :	Quito
Telefono :	2342312
Nombre de Usuario :	jcardenas
Contraseña :	cardenas
Pregunta de Seguridad :	cual es su apellido?
Respuesta de Seguridad :	Cardenas

An 'Actualizar' button is located at the bottom right of the form.

Figura 57. Ventana de actualización de usuarios

La información que se actualizó del usuario fue su dirección y número telefónico como se muestra en la figura 58.



The screenshot shows the same 'GESTION DE USUARIOS' window, but with updated information in the 'Direccion' and 'Telefono' fields:

Nº Cédula :	1732497387
Escoga una Opcion:	Actualizar
Apellidos y Nombres :	Cardenas Juan
Direccion :	Av. Amazonas y Naciones Unidas
Telefono :	2412248
Nombre de Usuario :	jcardenas
Contraseña :	cardenas
Pregunta de Seguridad :	cual es su apellido?
Respuesta de Seguridad :	Cardenas

The 'Actualizar' button remains at the bottom right.

Figura 58. Información actualizada

El sistema indicó que los datos fueron actualizados correctamente como se muestra en la figura 59.

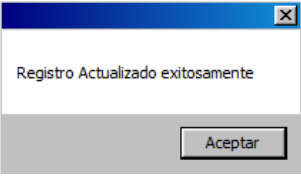


Figura 59 Mensaje de actualización exitosa

Para comprobar que la información del usuario fue actualizada correctamente se revisó en la base de datos listando la información de todos los usuarios, como se muestra en la figura 60 y se pudo apreciar que la dirección y teléfono del usuario Juan Cárdenas fue modificado.

	CI	Apellidos y Nombres	Nombre Usuario	Dirección	Teléfono	Contraseña	Pregunta_Seguridad	Respuesta_Seguridad	Cargo
1	603598756	Ortega Catherine	caortega	Todedo y Madrid	2550632	98765	Auto Favorito?	Audi	Operador
2	604761577	Novillo Danny	dnovillo	Toledo y Madrid	2550632	12345	Numeros primeros	cinco primeros numeros	Administrador
3	1111111111	Usuario1	User1	usuario	9999999	4321	Numeros inversos	Cuatro inversos	Monitoreo
4	1111111112	qqqqqqq	sssssssss	qqqqqq	1111111	1232	aaaaaaaaaaaa	aaaaaaaaaaaa	Administrador
5	1732497387	Cardenas Juan	jcardenas	Av. Amazonas y Naciones Unidas	2412248	cardenas	cual es su apellido?	Cardenas	Sin Asignar

Figura 60. Consulta información actualizada

3.3.3 Eliminación de Usuarios

Para comprobar que un usuario se ha eliminado completamente del sistema se realizó una prueba desde la base de datos revisando los usuarios actualmente activos como se muestra en la figura 60 de la sección anterior, desde la ventana de gestión de usuarios se digitó la CI ya que este identifica a cada usuario y en el combo box se seleccionó eliminar, aceptando de esta manera que se eliminará dicho usuario del sistema, como se muestra en la figura 61.



Figura 61. Ventana eliminación de usuarios

El sistema indica que el usuario fue eliminado correctamente como se muestra en la figura 62.

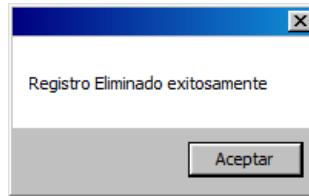


Figura 62. Mensaje de eliminación exitosa

Se puede comprobar que el usuario fue eliminado del sistema revisándolo en la base de datos, como se muestra en la figura 63.

	CI	ApellidosyNombres	Nombre Usuario	Direccion	Telefono	Contraseña	Pregunta_Seguridad	Respuesta_Seguridad	Cargo
1	603598756	Ortega Catherine	caortega	Todedo y Madrid	2550632	98765	Auto Favorito?	Audi	Operador
2	604761577	Novillo Danny	dnovillo	Toledo y Madrid	2550632	12345	Numeros primeros	cinco primeros numeros	Administrador
3	11111111111	Usuario1	User1	usuario	9999999	4321	Numeros inversos	Cuatro inversos	Monitoreo
4	1732497387	Cardenas Juan	jcardenas	Av. Amazonas y Naciones Unidas	2412248	cardenas	cual es su apellido?	Cardenas	Sin Asignar

Figura 63. Consulta usuarios

3.3.4 Asignación de Permisos a Usuarios.

Cuando se crea un usuario en el sistema este por defecto no tiene ningún tipo de permiso y se almacena automáticamente sin asignar, el administrador una vez creado el usuario tiene que asignarle permisos según el nivel de acceso al que pertenezca que puede ser administrador, operario o monitoreo, como se muestra en la figura 64, se utilizó en este caso al usuario Juan Cárdenas que fue ingresado en la sección anterior.

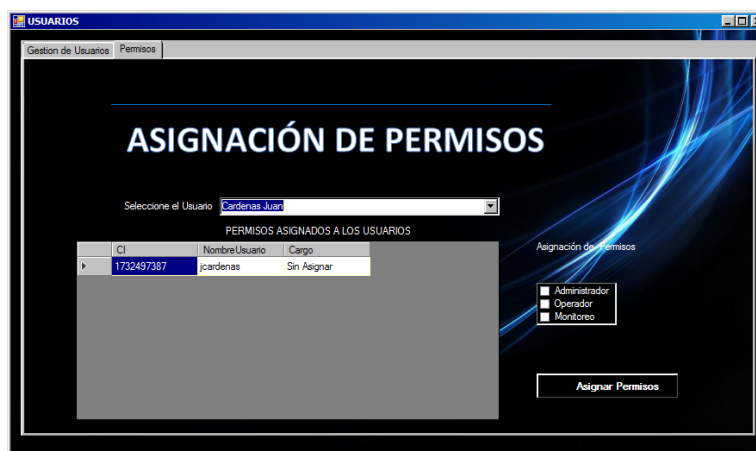


Figura 64. Ventana de asignación de permisos

A este usuario se le asignó permisos de operario, el sistema mostró que fue asignado exitosamente los permisos, como se muestra en la figura 65.

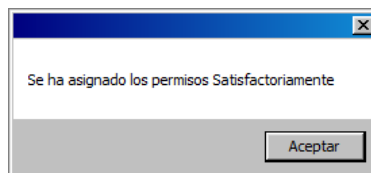


Figura 65. Mensaje exitoso de asignación de permisos

Para comprobar que fueron asignados los permisos se revisó este usuario en la base datos, como se muestra en la figura 66, donde se visualizó en el campo Cargo su nivel de acceso al sistema.

	CI	Apellidos y Nombres	Nombre Usuario	Dirección	Teléfono	Contraseña	Pregunta_Seguridad	Respuesta_Seguridad	Cargo
1	603598756	Ortega Catherine	caortega	Todedo y Madrid	2550632	98765	Auto Favorto?	Audi	Operador
2	604761577	Novillo Danny	dnovillo	Toledo y Madrid	2550632	12345	Numeros primeros	cinco primeros numeros	Administrador
3	1111111111	Usuario1	User1	usuario	9999999	4321	Numeros inversos	Cuatro inversos	Monitoreo
4	1732497387	Cardenas Juan	jcardenas	Av. Amazonas y Naciones Unidas	2412248	cardenas	cual es su apellido?	Cardenas	Operador

Figura 66. Consulta de permisos de usuarios

3.4 Pruebas de Gestión de Productos

En este módulo del sistema se realizaron validaciones del ingreso, actualización, eliminación, consulta y asignación de Tags a productos para verificar su buen funcionamiento.

3.4.1 Ingreso de Productos

Para validar el ingreso de productos se lo hizo directamente en el módulo gestión de productos en la pestaña ingreso, donde se colocó todos los datos del nuevo producto a ingresar, como se muestra en la figura 67.

Figura 67. Ventana ingreso de productos

Después de completar los datos del formulario al guardarlo el sistema indica que el registro fue almacenado exitosamente, como se muestra en la figura 68.

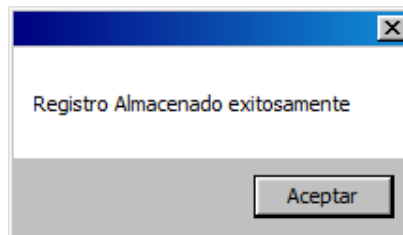


Figura 68. Mensaje de ingreso exitoso

Para comprobar que el producto fue creado se lo revisó en la base de datos listando todos los productos, como se muestra en la figura 69.

