



**BENEMÉRITA UNIVERSIDAD AUTÓNOMA  
DE PUEBLA**

**FACULTAD DE CIENCIAS DE LA  
COMPUTACIÓN**

***“BASE DE DATOS PARA EXPEDIENTES MÉDICOS  
UTILIZANDO ETIQUETAS RFID”***

**T E S I S**

**Para Obtener el Título de:**

**Licenciado en Ciencias de la Computación**

**PRESENTA:**

**López Meléndez Elizabeth**

**ASESOR:**

**Dra. Bárbara Emma Sánchez Rinza**

**PUEBLA, PUEBLA**

**AGOSTO 2009**

## ÍNDICE

<b>Introducción</b> .....	<b>4</b>
<b>Capítulo 1: Identificadores por Radiofrecuencia</b> .....	<b>6</b>
<b>Introducción</b> .....	<b>6</b>
<b>1.1 Historia de las Etiquetas RFID</b> .....	<b>6</b>
<b>1.2 Sistema RFID</b> .....	<b>7</b>
<b>1.2.1 Sistema RFID</b> .....	<b>7</b>
<b>1.2.2 Etiquetas RFID</b> .....	<b>10</b>
<b>1.2.3 Tipos de Etiquetas RFID</b> .....	<b>10</b>
<b>1.2.4 Funcionamiento de las Etiquetas RFID</b> .....	<b>13</b>
<b>1.2.5 Lectores</b> .....	<b>15</b>
<b>1.3 Frecuencia Utilizada</b> .....	<b>16</b>
<b>1.4 Tipos de Antena</b> .....	<b>18</b>
<b>1.5 Ventajas y Desventajas</b> .....	<b>19</b>
<b>1.6 RFID Middleware</b> .....	<b>21</b>
<b>1.7 Aplicaciones Actuales de las Etiquetas RFID</b> .....	<b>22</b>
<b>1.8 Aplicación de las Etiquetas RFID en la Medicina</b> .....	<b>23</b>
<b>Capítulo 2: Análisis y Diseño</b> .....	<b>24</b>
<b>Introducción</b> .....	<b>24</b>
<b>2.1 Bases de Datos</b> .....	<b>24</b>
<b>2.1.1 Modelado de Base de Datos</b> .....	<b>26</b>
<b>2.1.2 Modelo Entidad-Relación</b> .....	<b>27</b>
<b>2.1.2.1 Conceptos del Modelo Entidad – Relación</b> .....	<b>28</b>
<b>2.1.3 Diagrama Entidad-Relación</b> .....	<b>29</b>
<b>2.1.3.1 Notación del Diagrama Entidad-Relación</b> .....	<b>30</b>
<b>2.1.4 Diagrama Entidad-Relación Para El Sistema</b> .....	<b>31</b>
<b>2.2 UML (Lenguaje de Modelado Unificado)</b> .....	<b>32</b>
<b>2.2.1 Diagramas de Clase</b> .....	<b>33</b>
<b>2.2.1.1 Notación del Diagrama de Clase</b> .....	<b>35</b>
<b>2.2.1.2 Diagrama De Clases Para El Sistema</b> .....	<b>36</b>

2.2.2 Diagramas de Casos de Uso-----	38
2.2.2.1 Notación del Diagrama de Casos de Uso-----	39
2.2.2.2 Diagrama De Casos de Uso Para El Sistema -----	40
2.2.3 Diagramas de Estados -----	41
2.2.3.1 Notación del Diagrama de Estados -----	42
2.2.3.2 Diagrama De Estados Para El Sistema -----	43
2.2.4 Diagramas de Secuencia-----	44
2.2.4.1 Notación del Diagrama de Secuencia-----	45
2.2.4.2 Diagrama De Secuencia Para El Sistema -----	46
<b>Capitulo 3: Implementación -----</b>	<b>47</b>
<b>Introducción -----</b>	<b>47</b>
<b>3.1 Lenguaje de Programación-----</b>	<b>48</b>
<b>3.2 Programación Orientada a Objetos-----</b>	<b>48</b>
<b>3.3 CSharp (C#) -----</b>	<b>50</b>
<b>3.4 Implementación -----</b>	<b>51</b>
<b>Capitulo 4: Pruebas del Sistema -----</b>	<b>57</b>
<b>Introducción -----</b>	<b>57</b>
<b>4.1 Pruebas del Sistema-----</b>	<b>57</b>
<b>Conclusiones -----</b>	<b>63</b>
<b>Bibliografía -----</b>	<b>64</b>

## INTRODUCCIÓN

Actualmente, vivimos en un mundo globalizado donde los avances en ciencia y tecnología en algún tiempo inimaginables se han hecho presentes hoy en día, de una forma tal que es ineludible no utilizarlos en la vida diaria, razón por la cual muchas cosas han cambiado, y con el presente avance en materia de computación y telecomunicaciones, el uso tan difundido de dispositivos móviles y la ideología siempre presente de que la tecnología sirva al hombre para hacer de su vida algo más fácil mediante la automatización de procesos, nos han llevado al planteamiento del problema el cual es crear otros métodos para manejar la información en los hospitales, tomando en cuenta el ejemplo de muchos países de primer mundo.

En México el uso de la tecnología RFID es casi nulo, es por eso que se quiere implementar esta tecnología, para que México empiece a crecer tecnológicamente, científicamente, también que se modernice y se tenga mejor eficiencia en los hospitales.

Dentro del país, algunos hospitales tienen un sistema infructuoso para poder acceder a los archivos clínicos o la información del paciente, a su vez acceder al inventario del material del hospital, entre otros problemas; por eso que surge la necesidad de implementar un sistema que ayude a agilizar la labor de los médicos en cuanto al manejo de la información clínica, por la deficiencia en el manejo de los archivos clínicos surge la idea de realizar un sistema el cual permita acceder a la información de manera fácil y rápida a los médicos, ayudando así a su organización.