	Codigo	Nombre	Descripcion	Precio	Fecha_de_Ingreso	Imagen	Marca
1	1	LAPTOP	4GB RAM, CORE I3, 500GB HDD	500	2016-06-30 19:00:00.000	C:\Users\Danny\Pictures\DELL.png	DELL
2	2	as	asdas	200	2016-06-30 19:00:00.000	C:\Users\Danny\Pictures\DELL.png	asd
3	3	ASDASD	ASD	1233	2016-06-30 19:00:00.000	C:\Users\Danny\Pictures\DELL.png	ASDASD
4	20	CZXVZC	BNZNB	20	2016-06-30 19:00:00.000	C:\Users\Danny\Pictures\DELL.png	GAWDA
5	100	PC HP	Laptop Pavilion	1000	2016-11-27 19:00:00.000	C:\Users\Danny\Pictures\hvp-pavilion-p6.jpg	HP
6	101	PC HP	LAPTOP PAVILION	1000	2016-11-27 19:00:00.000	C:\Users\Danny\Pictures\hvp-pavilion-p6.jpg	HP
7	102	iPAD	IPAD AIR 2	500	2016-11-27 19:00:00.000	C:\Users\Danny\Pictures\identify-ipad-pro.jpg	APPLE
8	103	iPAD	IPAD AIR 2	500	2016-11-27 19:00:00.000	C:\Users\Danny\Pictures\identify-ipad-pro.jpg	APPLE
9	104	SPEAKER	SPEAKER 15W	100	2016-11-27 19:00:00.000	C:\Users\Danny\Pictures\1380322449.jpg	REALTEK
10	105	Impresora HP	ipmresora Laser Jet	400	2016-11-29 19:00:00.000	C:\Users\Danny\Pictures\616969102e5c42d4a40e430...	HP

Figura 69. Verificación de productos existentes

3.4.2 Actualización de Productos

En la gestión de productos al ingresar en la ventana actualizar permite modificar información de un producto ingresando su código, nombre o número de serie que identifique dicho producto, primero para poder saber los productos que se encuentran registrados se realizó una consulta en la base de datos como se muestra en la figura 69 de la sección anterior. Se realizaron las modificaciones necesarias en la información del producto en este caso se cambió la descripción de producto, precio y fecha de ingreso, como se muestra en la figura 70



Figura 70 Ventana de actualización de productos

Una vez finalizado, el sistema indica que el registro fue almacenado exitosamente, como se muestra en la figura 71.

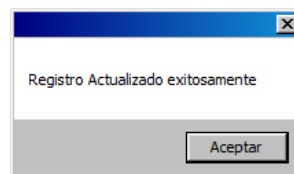


Figura 71. Mensaje de registro actualizado exitosamente

Se realizó un listado de los productos en la base de datos para verificar que la información fue modificada, como se muestra en la figura 72.

	Codigo	Nombre	Descripcion	Precio	Fecha_de_ingreso	Imagen	Marca	Grupo
1	1	LAPTOP	4GB RAM, CORE I3, 500GB HDD	500	2016-06-30 19:00:00.000	C:\Users\Danny\Pictures\DELL.png	DELL	Electronica
2	2	as	asdas	200	2016-06-30 19:00:00.000	C:\Users\Danny\Pictures\DELL.png	asd	Electronica
3	3	ASDASD	ASD	1233	2016-06-30 19:00:00.000	C:\Users\Danny\Pictures\DELL.png	ASDASD	Electronica
4	20	CZXVZC	BNZNB	20	2016-06-30 19:00:00.000	C:\Users\Danny\Pictures\DELL.png	GAWDA	Electronica
5	100	PC HP	Laptop Pavilion	1000	2016-11-27 19:00:00.000	C:\Users\Danny\Pictures\hp-pavilion-p6.jpg	HP	Electronica
6	101	PC HP	LAPTOP PAVILION	1000	2016-11-27 19:00:00.000	C:\Users\Danny\Pictures\hp-pavilion-p6.jpg	HP	Electronica
7	102	iPAD	IPAD AIR 2	500	2016-11-27 19:00:00.000	C:\Users\Danny\Pictures\identify-ipad-pro.jpg	APPLE	Electronica
8	103	iPAD	IPAD AIR 2	500	2016-11-27 19:00:00.000	C:\Users\Danny\Pictures\identify-ipad-pro.jpg	APPLE	Electronica
9	104	SPEAKER	SPEAKER 15W	100	2016-11-27 19:00:00.000	C:\Users\Danny\Pictures\1380322449.jpg	REALTEK	Accesorios
10	105	Impresora HP	ipmresora Multifuncion	700	2016-11-30 19:00:00.000	C:\Users\Danny\Pictures\e616969102e5c42d4a40e430...	HP	Electronica

Figura 72 Verificación de actualización de productos

3.4.3 Eliminación de Productos

Para eliminar un producto del sistema se lo realizó desde el módulo de gestión de productos en la sección eliminar en la cual se colocó el nombre, código o serie del producto a ser eliminado. Primero se verificaron los productos existentes en el sistema desde la base de datos como se muestra en la figura 3.32 de la sección anterior, en este caso se eliminó el producto de código 20, como se muestra en la figura 73.

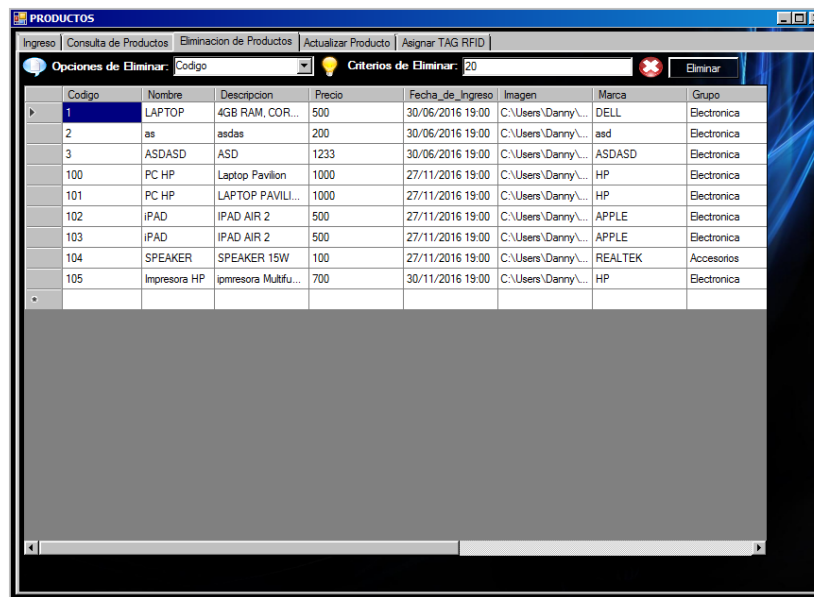


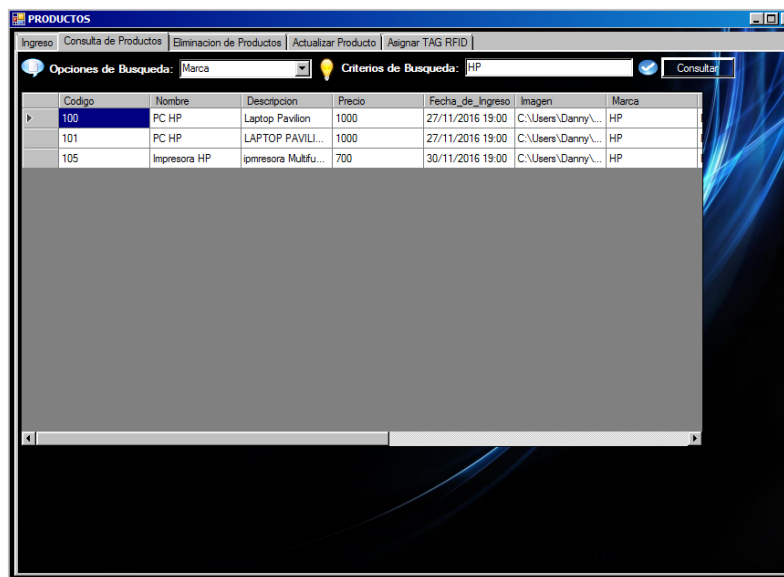
Figura 73. Ventana de eliminación de productos

El sistema al eliminar el producto hace una petición a la base de datos donde muestra en pantalla el listado de los productos actuales en el que ya no consta el producto eliminado, como se muestra en la figura 73.

3.4.4 Consulta de Productos

En esta ventana el usuario podrá realizar consultas de productos a la base de datos directamente desde la aplicación, para lo cual se selecciona una opción de búsqueda que puede ser código, nombre, marca, serie o todos los productos existentes. Se establece el criterio de búsqueda en función de la opción escogida, como se muestra en la figura 74.

El sistema despliega en pantalla la información solicitada por el usuario, de forma similar se realiza con las demás opciones de búsqueda.



The screenshot shows a software window titled 'PRODUCTOS' with a menu bar containing 'Ingreso', 'Consulta de Productos', 'Eliminación de Productos', 'Actualizar Producto', and 'Asignar TAG RFID'. Below the menu bar, there are search controls: 'Opciones de Búsqueda:' with a dropdown menu set to 'Marca', and 'Criterios de Búsqueda:' with a text input field containing 'HP' and a 'Consultar' button. The main area displays a table with the following data:

Código	Nombre	Descripción	Precio	Fecha de Ingreso	Imagen	Marca
100	PC HP	Laptop Pavilion	1000	27/11/2016 19:00	C:\Users\Danny\...	HP
101	PC HP	LAPTOP PAVILI...	1000	27/11/2016 19:00	C:\Users\Danny\...	HP
105	Impresora HP	Impresora Multifu...	700	30/11/2016 19:00	C:\Users\Danny\...	HP

Figura 74. Ventana de consulta de productos

3.4.5 Asignación de TAG's a Productos

Para realizar la asignación de Tags y su respectiva validación es necesario que los productos ya estén registrados en el sistema, en la pestaña asignación de Tags se activa el lector RFID para que pueda realizar un escaneo de Tags y se genera un tag, como se muestra en la figura 75.

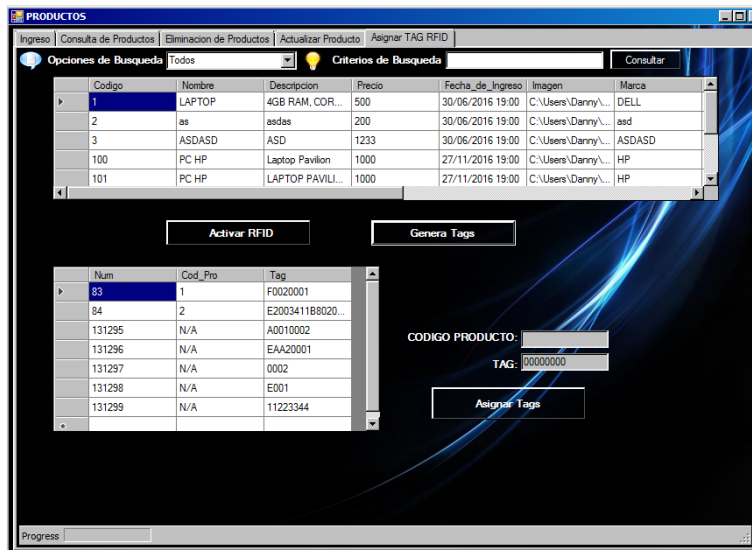


Figura 75. Visualización de productos y Tags

Se seleccionó un producto de la lista desplegada y se lo asocia con un Tag leído por el lector, como se muestra en la figura 76.

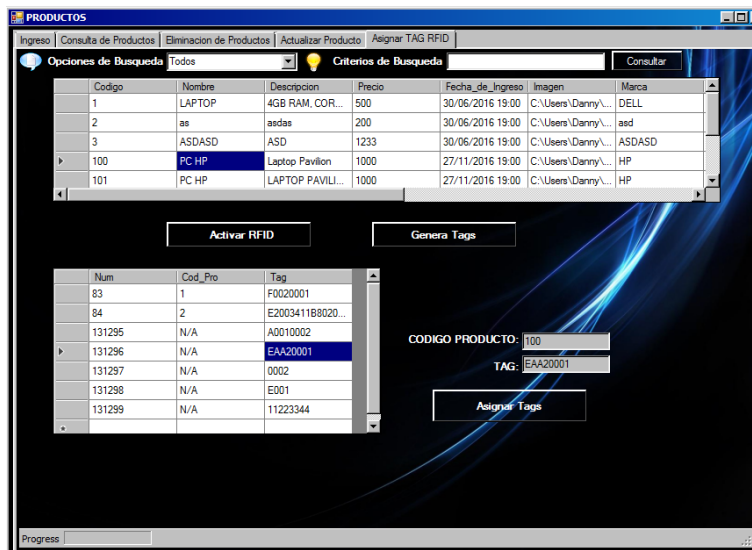


Figura 76. Asociación de producto y tag

Una vez asociado el producto a su respectivo tag el sistema indica que el tag fue asignado exitosamente, como se muestra en la figura 77.

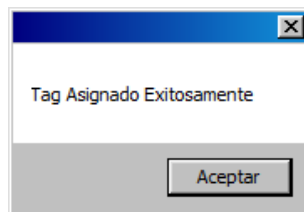


Figura 77. Mensaje de asignación exitosa

Y se lo puede visualizar directamente en pantalla el ID del Tag con el producto asociado, como se muestra en la figura 78, en este ejemplo se asoció el Tag EAA20001 al producto 100 (Una laptop HP pavilion).

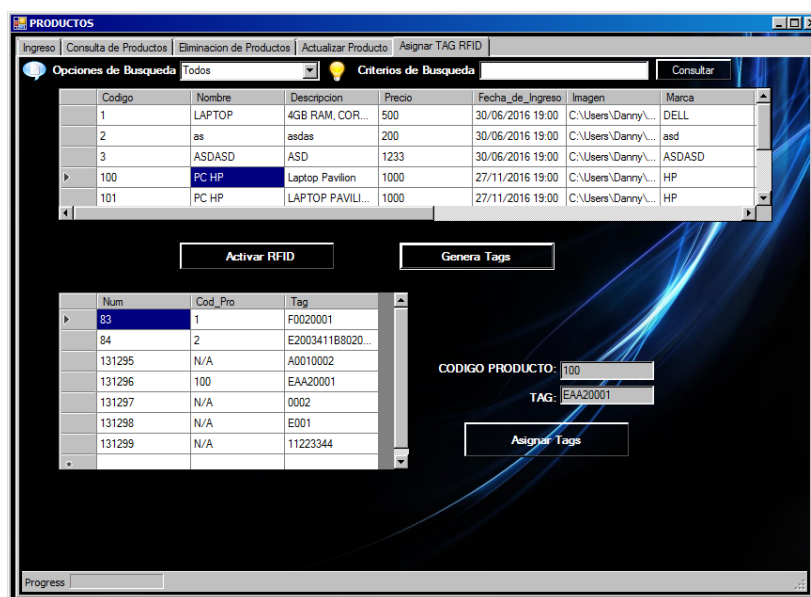


Figura 78. Revisión de tag asignado a producto

3.5 Pruebas de Gestión de Inventario

Aquí se llevaron a cabo las pruebas de ingreso y salida de productos inventariados, validación del total de productos existentes a los cuales se les ha asignado Tags, los productos sin asignar y los productos fuera de stock.

Para realizar las pruebas se asignaron Tags a más productos, como se muestra en la figura 79. Se agregaron Tags a los productos con código 100, 101, 102, 103, 104 respectivamente.

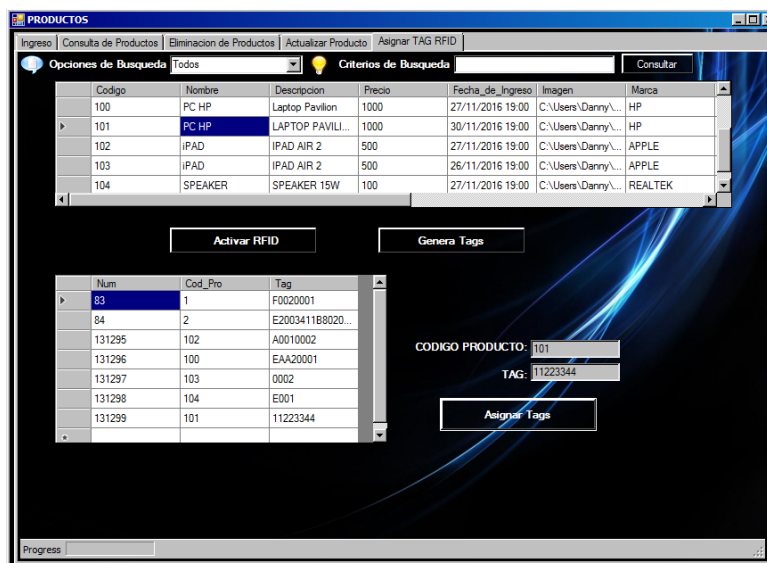


Figura 79. Asignación de Tags

3.5.1 Escaneo de Productos

Para validar que el sistema funcione correctamente primero se realizó un test de lectura de Tags donde se consultó que tag está asignado a cada producto y los Tags que aún no han sido asignados como se muestra en la figura 80.

	Num	Cod_Pro	Tag
▶	83	1	F0020001
	84	2	E2003411B8020...
	131300	N/A	A0060002
	131295	102	A0010002
	131296	100	EAA20001
	131297	103	0002
	131298	104	E001
	131299	101	11223344
•			

Figura 80. Consulta Tags asignados

En este caso siete producto tienen un tag asignado y hay uno que no se encuentra asignado.

3.5.2 Productos Existentes

El sistema al seleccionar esta opción indica el total de productos existentes que llevan asignado un tag, como se muestra en la figura 81.

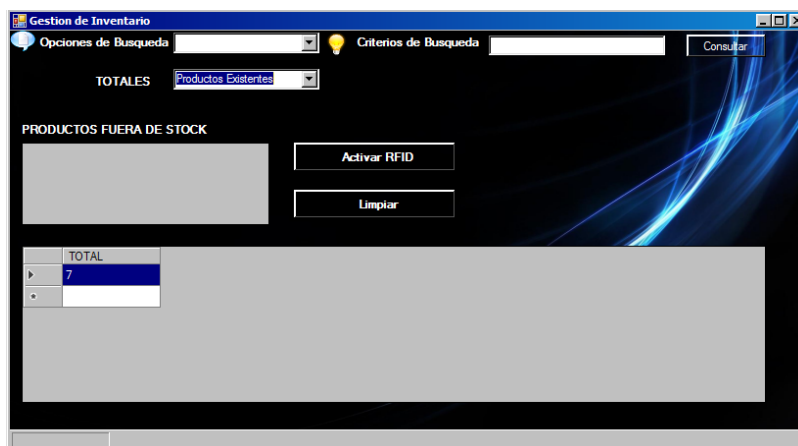


Figura 81. Total de productos existentes

Al compararlo con la base de datos se comprobó que la información que muestra el sistema es correcta, como se muestra en la figura 82. El tag de código A0060002 no cuenta ya que no está asociado a un producto por lo que son solo 7 los productos totales.

	Num	Cod_Pro	Tag
1	83	1	F0020001
2	84	2	E2003411B802011618305183
3	131300	N/A	A0060002
4	131295	102	A0010002
5	131296	100	EAA20001
6	131297	103	0002
7	131298	104	E001
8	131299	101	11223344

Figura 82. Consulta de productos y Tags

3.5.3 Productos sin Asignar

En esta opción el sistema indica el total de Tags que aún no están asignados a un producto, como se muestra en la figura 83.

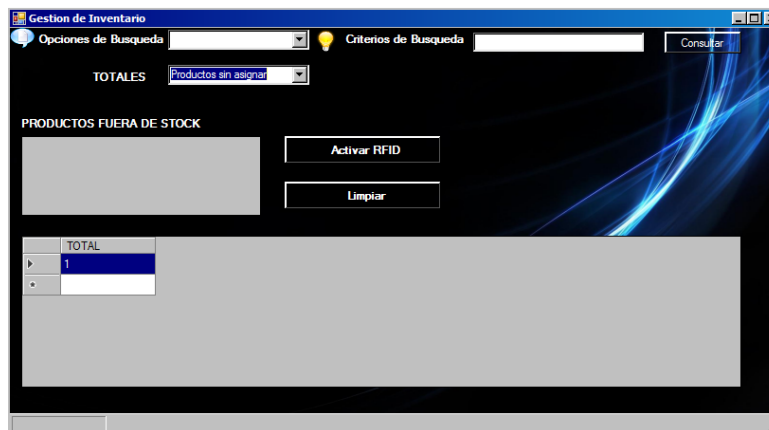


Figura 83. Total de productos sin asignar

Para comprobar esta información se realizó una consulta a la base de datos, como se muestra en la figura 82 de la sección anterior, la cual indica claramente que existe un tag cuyo código es N/A es decir que no está asignado a un producto.