El sistema de Base de Datos de Archivos Clínicos utilizando etiquetas RFID pretende facilitar la labor del médico, haciendo que desde cualquier computadora que cuente con este sistema, se pueda consultar información de los pacientes que un médico tiene asignado y a su vez, ver información como lo es su expediente médico, próximas consultas y en el caso de estar internado, se podrá obtener información como la temperatura corporal del paciente, presión en general signos vitales y su evolución y como esta evolucionando con respecto a algunos medicamentos, entre otros datos.

Para la realización del sistema utilizaremos la identificación por radiofrecuencias (RFID) que es una tecnología de identificación remota e inalámbrica en la cual un dispositivo lector vinculado a un equipo de computo, se comunica a través de una antena con una etiqueta mediante ondas de radio, en este caso las etiquetas contendrán la información de cada paciente, esta información se mandará a la base de datos.

# CAPÍTULO I

## RFID (Identificadores por radio frecuencias)

### Introducción

La identificación por radiofrecuencia o RFID por su abreviatura en inglés (radio frequency identification), es una tecnología de identificación remota e inalámbrica en la cual un dispositivo lector vinculado a un equipo de computo, se comunica a través de una antena con una etiqueta mediante ondas de radio.

Esta tecnología que existe desde los años 40, se ha utilizado y se sigue utilizando para múltiples aplicaciones incluyendo casetas de peaje, control de acceso, identificación de ganado y tarjetas electrónicas de transporte.

En los últimos años, la tecnología RFID ha entrado al campo tecnológico gracias a su creciente difusión en aplicaciones de cadena de suministro motivada por las iniciativas de las cadenas de autoservicio y departamentales.

### 1.1 Historia de las Etiquetas RFID

Las raíces de la identificación por radiofrecuencia se remontan a la Segunda Guerra Mundial. Los alemanes, los estadounidenses y británicos usaban radares, los cuales fueron descubiertos en el año de 1935 por el físico Sir Robert Alexander Watson-Watt con el objetivo de identificar que aviones pertenecían al enemigo y quienes regresaban de una misión ver Figura 1.1.1.

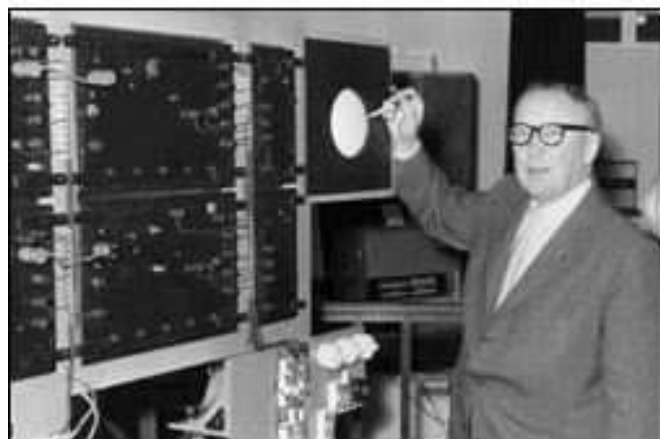


Figura 1.1.1 Alexander Watson Watt con el primer radar.

Los alemanes descubrieron que si los pilotos cambiaban los laminados de sus aviones cuando regresaban a la base, cambiaria de regreso la señal de radio que reflejaba. Este método alertaba a la gente que vigilaba el radar, de esta manera se creo el primer sistema RFID.

Los británicos desarrollaron un sistema que contenía un identificador activo, ellos pusieron un transmisor en cada uno de sus aeroplanos, cuando se recibía la señal del las estaciones de radio, empezaba a mandar la señal de vuelta y esta señal los identificaba como amistosos, la base de los RFIDs trabaja utilizando el mismo concepto, una señal es enviada a un transponder o transmisor que se despierta y refleja la señal (con un sistema pasivo) o emite una señal (con un sistema activo) [1].

### 1.2.1 Sistema RFID

Un sistema RFID es una colección de componentes que implementan etiquetas RFID, este sistema de manera general se compone de dos partes (Ver Figura 1.2.1):

- ❖ Un lector con el hardware y software adecuados para interpretar los datos.
- ❖ Un transponder o también llamado transmisor programado con la información necesaria, que se encuentra en el objeto que será identificado.

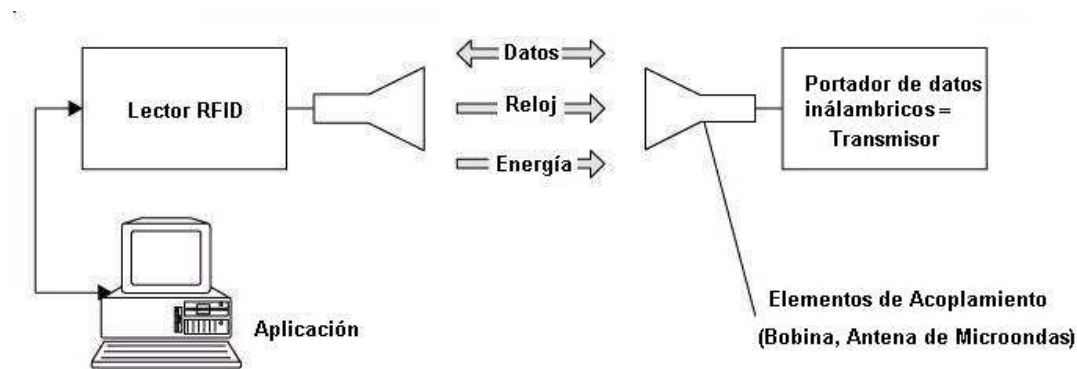


Figura 1.2.1 Componentes Principales de un sistema RFID

Un lector usualmente contiene un módulo de radiofrecuencia (transmisor y receptor), una unidad de control, un elemento de acoplamiento para el transponder, en algunos casos algunos lectores traen una interfaz adicional para permitirles enviar la información recibida a algún otro sistema. El transponder es el que representa la información actual de un dispositivo RFID, el cual consiste normalmente de un elemento de acoplamiento y un microchip electrónico como se observa en la

Figura 1.2.2, cuando el transponder no posee su propio suministro de voltaje y no se encuentra dentro de la zona del lector, se vuelve pasivo, el transponder solo se activa cuando el lector esta dentro de la misma zona que el transponder. La energía que necesita el transponder para activarse es dada por el elemento de acoplamiento, como lo son el pulso y la información [2].

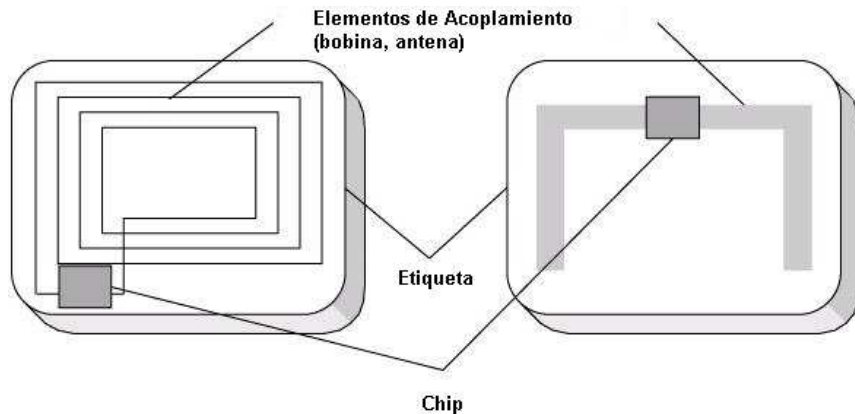


Figura 1.2.2 Elementos de Acoplamiento (bobina, antena) y Microchip electrónico

Por otro lado de manera más específica tenemos los siguientes componentes (ver Figura 1.2.3):

- ❏ Etiqueta RFID (RFID Tag): Es un componente para transmitir y almacenar información muy importante en un sistema RFID.
- ❏ Lector (Reader): También llamado interrogador, es un dispositivo que puede leer y escribir información de las etiquetas, es muy importante en un sistema RFID.
- ❏ Antena Lectora (Reader Antenna): La antena lectora es muy importante para el sistema ya que comunica la etiqueta con las otras antenas lectoras.
- ❏ Controlador (Controller): es un agente intermediario, los lectores actuales poseen controladores integrados.
- ❏ Sensor, Actor y Anunciador (Sensor, Actuator and Annunciator): en este caso, solo si el sistema lo necesita se pueden utilizar para entradas y salidas externas.
- ❏ Dominio y Software del Sistema (Host y Software System): Un sistema RFID puede funcionar independientemente de este componente pero es necesario para poder crear aplicaciones RFID.

☞ **Infraestructura de Comunicación (Communication Infrastructure):** Se trata de los componentes de red, ya sea alámbrica e inalámbrica y una infraestructura serial para que haya una comunicación efectiva en el sistema. [5]

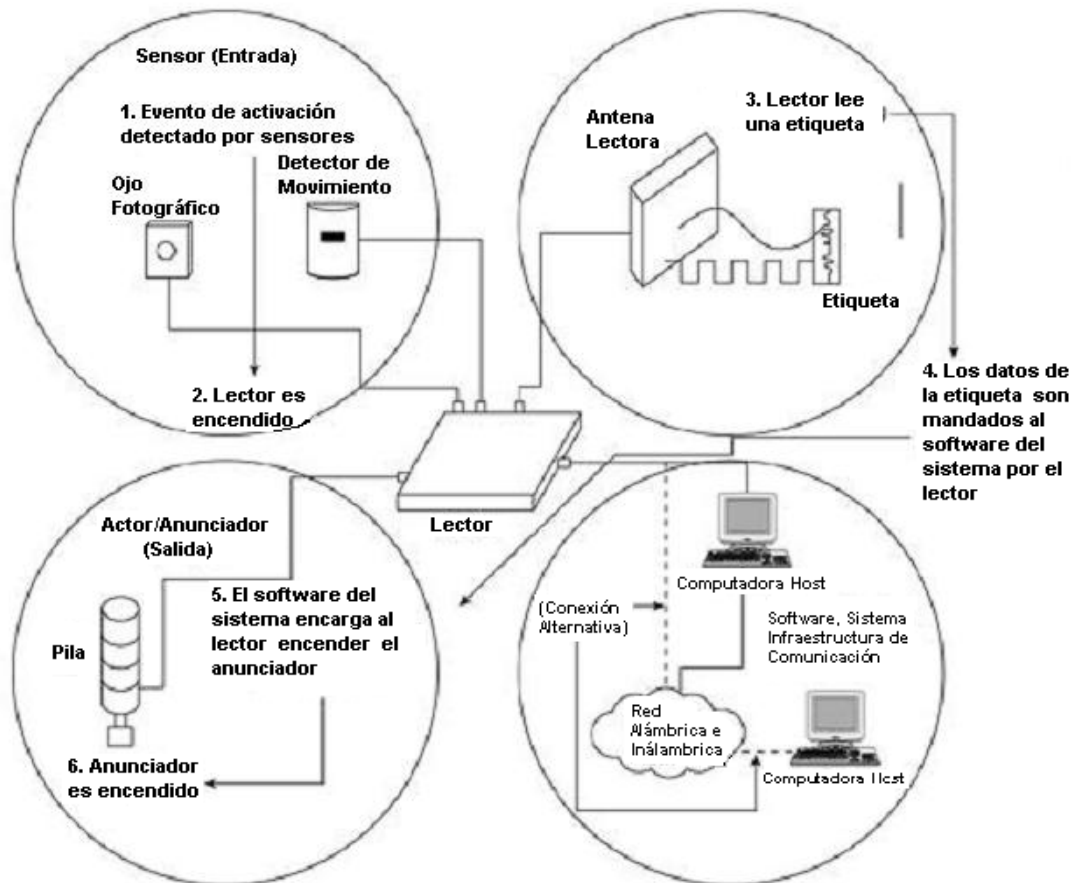


Figura 1.2.3 Diagrama de un sistema RFID y sus componentes

Un sistema RFID nos debe proveer las siguientes características [4]:

1. La habilidad de codificar etiquetas RFID
2. La habilidad de adjuntar las etiquetas RFID a los objetos
3. La habilidad para rastrear el movimiento de los objetos con etiquetas
4. La habilidad de integrar la información de RFID en los negocios
5. La habilidad de producir información que pueda ser compartida
6. La habilidad de desarrollar dispositivos de organización inteligente

### 1.2.2 Etiquetas RFID

Las etiquetas RFID son pequeños circuitos electrónicos, que se utilizan para transmitir la información de manera inalámbrica al lector usando radioondas, este circuito electrónico alberga un pequeño microcontrolador asociado a un circuito emisor-receptor de Radiofrecuencia como podemos ver en la Figura 1.2.4. Se graba mediante láser el identificador numérico único para cada transponder en el interior del microcontrolador. Estas etiquetas permiten conocer datos o información de la persona que lo posea con el simple hecho de tener un lector de las frecuencias.

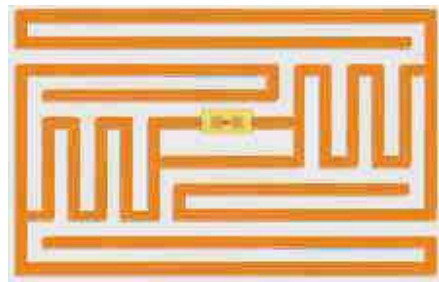


Figura 1.2.4 Una etiqueta RFID EPC (Código de Producto Electrónico)

Las etiquetas RFID contienen una pequeña antena emisora que puede ser activa o pasiva, esto es, en el caso de que sea pasiva permanecerá inactiva hasta que se le solicite la información. La información que almacena debe ser leída con un receptor adecuado. Un dispositivo RFID proporciona la misma funcionalidad que un código de barras o una banda magnética, la posibilidad de identificar unívocamente un objeto.

### 1.2.3 Tipos de Etiquetas

❖ **Etiquetas Pasivas:** reciben la energía necesaria del propio campo generado por la antena, y por lo tanto no necesitan baterías de alimentación. En consecuencia, ni su tamaño, ni su ciclo de vida esta condicionado por la vida de las baterías, es decir tiene vida ilimitada.

Una etiqueta pasiva es más pequeña que una etiqueta activa o semi-activa, en cuanto a la comunicación etiqueta pasiva – lector, el lector siempre se comunica primero seguido por la etiqueta, los componentes más importantes de una etiqueta pasiva como podemos apreciar en la Figura 1.2.5 son: Microchip y una Antena.

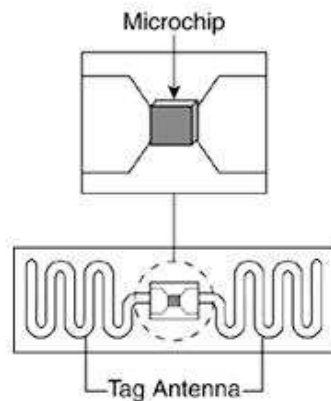


Figura 1.2.5 Componentes de una Etiqueta RFID Pasiva

El microchip o también llamado circuito integrado (CI) está hecho de una placa de silicio pequeña en la que se encuentran miles de dispositivos electrónicos interconectados (diodos, transistores, resistencia, capacitores, entre otros).

La antena se utiliza para aprovechar la energía de la señal del lector para proveer energía a la etiqueta para recibir y enviar información del lector, esta antena está físicamente integrada al microchip, la antena es el centro de operaciones de la etiqueta, la longitud de la antena depende mucho de las operaciones de la etiqueta y la longitud de onda, es muy importante mantener en buen estado la antena ya que si estos puntos de conexión son dañados, la etiqueta puede no funcionar o no puede tener un buen desempeño.

❖ **Etiquetas Activas:** se caracterizan por tener su propia fuente de energía, en forma de baterías (ver figura 1.2.6). Pueden comunicarse de forma independiente, sin necesidad de estar alimentados por un campo electromagnético, permitiendo un radio de alcance mayor que los pasivos. La dependencia de un suministro de energía local, limita sus expectativas de vida, no alcanzando más allá de unos 5-10 años, dependiendo de las baterías y el uso que se dé, así como de las condiciones del entorno de trabajo y de la temperatura exterior, puesto que ésta afecta al comportamiento de las baterías. Al mismo tiempo las baterías y la necesidad de más componentes electrónicos, incrementa el precio por unidad; haciendo que únicamente se utilice cuando la aplicación realmente lo necesite.