3.5.4 Productos Fuera de Bodega

En esta sección el sistema indica cuando un producto salió de inventario es decir cuando un producto se encuentra fuera de la cobertura del lector RFID y lo despliega en pantalla, como se muestra en la figura 84.

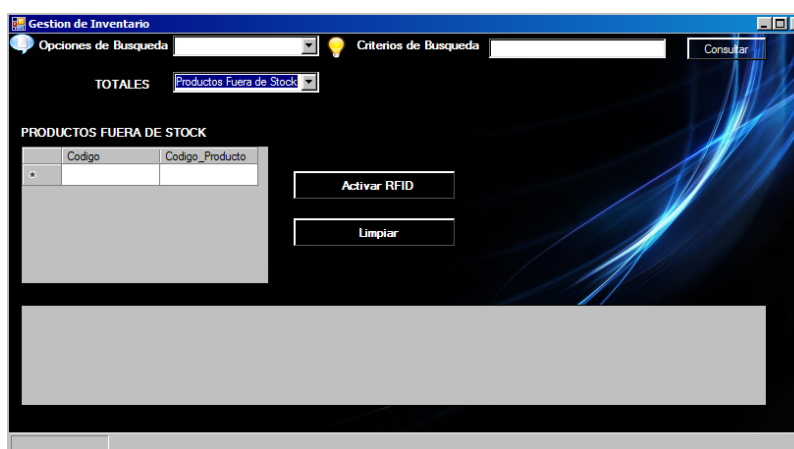


Figura 84. Productos fuera de bodega

En este caso todos los productos se encuentran en bodega ya que el lector los puede leer, por lo que en pantalla no refleja ningún producto fuera.

Se realizó otra prueba en la que se sacaron 4 productos de bodega para que el sistema los pueda validar e identificar que productos salieron, como se muestra en la figura 85.

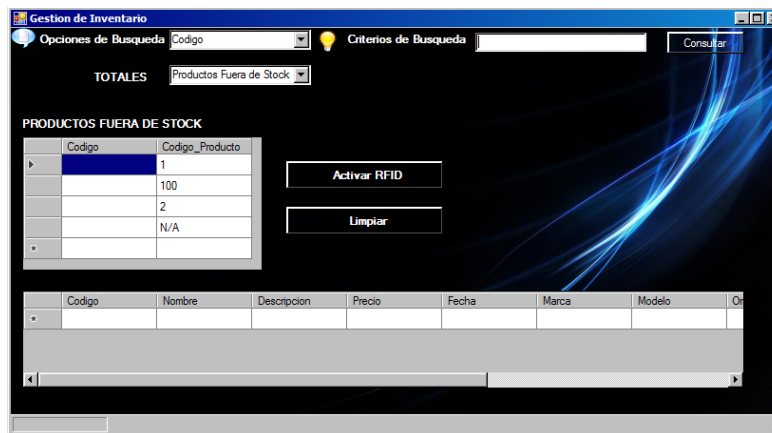


Figura 85. Código de productos fuera de stock

El sistema indica que salieron 4 productos cuyos códigos son 1, 100, 2 y N/A un tag que no estaba asociado a un producto, validando de esta manera los productos que se encuentran fuera de bodega.

3.6 Pruebas de Cobertura del Lector RFID

Para realizar las pruebas de cobertura del lector RFID partimos de las sugerencias que realiza el fabricante con respecto a la colocación y orientación del equipo. En la Figura 86 se muestra como se encuentra ubicado el lector para realizar las pruebas de cobertura.

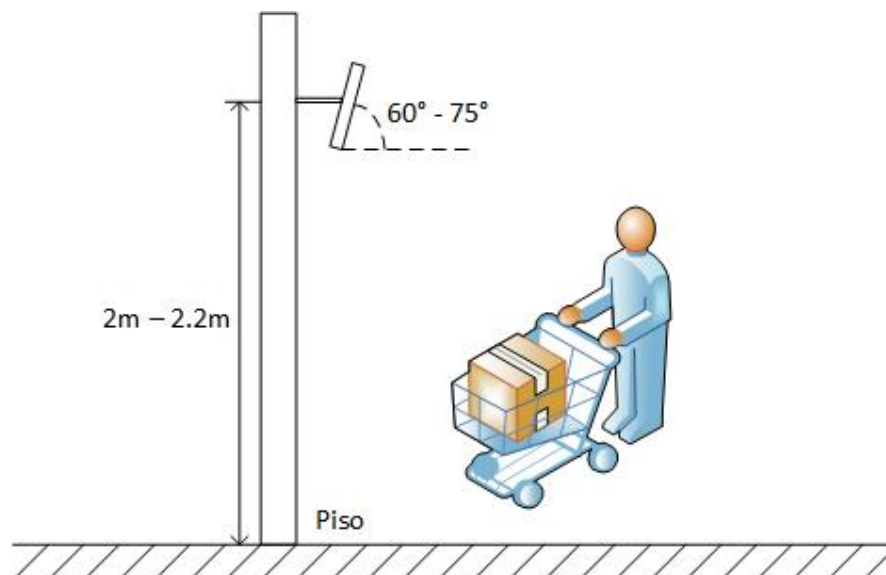


Figura 86. Ubicación del lector

Para realizar las pruebas se tomarán las medidas mínimas y máximas que el lector puede localizar al tag si se encuentra en la misma línea de vista y también con las desviaciones hacia los extremos izquierdo y derecho.

Tabla 28.

Cobertura del Lector RFID

Distancia	Cobertura Posición frontal del TAG	Cobertura Desviación Izquierda del TAG			Cobertura Desviación derecha del TAG		
		20°	40°	60°	20°	40°	60°
1m	100%	100%	100%	100%	100%	100%	100%
2m	100%	100%	100%	100%	100%	100%	100%
3m	95%	95%	90%	85%	95%	90%	85%
4m	90%	90%	80%	70%	90%	80%	70%
5m	60%	50%	40%	30%	50%	40%	30%
6m	20%	10%	0%	0%	10%	0%	0%

Analizando los valores de cobertura de la Tabla 28 encontramos que a partir de una distancia de 1m de ubicación del lector RFID cumple para cubrir una puerta de acceso con un área de 4m²

4. ANÁLISIS COSTO/BENEFICIO

Para la elaboración del prototipo de gestión de inventario cuyo objetivo es que utilizando una tecnología RFID nos permita llevar una administración más eficiente de un inventario al centralizarlo en un sistema de gestión con una interfaz amigable con el usuario.

Para comprobar que el desarrollo del prototipo es viable se realizará un análisis de costos, en donde se tomarán en cuenta características fundamentales para la implementación del prototipo (software y hardware), de esta forma estableceremos la factibilidad económica del proyecto.

El estudio de costos se establecerá en período de un año aproximadamente, los valores que se analizarán a continuación son estimados de los posibles valores si el prototipo fuese comercializado y los beneficios se estiman en función de la obtención inmediata de información del estado de los productos en bodega el cual ayuda a tener un reabastecimiento adecuado de productos para ser comercializados y entregados a clientes en las fechas establecidas, sin contratiempos quedando el cliente satisfecho. Como se indica en la tabla 29.

Tabla 29.
Análisis Costo/Beneficio.

ANALISIS COSTO/BENEFICIO			
COSTOS		BENEFICIOS	
DESCRIPCION	VALOR	DESCRIPCION	VALOR
Computadora de Escritorio con sistema operativo Windows XP o posterior	\$ 400	Mayor agilidad en manejo de inventario, stock, entradas y salidas de productos	\$ 250
Base de datos SQL Express server 2008 o superior	\$ 0	Mejor manejo de información, organización y seguridad en los datos.	\$ 100
Equipo RFID	\$ 300	Mejorada gestión de inventario	\$ 200
Tag's RFID (30)	\$ 60	Mejora la identificación de cada producto	\$ 200
Cables de conexión	\$ 5	Conexión rápida y segura entre el lector y PC	\$ 100
Costo Total	\$ 765	Beneficios Totales	\$ 850

Los beneficios descritos en la tabla 1.17 se estiman en función de:

- Mayor agilidad en manejo de inventario, stock, entradas y salidas de productos. Este beneficio se estima en función de que va a tener un mejor control sobre todos los productos en bodega así como sus movimientos (entradas/salidas) en tiempo real, ahorrándose en este sentido la necesidad de invertir demasiadas horas de trabajo realizando ese proceso manualmente. Por lo que este beneficio es valorado en \$250 anual.
- Mejor manejo de información, organización y seguridad en los datos. Este beneficio se estima en función de que al utilizar SQL Server como gestor de base de datos el manejo de la información es más rápido, se estructura y organiza de una manera eficiente, tiene un alto grado de seguridad ya que las transacciones y obtención de la información es encriptada y su acceso es través de permisos otorgados a cada usuario. Con esto se evita el almacenamiento de información de los productos físicamente en papeles y

posteriormente archivarlos en un lugar seguro para evitar pérdidas de información, su recuperación y seguimiento de un producto de esta forma manual toma demasiado tiempo, recursos y da cabida a errores. Por lo que este beneficio es valorado en \$100 anual.