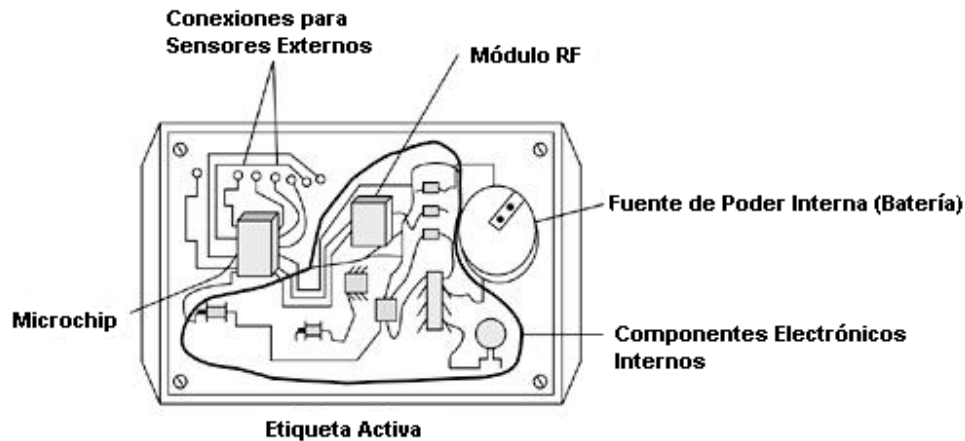


Figura 1.2.6 Componentes de una Etiqueta RFID Activa

Una etiqueta activa usa su fuente de energía para enviar y recibir la información, no necesita que el lector emita energía para la transmisión de los datos, en la comunicación etiqueta activa – lector, a diferencia con una pasiva, la etiqueta activa siempre se comunica primero seguido del lector ya que la presencia del lector no es necesaria para la transmisión de información, una etiqueta activa puede emitir la información a sus alrededores aun sin tener un lector, este tipo de etiqueta se le llama también transmisor porque transmite la información continuamente con o sin el lector, otros tipos de etiquetas activas pueden entrar en espera o en ahorro de energía cuando no esta el lector presente, cuando una etiqueta es transmisora/receptora se le llama transponder, las etiquetas activas contienen los mismos componente que las etiquetas pasivas , contienen componentes electrónicos internos, además de manejar su propia fuente de energía.

Los microchips que utilizan las etiquetas activas tienen mayor capacidad que los que utilizan las pasivas, en cuanto a la antena funciona de manera similar a las pasivas con la diferencia que puede transmitir las señales de las etiquetas y recibir una señal en respuesta al lector.

Los componentes electrónicos que contiene hacen que una etiqueta pueda actuar como transmisor y ejecutar tareas específicas, esto hace que las etiquetas activas puedan aplicarse a un rango mucho más amplio de aplicaciones que las pasivas.

La fuente de energía que contiene una etiqueta generalmente es una batería que provee la energía a la etiqueta y transmitir la información, cuando la fuente de energía se descarga, la etiqueta deja de transmitir los mensajes.

❖ Etiquetas semi-activas o semi-pasivas: estas etiquetas contienen una fuente de energía integrada y componentes electrónicos que realizan una tarea en específico ver Figura 1.2.7.

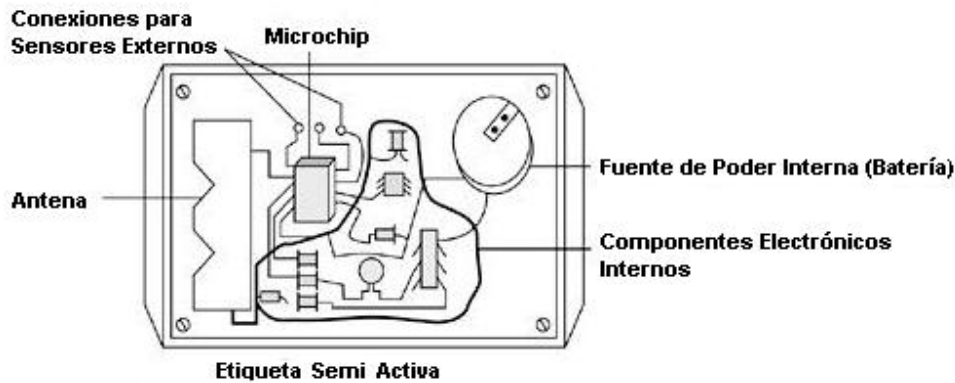


Figura 1.2.7 Componentes de una Etiqueta RFID Semi – Activa (Semi - Pasiva)

La batería provee a la etiqueta con energía para que ejecute sus operaciones designadas pero para transmitir utiliza la energía que provee el lector, en este tipo de comunicación el primero en comunicarse es el lector seguido de la etiqueta, se puede leer de una distancia mas larga si lo comparamos con un tag pasivo, es más rápido ya que no se necesita el tiempo de espera para que inicie la transmisión, la máxima distancia que una etiqueta semi activa puede tener es de 30 metros aproximadamente bajo condiciones ideales.

Cabe mencionar que las etiquetas dependiendo de su capacidad pueden ser de lectura (RO – Read Only), única escritura – lectura constante (WORM – write once, read many) y lectura – escritura (RW – Read Write).

#### 1.2.4 Funcionamiento de las etiquetas RFID

El procedimiento de radiofrecuencia se basa en circuitos resonadores que se ajustan a una resonancia definida, primeras versiones eran resistores inductivos hechos con madera y barnizados con cobre en un capacitor con cubierta de plástico. Ahora los sistemas modernos utilizan una bobina entre cada lámina.

El lector o antena, suele montarse fijo y es el transponder el que se encuentra en movimiento y entra en el campo de lectura de la antena ver Figura 1.2.8.