- Mejorada gestión de inventario. Este beneficio se estima ya que utilizando la tecnología RFID el prototipo automatiza la gestión de inventario, eliminando el manejo de inventario manual y los errores propios de este método, adicionalmente que se necesita mucho tiempo para realizarlo. Por lo que este beneficio es valorado en \$200 anual.

- Mejora la identificación de cada producto. Este beneficio se estima ya que cada producto al utilizar un TAG RFID asociado puede ser plenamente identificado, evitando conflictos o errores de ambigüedad en los detalles de productos si se los maneja de forma manual. Por lo que este beneficio es valorado en \$200 anual.

- Conexión rápida y segura entre el lector y PC. Este beneficio se estima en función de que la conexión entre el lector y PC es a través de un cable con comunicación serial con bits de paridad para aumentar la seguridad en la transmisión de información y evitar el acceso a personas no autorizadas. En el sistema tradicional de llevar inventario de forma manual no tiene seguridad (confidencialidad) del inventario que posee ya que fácilmente personas no autorizadas pueden tener acceso a esa información. Por lo que este beneficio es valorado en \$100 anual.

Análisis de Resultados.

Los resultados según la tabla 1.17. son alentadores con la idea de realizar un sistema de gestión de inventario automatizado y centralizado utilizando RFID. Los cálculos demostraron un beneficio básico para el primer año (\$85), sin embargo, la relación de beneficio a costo es de \$1,11 de retorno por cada dólar gastado (\$850/\$765). Es un retorno positivo ya que se trata de un prototipo.

4.1 Punto de Equilibrio

El punto de equilibrio es un indicador sencillo que se obtiene del análisis costo/beneficio el cual indica el tiempo que tomaría para que el total de los beneficios sea igual al costo total, en el caso de este prototipo de 1 año. Como se muestra en la tabla 30.

Para el cálculo del punto de equilibrio definiremos algunas abreviaturas que son:

PE = Punto de Equilibrio.

CT = Costo Total (Obtenido del análisis del Costo/Beneficio).

BT= Beneficio Total (Obtenido del análisis del Costo/Beneficio).

T = Período de tiempo estimado.

Tabla 30.

Punto de Equilibrio.

Punto de Equilibrio			
	\$	Beneficio	\$
Costo Total	765,00	Total	850,00
Cálculo			
$PE = \frac{CT}{\frac{BE}{765}} \times T$ $PE = \frac{850}{850} \times 12$ $PE = 10,8$			
PE: 10,8			

Este resultado indica que a partir del primer año se empezará a obtener utilidades las cuales no serían muy altas ni notorias tomando en cuenta que es un prototipo.

4.2 Período de Devolución

Es el tiempo requerido para recuperar el monto inicial total requerido en la construcción del prototipo de gestión de inventario, el análisis del período de devolución al igual que el análisis anterior de punto de equilibrio no toman en cuenta el valor del dinero en el tiempo. Como se muestra en la tabla 31.

Para el cálculo del período de devolución definiremos algunas abreviaturas que son:

PE = Punto de Equilibrio.

VA = Valor asegurado

CT = Costo Total (Obtenido del análisis del Costo/Beneficio).

BT= Beneficio Total (Obtenido del análisis del Costo/Beneficio).

T = Período de tiempo estimado.

Tabla 31.
Período de Devolución.

PERIODO DE DEVOLUCION		
DATOS		
COSTO TOTAL	BENEFICIOS TOTALES	VALOR ASEGURADO
\$ 765	\$ 850	\$ 200
FORMULA		
$PD = \frac{CT - VA}{\frac{BT \times T}{765 - 200}}$ $PD = \frac{850 \times 12}{850 \times 12}$ $PD = 0,055$		
EL PERIODO DE DEVOLUCION ES: 0,055		

Según los resultados obtenidos los costos de operación para la construcción del prototipo son equilibrados con los beneficios, por lo que si el prototipo fuese comercializado no sería rechazado.

4.3 Valor Actual Neto (VAN)

Es un indicador financiero para poder establecer los flujos de los futuros ingresos y egresos de un proyecto, se lo determina si luego de descontar la inversión inicial queda alguna ganancia, lo cual indica que la construcción de este prototipo es viable. Como se puede apreciar en la tabla 32.

Para el cálculo del valor actual neto (VAN) definiremos algunas abreviaturas que son:

BNA = Beneficio Neto Actualizado,

Inv = Inversión.

d = Tasa de descuento (Según el BCE)

t1 = tiempo en año 1

t2 = tiempo en año 2

t3 = tiempo en año 3

Tabla 32.
Valor Actual Neto.

VALOR ACTUAL NETO (VAN)		
DATOS		
INVERSION: \$765	BNA PROYECTADO \$150, \$250, \$400 TIEMPO ESTIMADO DE 1 A 3 AÑOS	d = 12%
FORMULA		
$VAN = BNA - Inv.$ $VAN = \left(\frac{BNA1}{1 + d^{t1}} + \frac{BNA2}{1 + d^{t2}} + \frac{BNA3}{1 + d^{t3}} \right) - Inv$ $VAN = \left(\frac{150}{1 + 0,12^1} + \frac{250}{1 + 0,12^2} + \frac{400}{1 + 0,12^3} \right) - 765$ $VAN = 14,8$		
VAN= 14,8		

Los resultados se avalúan en función del valor neto actual que indica si:

- VAN \geq 0 el proyecto es rentable.
- VAN $<$ 0 el proyecto no es rentable.

Con los datos obtenidos en la tabla analizando el VAN del prototipo demostramos que es rentable su implementación.

4.4 Tasa Interna de Retorno (TIR)

Es la tasa de descuento que tendrá el proyecto que le permita que el beneficio neto sea mayor o igual a la inversión, es decir la tasa máxima de descuento que se puede tener para que el prototipo sea rentable.

Para el cálculo del TIR se iguala el valor actual neto a cero para obtener la mínima tasa de descuento. Como se muestra en la tabla 33.

Tabla 33.
Tasa Interna de Retorno.

TASA INTERNA DE RETORNO	
DATOS	
INVERSION: \$765	Beneficio Neto actualizado BNA proyectado: \$150 a \$400 tiempo estimado de 1 a 3 años
FORMULA	
$VAN = (BNA - Inv) = 0$ $0 = \left(\frac{BNA1}{1 + i^{t1}} + \frac{BNA2}{1 + i^{t2}} + \frac{BNA3}{1 + i^{t3}} \right) - Inv$ $0 = \left(\frac{150}{1 + i^1} + \frac{250}{1 + i^2} + \frac{400}{1 + i^3} \right) - 765$ $i = 2\%$	
TIR = 2%	

Obteniendo el cálculo del TIR es del 2% por lo que a menor tasa el prototipo sería más rentable, ya que el beneficio neto sería mayor a la inversión.

4.5 Análisis de resultados del Estudio Costo/Beneficio

Después de realizar el análisis Costo/Beneficio se pudo verificar que la implementación de este prototipo de gestión de inventario es viable y rentable, ya que presenta beneficios dentro de un ciclo determinado. Como se indica en la figura 87.

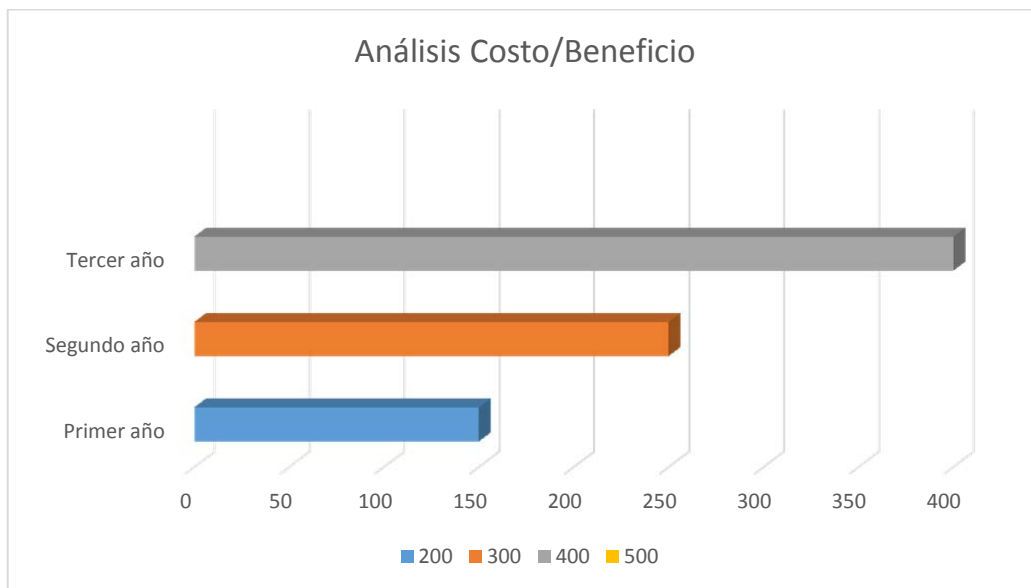


Figura 87. Análisis Costo/Beneficio

5. CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

El desarrollo del prototipo fue orientado de una manera específica a una bodega tipo, pero puede ser escalable en función de los requerimientos necesarios y los equipos utilizados.

La integración entre el hardware y software es esencial por lo que hay que tomar en cuenta que los drivers para cada lector RFID es diferente dependiendo de la versión de Windows utilizada.

Utilizando RFID se puede realizar un monitoreo a tiempo real de los productos inventariados por lo que su administración se torna fácil y eficiente.

El diseño de software a través de módulos facilita a los usuarios el uso y administración, permitiendo manejar un gran número de opciones de una manera sencilla.