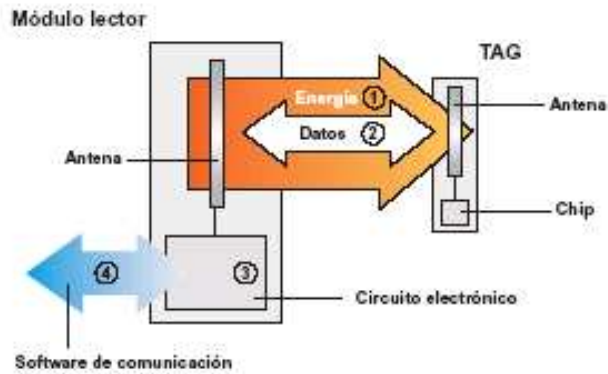


Figura 1.2.8 Funcionamiento de los sistemas RFID

El lector proporciona energía al transponder generando un campo electromagnético mediante un circuito resonante en serie, el transponder recibe esta energía y mediante su circuito resonante en paralelo, la transforma en energía eléctrica que le permite “despertar” ver Figura 1.2.9. Después, modifica el propio campo electromagnético generado por el lector, para codificar la información de respuesta necesaria. Del mismo modo, el lector también puede codificar información en el pulso de energía generado, mediante la modulación de la señal. El tamaño y forma de la antena que emite y recibe el campo electromagnético también es muy importante, puesto que proporciona el medio de comunicación con el transponder, y en función de la distancia de lectura, del tipo de etiqueta y las condiciones de lectura, deberá prestarse especial atención a su diseño. La señal recibida por el lector desde la tarjeta está a un nivel de -60 db por debajo de la portadora de transmisión.

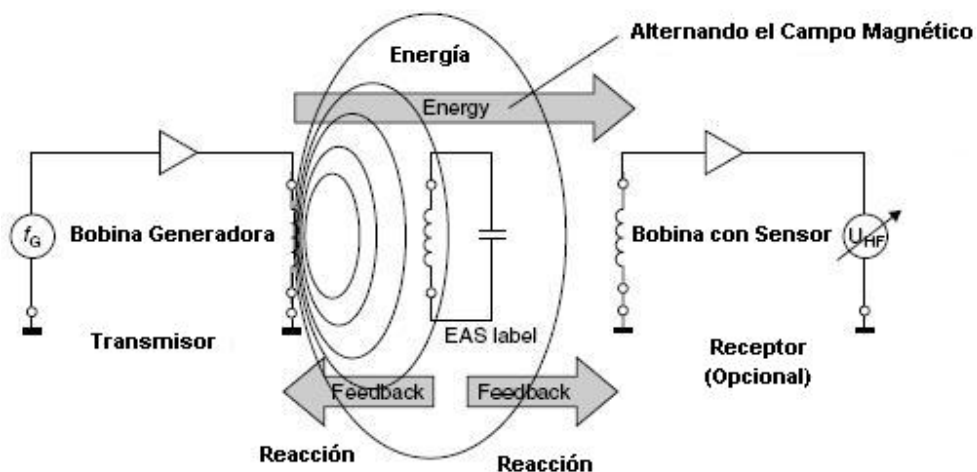


Figura 1.2.9 Principio de Operación de los Procedimientos de Radiofrecuencia

Podemos encontrar además dos tipos de lectores diferentes:

- ☒ Sistemas con bobina simple, la misma bobina sirve para transmitir la energía y los datos. Son más simples y más baratos, pero tienen menos alcance.
- ☒ Sistemas interrogadores con dos bobinas, una para transmitir energía y otra para transmitir datos. Son más caros, pero consiguen unas prestaciones mayores [2].

### 1.2.5 Lectores

Un lector de RFID también llamado interrogador es un dispositivo que lee la información escrita en las etiquetas RFID, el tiempo que dura transmitiendo la energía para leer etiquetas se le llama duty cycle del lector, un lector es el sistema central del sistema de RFID.

En general el funcionamiento del lector es cuando existe una antena cerca de una etiqueta, esta recoge la energía y la etiqueta se electriza por medio de inducción, la energía es lo suficientemente potente como para encender la antena interna que contiene, la cual guarda el número único de la etiqueta, la mayor parte de las etiquetas RFIDS funcionan de esa manera.

Los componentes de un RFID son: transmisores, receptores, microprocesador, memoria, controlador, fuente de energía y una interfaz de comunicación ver Figura 1.2.10.

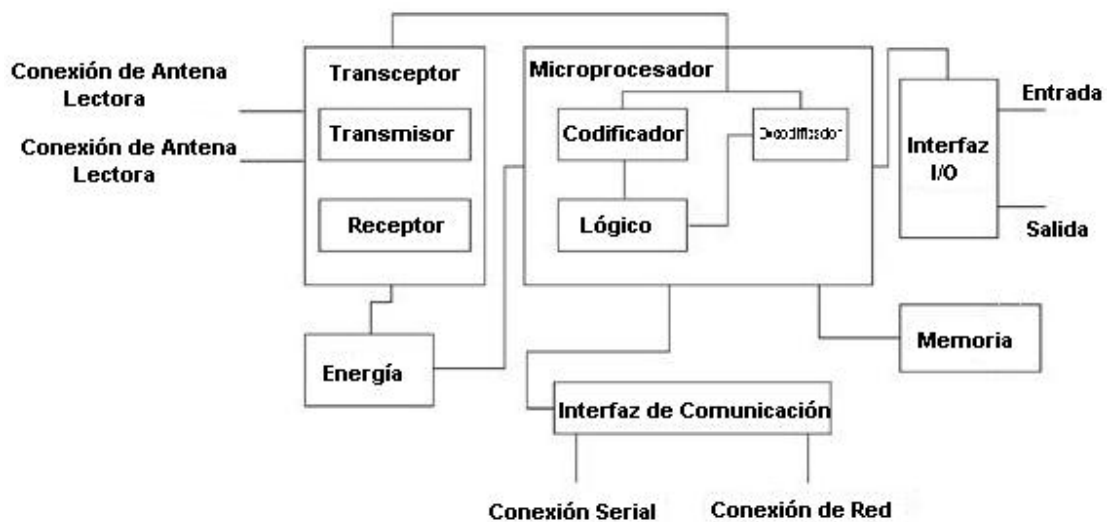


Figura 1.2.10 Componentes de un lector RFID

El lector puede tener diferentes tamaños y formas, se pueden encontrar lectores fijos o portables, a su vez un lector tiene varios subsistemas entre los cuales se encuentran:

- ☞ El lector API que es la interfaz de programación que permite a los programas manejar y registrar los eventos ocurridos en las etiquetas RFID, también provee algunas configuraciones
- ☞ Las comunicaciones manejan las funciones de la red y están conectadas entre si a toda la red.
- ☞ El manejo de eventos ocurre se observa la comunicación entre la etiqueta y el lector, a esto se le llama observación, si esta difiere se le llama evento, entonces nuestro manejador de eventos va a determinar cuales son eventos y cuales son errores y en el momento enviará reportes de error.
- ☞ La antena del subsistema consiste en una o más antenas y las interfaces que soportan y habilitan las etiquetas lectoras e interrogadoras. [4]

### 1.3 Frecuencias Utilizadas

La frecuencia es una medida para indicar el número de repeticiones de cualquier fenómeno o suceso periódico en la unidad de tiempo. Para poder calcular la frecuencia de un evento, se contabilizan un número de ocurrencias de este teniendo en cuenta un intervalo temporal, luego estas repeticiones se dividen por el tiempo transcurrido en la figura 1.3.1 se pueden observar diferentes ondas con diferente frecuencia.

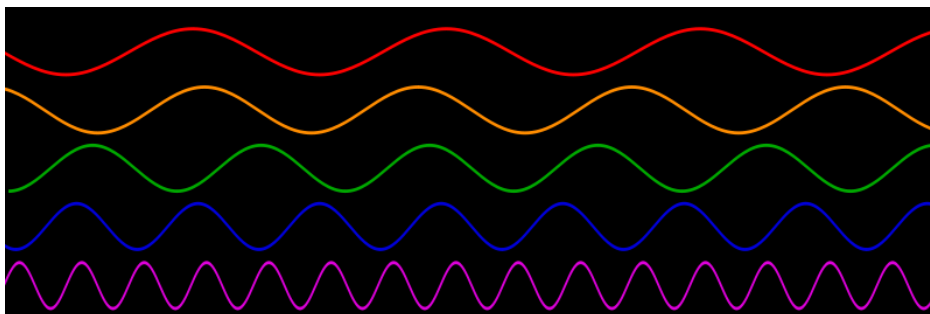


Figura 1.3.1 Ondas con diferente frecuencia (LF, HF)

Los transmisores y lectores RFID operan a varias frecuencias. Cada rango de frecuencias tiene ventajas e desventajas, que la hacen más eficaz para una determinada aplicación. Originalmente, cuanto más baja es la frecuencia, más baja es la velocidad de lectura de los datos contenidos en la etiqueta y mejor es la lectura del transponder cerca o a través de superficies metálicas o líquidos, que suelen interferir con las ondas de radio.

Las cuatro frecuencias más utilizadas en la actualidad son (ver figura 1.3.2):

❖ **Baja Frecuencia (LF).** Estos transmisores funcionan habitualmente a 125 y 134,2 Khz. y típicamente proporcionan un rango de lectura inferior a 50 cm. Son los menos sensibles a las interferencias, y se usan en controles de acceso, el rastreo de animales, gestión de residuos y diversas aplicaciones de punto de venta. Puesto que el espectro de bajas frecuencias está disponible globalmente y debido a sus ventajas, este tipo de etiquetas es un estándar de para aplicaciones industriales y de rastreo de animales.

❖ **Alta Frecuencia (HF).** Estos funcionan a la frecuencia de 13,56 MHz, tienen el mismo rango de lectura que los de LF, pero pueden transmitir sus datos más rápido. Se usan habitualmente para tarjetas inteligentes (smart cards) y etiquetas autoadhesivas adheridas a productos comerciales. Por contra son mucho más sensibles a entornos adversos, tanto por ruidos como por presencia de metales, por lo que su campo de aplicación queda limitado a aquellos sistemas considerados limpios: controles de acceso, logística en almacenes.

❖ **Ultra Alta Frecuencia (UHF).** Ellos funcionan normalmente en el rango que se extiende desde los 860 MHz hasta los 930 MHz, tienen un alcance mayor que sus competidores (entre 2 y 10 m.) y permiten mayores velocidades de transmisión de los datos. Su punto débil es que no pueden ser leídos a través de materiales con un alto contenido líquido o metálico, el campo es más direccional, y presenta problemáticas de rebotes. Actualmente, el espectro de UHF no está disponible en todo el mundo con las mismas frecuencias y potencias de emisión.

❖ **Microondas.** Estos transmisores, generalmente operan en 2,45 GHz ó 5,8 GHz y tienen una velocidad de comunicación mayor que sus competidores. Actualmente no tienen unas áreas de aplicación definidas con claridad, puesto que se usan para sustituir a sistemas basados en etiquetas UHF mayoritariamente. Debido a las diferentes reglamentaciones en materia de niveles de emisión, presentes en el mundo, resultará difícil que se convierta en una frecuencia global.