El utilizar RFID para control de inventario es más seguro ya que los lectores incorporan hardware de alta seguridad y cifrado en el almacenamiento y transmisión de información.

En el diseño e implementación del prototipo se utilizaron conocimientos obtenidos en redes inalámbricas, seguridad de redes, programación orientada a objetos y bases de datos.

El desarrollo de la aplicación en un lenguaje tal como C# que es ampliamente utilizado, garantiza la compatibilidad del software con los diferentes sistemas operativos de PC's en el mercado.

El desarrollo de la aplicación en un esquema de programación por capas facilita el mantenimiento de la misma, ya que permite el acceso a la capa

específica que se necesita modificar o actualizar sin incurrir en grandes cambios que afecten el funcionamiento total del sistema y de esta manera también permite que el escalamiento de la aplicación sea sencillo.

El uso de un gestor de base de datos facilita la recuperación de la información de una manera rápida y eficiente optimizando recursos.

La información inventariada es segura ya que al ser almacenada en una base de datos solo tienen acceso personas previamente autorizadas.

5.2 Recomendaciones

Al manejar una base de datos donde se guarda información importante se recomienda la generación de respaldos de la misma ya que de esta manera brindamos seguridad y confiabilidad al sistema.

Se recomienda que el equipo donde funciona la aplicación y la base de datos cumpla con el mínimo de requerimiento establecido para garantizar su fluidez y rendimiento.

Se recomienda utilizar el prototipo en lugares donde no existan interferencias de frecuencia ya que pueden generar conflictos en la transmisión de información a los Tags y podría haber pérdidas de comunicación.

Según el tipo de producto a inventariar se recomienda buscar un Tag apropiado para su identificación esta pueden ser tipo etiquetas adhesivas, tarjetas, barras para metales, entre otros.

Para poder manipular el lector RFID es necesario importar su driver que el fabrica lo debe proveer junto al equipo, esto es muy importante a la hora de poner en marcha el sistema.

REFERENCIAS

- EPC-RFID INFO. (s.f.). *How does a RFID system work?*. Recuperado el 18 de octubre de 2016 de <http://www.epc-rfid.info/rfid>
- RFID Journal. (s.f.). *Frequently asked questions* . Recuperado el 18 de octubre de 2016 de <http://www.rfidjournal.com/site/faqs#Anchor-What-363>
- Tecnología RFID. (s.f.). Tecnología RFID. Recuperado el 18 de octubre de 2016 de <https://fiordalisapena.wordpress.com/2011/08/08/tecnologia-rfid/>
- IMPINJ. (s.f.). *RFID Frequencies*. Recuperado el 20 de octubre del 2016 de <http://www.impinj.com/resources/about-rfid/the-different-types-of-rfid-systems/>
- Landt J. (2001). *Shrouds of Time The history of RFID*. Recuperado el 30 de octubre de 2016 de <https://www.transcore.com/sites/default/files/History%20of%20RFID%20White%20Paper.pdf>
- AB&R. (s.f.). *RFID Applications*. Recuperado el 30 de octubre de 2016 de <http://www.abr.com/what-is-rfid-how-does-rfid-work/>
- nControl. (s.f.). *O que é RFID?*. Recuperado el 1 de noviembre de 2016 de <http://www.ncontrol.com.pt/o-que-e-rfid.html>
- ERPlan. (s.f.). RFID. Recuperado el 1 de noviembre de 2016 de <http://www.erplan.it/wordpress/rfid/>
- Microsoft. (s.f.). *Tutorial 2: Creating a Business Logic Layer*. Recuperado el 4 de noviembre de 2016 de <https://msdn.microsoft.com/en-us/library/aa581779.aspx>
- WIKILIBROS. (s.f.). Programación en PHP. Recuperado el 4 de noviembre de 2016 de https://es.wikibooks.org/wiki/Programación_en_PHP
- La Revista Informática. (s.f.). LENGUAJE DE PROGRAMACION DELPHI. Recuperado el 5 de noviembre de 2016 de <http://www.larevistainformatica.com/Delphi.htm>
- Ramírez J. (2008). ¿Qué es Delphi?. Recuperado el 7 de noviembre de 2016 de <http://ticjunior.blogspot.com/2008/07/que-es-delphi.html>
- Microsoft. (s.f.). Lenguaje Visual C#. Recuperado el 7 de noviembre de 2016 de [https://msdn.microsoft.com/es-ec/library/aa287558\(v=vs.71\).aspx](https://msdn.microsoft.com/es-ec/library/aa287558(v=vs.71).aspx)

- Microsoft. (s.f.). Microsoft SQL Server. Recuperado el 7 de noviembre de 2016 de <https://msdn.microsoft.com/es-es/library/bb545450.aspx>
- Masip D. (2002). Oracle. Recuperado el 7 de noviembre de 2016 de <http://www.desarrolloweb.com/articulos/840.php>
- Rouse M. (2015). MySQL. Recuperado el 10 de noviembre de 2016 de <http://searchdatacenter.techtarget.com/es/definicion/MySQL>
- Kimaldi. (s.f.). Lector RFID UHF. Recuperado el 10 de noviembre de 2016 de http://www.kimaldi.com/productos/sistemas_rfid/lector_de_tarjetas_rfid_y_tags_uhf/lectores_rfid_uhf/lectores_rfid_uhf_para_aplicaciones_industriales/lector_uhf_de_largo_alcance
- Strong Link. (s.f.). Lector Integrado UHF SL130. Recuperado el 12 de noviembre de 2016 de <http://www.stronglink-rfid.com/es/rfid-readers/sl130.html>
- RFID SHOP. (s.f.). UHF-RW-MP-232-V1. Recuperado el 12 de noviembre de 2016 de <http://rfidshop.com.hk/datasheet/UHF-Reader/RR-RFID/UHF-RW-MP-232-V1%20user%20guide.pdf>
- DATE, C.J. (2001). Introducción a los sistemas de bases de datos. (7.a ed.). México: PEARSON EDUCATION

ANEXOS

ANEXO I

Manual de Usuario

1. Inicio de sesión, el usuario y contraseña definido para ingresar a la aplicación por defecto es "root" y contraseña "12345". El usuario puede ingresar a la aplicación aún sin haber configurado ni tener usuarios registrados en la base de datos, como se muestra en la figura 1.

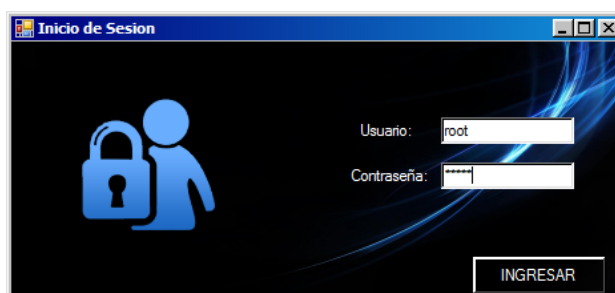


Figura 1.

2. Al ingresar al menú del sistema se despliegan todos los módulos ya que este usuario es un administrador local, como se muestra en la figura 2.



Figura 2.

3. Una vez instalada y configurada la base de datos, se hace el ingreso de usuarios al sistema, por lo que se selecciona el módulo gestión de usuarios y se llenan los campos solicitados en el formulario, en el combo box se selecciona la opción de nuevo, como se muestra en la figura 3.

Figura 3.

Creado el usuario se le asigna el permisos pertinente desde la ventana permisos en el módulo de gestión de productos, como se muestra en la figura Este paso también se utiliza para cambiar de permisos a un usuarios ya existente.

Figura 4.

De la misma manera se realiza la actualización o eliminación de un usuario llenando los campos del formulario y en el combo box seleccionando actualizar o borrar según sea el caso.

4. En el módulo parámetros RFID el usuario establece la comunicación hacia el lector RFID, seleccionando el puerto COM al cual se conecta o dejándolo en automático para que el sistema reconozca al puerto que está conectado, como se muestra en la figura 5.

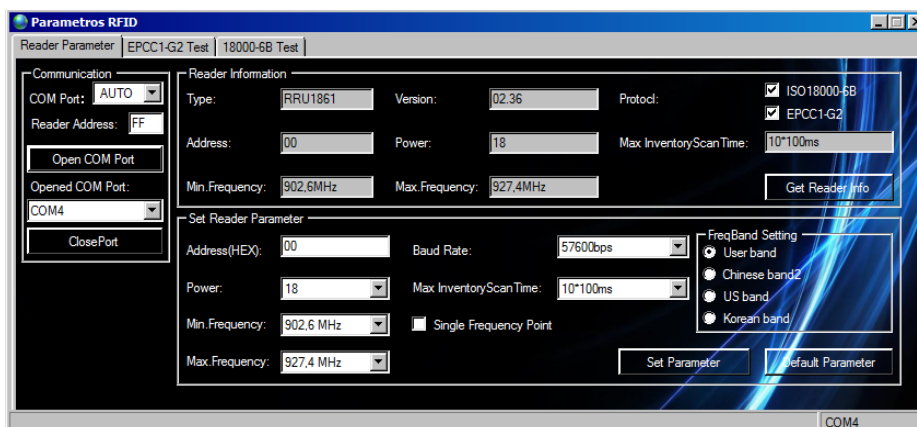


Figura 5.

El equipo se conecta automáticamente y despliega en la pantalla sus características principales, como se muestra en la figura 5.

5. En el módulo de gestión de productos el usuario puede crear, actualizar, eliminar, realizar consultas sobre los productos registrados y asignar un tag a un producto.

5.1 Crear producto.

El usuario selecciona la pestaña ingreso y llena la información solicitada, como se muestra en la figura 6. Cuando se guarde el producto el sistema indica que se ha guardado el registro exitosamente.

Figura 6.

5.2 Consultar Productos.

El usuario debe ingresar a la pestaña consulta de productos, como se muestra en la figura 7, donde en opciones de búsqueda escoge si es por código, nombre, marca, serie o todos y en criterio de búsqueda se digita la información solicitada según la opción.

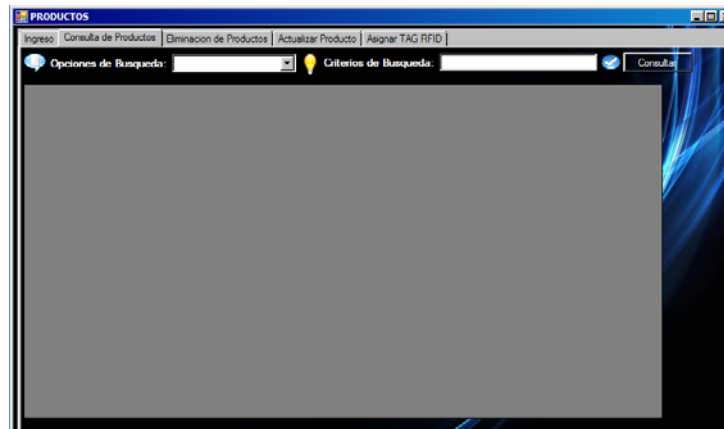


Figura 7.

5.3 Eliminar Productos.


El usuario selecciona la pestaña eliminar producto como se muestra en la figura 8, en opciones a eliminar se escoge entre código, nombre o serie para realizar la acción, inmediatamente el sistema mostrara en pantalla la lista de todos los productos existentes.



Figura 8.

5.4 Actualizar Productos.

El usuario selecciona la pestaña actualizar producto, como se muestra en la figura 9, donde se debe escoger la opción de búsqueda (código, nombre o serie) como parámetros de actualización, se desplegará en pantalla la información del producto solicitado donde se podrán realizar los cambios necesarios y guardar la información



The screenshot shows a web browser window titled 'PRODUCTOS'. The navigation menu includes 'Ingreso', 'Consulta de Productos', 'Eliminación de Productos', 'Actualizar Producto', and 'Asignar TAG RFID'. The 'Actualizar Producto' tab is active. Below the navigation, there are two dropdown menus: 'Opciones Actualizar' and 'Criterios de Actualización', followed by an 'Examinar' button. The main content area features a large blue arrow icon and the title 'ACTUALIZAR PRODUCTOS'. Below this, there is a form with the following fields: 'Codigo del Producto', 'Nombre del Producto', 'Descripcion del Producto', 'Cantidad', 'Precio', 'Fecha de Ingreso' (with a date of 09/06/2011), 'Imagen del Producto', 'Marca', 'Grupo', 'Codigo del Fabricante', 'Serie', 'Origen', and 'Garantia'. There are 'Examinar' and 'Actualizar Registro' buttons.

Figura 9.

5.5 Asignación de tags a Productos.

Para asignar un tag a un producto el usuario selecciona la pestaña asignar tag RFID, como se muestra en la figura 10, realizada la búsqueda de productos ingresados mediante las opciones y criterios de búsqueda, una vez desplegados el o los productos, el usuario da click en activar RFID donde establece el puerto de comunicación (COM) y escanea los tags, como se muestra en la figura 10 y 11.

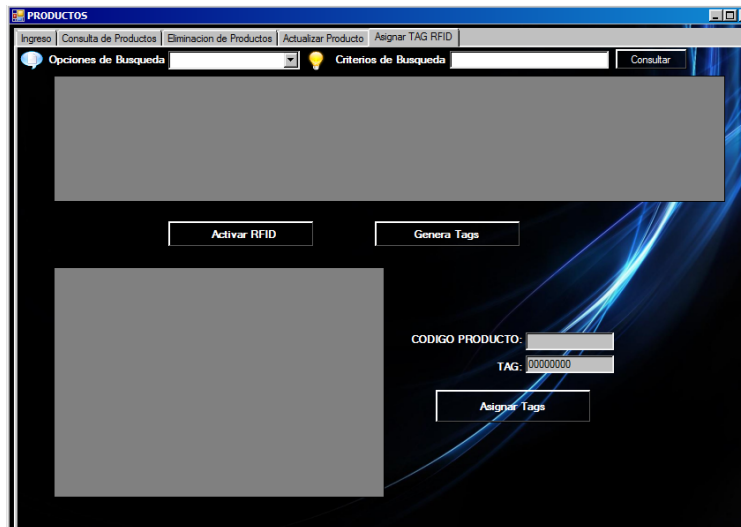


Figura 10

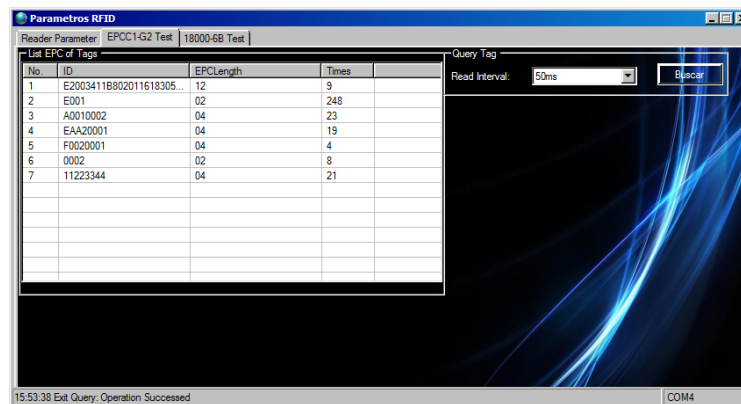


Figura 11

Se da clic en el botón generar tag donde la aplicación muestra los tags leídos por el lector, como se muestra en la figura 12

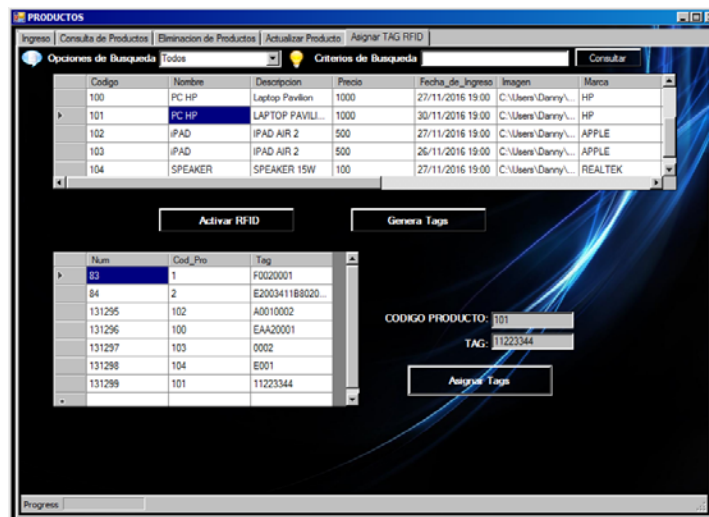


Figura 12.

Haciendo clic seleccionamos el producto de la tabla y tag de la otra tabla el cuál se va a asignar como se muestra en la figura 12.

Una vez asignado el tag el sistema indicará mediante un mensaje que fue exitosa la transacción y se reflejará en la pantalla el código del producto y tag asignado. Como se muestra en la figura 13.

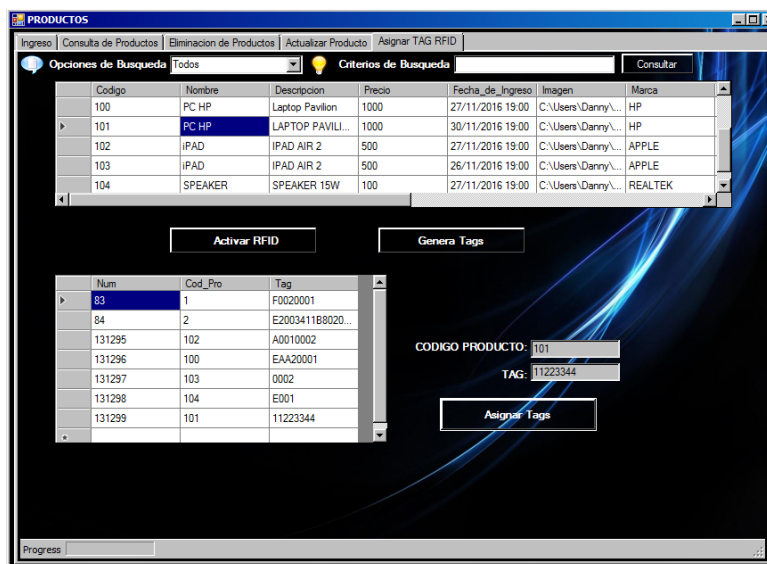


Figura 13.

6. En el módulo gestión inventario podemos monitorear los productos que están inventariados, los que están fuera o los que no han sido asignados un

tag. Para lo cual primero seleccionamos en el combo box TOTALES la operación que se desea realizar, como se muestra en la figura 14

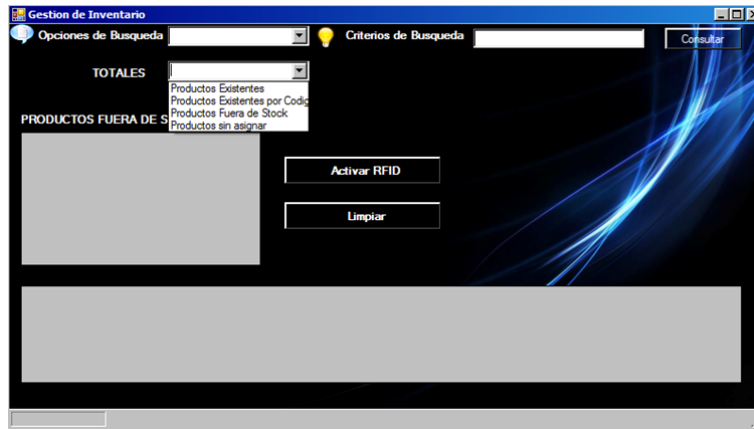


Figura 14.

Para ver el total de productos existentes se selecciona esa opción y automáticamente se muestra en pantalla esa información, como se muestra en la figura 15.

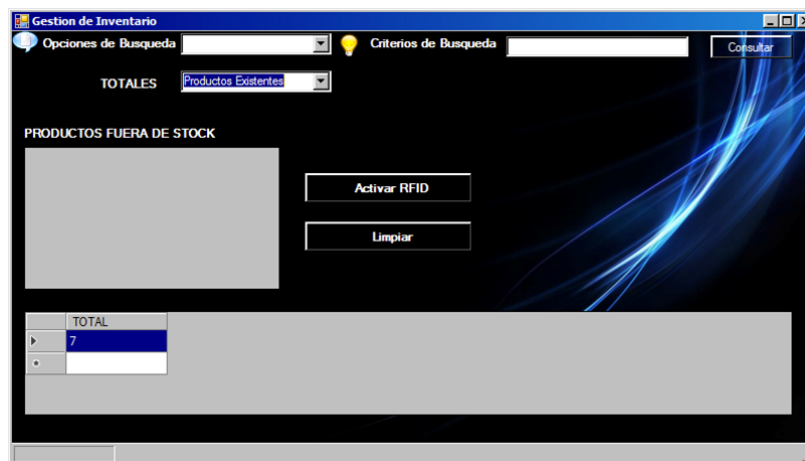


Figura 15.

Para mostrar los productos fuera de stock primero se activa el lector RFID como se lo realizó en la sección anterior de asignar un tag a un producto, después el sistema se encargará de leer los tags y los que se encuentren fuera de cobertura los asume como fuera de stock y se muestra en pantalla figura 16.

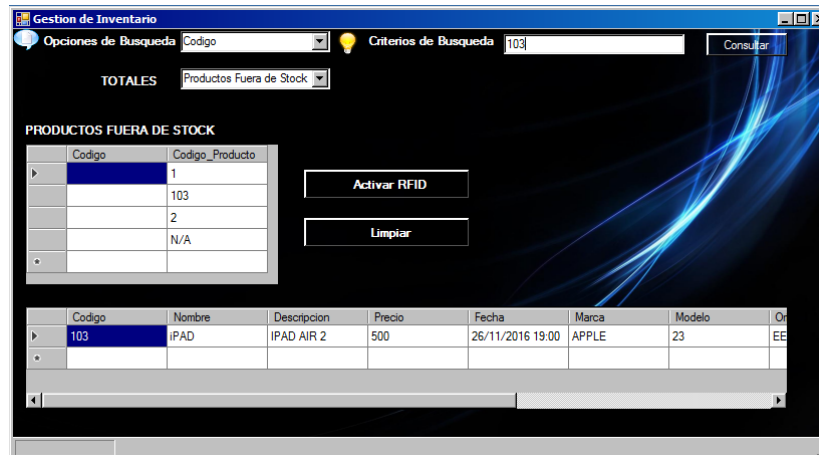


Figura16.

Y para mostrar el total de tags que no han sido asignados a un producto se selecciona esta opción y el sistema muestra en pantalla el total de tags no asignados Figura 17.

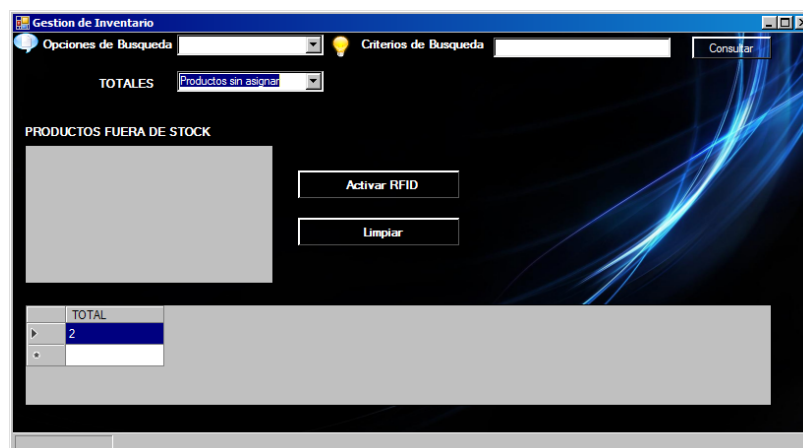


Figura 17

ANEXO II

Principales características de los lectores RFID

2.1 Lector Kimaldi

2.1.1 Características

- Frecuencia operacional: 902 - 928MHz, 865 - 868MHz, RF banda personalizable (opcional)
- Frecuencia Hopping FHSS o frecuencia fija.
- Potencia de salida: +20dBm ~ +30dBm@50Ω en carga, 1dBm en cada paso.
- Antena integrada de 12dBi de polarización lineal (personalizable horizontal/vertical).
- Interfaces de comunicación: RS-232, Wiegand 26/34
- Interfaces opcionales: RS-485, 802.11 Modulo y 10M/100M interfaces Ethernet
- Interfaces de I/O de 2 canales (Relay Output), 2 canales Trigger Input, 2 canales level Output (opcional)
- Actualización de firmware
- Plataforma de software de aplicación: API(C++, C#) kit de desarrollo
- Protocolos Tag: ISO18000-6B, EPC Class 1 Gen 2(ISO18000-6C)
- Rango de lectura: 0~15m
- Rango de escritura: 0~5m
- Indice de protección IP: IP63
- Indicador: LED y zumbador (Potencia, Antena, Lectura/Escritura)
- Potencia: DC 12V entrada, adaptador de potencia
- Consumo de energía: 2W(Standby), 10W(Trabajo)
- Dimensiones del equipo: 445mm x 445mm x 78mm (LxAxP)
- Dimensiones del paquete: 600mm x 500mm x 110mm (LxAxP)
- Peso neto: 3.5kg
- Peso bruto: 6.23kg

- Temperatura de operación: -40°C ~ +60°C
- Temperatura de almacenamiento: -60°C ~ +80°C
- Humedad: 5% ~ 95%(non-condensing)

2.1.2 Funcionalidades

- Banda UHF (860-920MHz), equipado con una antena de 12 dBi
- Soporta estándar RS-232 e interfaz Wiegand
- Puertos de I/O Multi-grupo, controlando el modo de funcionamiento del lector/escritor
- Ajuste del tiempo de disparo para las aplicaciones del cliente
- Actualizar el firmware a través del puerto serial en línea
- Fuerte interfaz de usuario API
- Interfaces opcionales: Ethernet, RS-485
- Módulos WIFI opcionales: GPRS, 802.11
- Memoria Flash opcional

2.1.3 Aplicaciones

- Estacionamiento de vehículos, pesaje inteligente
- Almacén, gestión logística
- Gestión de activos
- Gestión de la línea de producción

2.2 RFID UHF SL130

2.2.1 Características

- Etiqueta de Lectura/Escritura: ISO18000-6B, EPC CLASS1 G2 tag
- Frecuencia:865~956MHz(ligeramente diferente en los distintos países y regiones)
- Tipo de trabajo: FHSS o frecuencia fija de emisión

- Potencia de salida:20~30dBm(ligeramente diferente en los distintos países y regiones)
- Distancia de lectura/escritura: distancia de lectura>8m,distancia de escritura >1m(dependiendo de las etiquetas)
- Velocidad de lectura: 32bits/6ms.
- Velocidad de escritura:32bits/50ms
- Fuente de energía: +9/12VDC.
- Potencia: potencia promedio<5W.
- Temperatura de trabajo:-10°C~+55°C

2.2.2 Funcionalidades

- Etiqueta de Lectura/Escritura:ISO18000-6B、EPCClass1 Gen 2tag
- Puede leer diferentes longitudes de números EPC (16, 32, 48, 64, 80 or 96 bits) al mismo tiempo
- Puede leer números EPC seleccionados de la etiqueta
- Puede leer datos seleccionados de la etiqueta en la memoria del usuario
- Puede leer datos seleccionados de la etiqueta en la memoria TID
- Se puede leer la contraseña de visita y quitar la contraseña de la etiqueta (si ambos son de 32 bits)
- Existen funciones de protección contra escritura para EPC, TID y memoria del usuario
- Hay funciones de protección de lectura y escritura para contraseña de memoria
- Se puede matar la etiqueta

2.3 UHF-RW-MP-232-V1-7dbi

2.3.1 Características

- Modo de operación: lectura /escritura UHF Etiqueta ID y memoria
- Soporta lectura multi etiqueta
- Antena: integrada de polarización circular
- Frecuencia Estándar: 902-928Mhz (puede ser definida por el usuario)
- Frecuencia Europea: 865-868Mhz (puede ser definida por el usuario)
- Interfaces opcionales: RS232 / wiegand
- Potencia de Transmisión: < 30dbm (programable por software)
- Protocolos soportados: ISO18000-6C EPC G2 o ISO18000-6B etiquetas UHF
- Fuente de energía: +9V DC ,3A
- Paquete de software libre (VC, C#,VB.Net y Delphi, programa demo con código fuente)
- Antena integrada de 7 dbi
- Rango de lectura: ~3-5 metros (distancia efectiva que depende de las características de las etiquetas y del ambiente de instalación)
- Pico de potencia de transmisión: < 1 watt
- Dimensiones: (220x220x30)mm
- Peso: 2kg