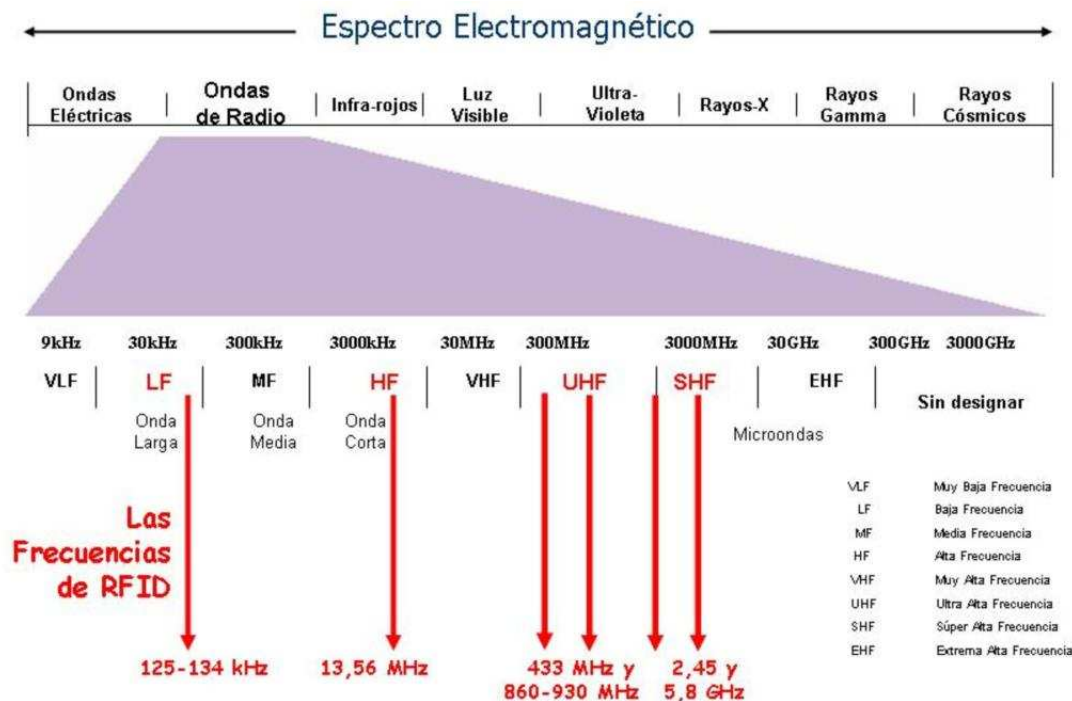


Figura 1.3.2 Diferentes Frecuencias y sus Espectros utilizadas en los RFID

En general la frecuencia que se utilizará con las etiquetas RFID dependerá mucho de la aplicación, por ejemplo en la frecuencia baja se pueden encontrar la identificación de libros, autos, entre otras cosas; la frecuencia alta a diferencia de la baja no funciona cerca de los metales y se usa normalmente para el rastreo de equipaje o productos, la frecuencia ultra alta y las microondas generalmente se usan en etiquetas activas.

#### 1.4 Tipos de Antena

El tipo de antena utilizado en una etiqueta depende de la aplicación objetivo y la frecuencia de operación.

Para baja frecuencia (LF) normalmente utiliza la inducción electromagnética. Como el voltaje inducido es proporcional a la frecuencia, se puede producir el necesario para alimentar un circuito integrado dado un número suficiente de espiras. Existen etiquetas de baja frecuencia que utilizan una antena en varios niveles (tres de 100-150 espiras cada uno) alrededor de un núcleo de ferrita.

En alta frecuencia (HF, 13,56 MHz) se utiliza una espiral plana con 5-7 vueltas y un factor de forma parecido al de una tarjeta de crédito para lograr distancias de decenas de centímetros. Estas antenas son más baratas que las LF ya que pueden producirse por medio de litografía en lugar de

espiración, aunque son necesarias dos superficies de metal y una aislante para realizar la conexión cruzada del nivel exterior al interior de la espiral, donde se encuentran el condensador de resonancia y el circuito integrado.

Las etiquetas pasivas de ultra alta frecuencia (UHF) y de microondas suelen acoplarse por radio a la antena del lector y utilizar antenas clásicas de dipolo. Sólo es necesaria una capa de metal, lo que reduce el coste. También pueden usarse estructuras de banda ancha. La ganancia de las antenas compactas suele ser menor que la de un dipolo (menos de 2 dBi) y pueden considerarse isótropas en el plano perpendicular a su eje.

Las antenas HF y UHF suelen ser de cobre o aluminio. Se han probado tintas conductoras en algunas antenas encontrando problemas con la adhesión al circuito integrado y la estabilidad del entorno.

### **1.5 Ventajas y Desventajas**

La ventaja significativa de los dispositivos RFID sobre el resto de sistemas mencionados anteriormente, es que no necesitan ser posicionados respecto al lector de manera precisa, ni ser deslizados por ranuras de lectura, ni hay que sufrir incomodidades para poder iluminar con el haz de lectura correctamente. De esta manera, el encapsulado del dispositivo lo hace resistente al entorno para el cual se haya diseñado. Las etiquetas RFID también pueden leerse muy rápido: en la mayoría de los casos responden en menos de 100 milisegundos.

Algunas de las ventajas de las etiquetas RFID son estas:

- ❑ La lectura es 100% precisa aunque no es considerada completamente como una ventaja.
- ❑ Permite leer múltiples etiquetas simultáneamente de forma automática.
- ❑ Tiene un código único, fijado en fábrica o escrito a distancia.
- ❑ Identifican cada producto de forma individual.
- ❑ Pueden contener información sobre el producto.
- ❑ Resistentes a la humedad y temperatura
- ❑ Pueden realizar tareas inteligentes, por ejemplo condiciones meteorológicas
- ❑ Amplia capacidad de rango ya que puede guardar pocos bytes o casi cualquier cantidad de información
- ❑ Puede alcanzar diversos rangos, desde pocos centímetros hasta Km.
- ❑ No necesitan tener contacto el lector y la etiqueta para la transmisión de datos

- ❏ La información puede rescrita muchas veces
- ❏ No necesita estar a la vista la etiqueta y el lector, esto permite que se pueda leer desde cualquier lugar, siempre cuando haya cobertura.

Algunas desventajas o limitaciones de los RFIDS son:

- ❏ Desempeño pobre con objetos opacos RF y objetos absorbentes RF, esto depende mucho del comportamiento de la frecuencia, la tecnología actual, no trabaja bien con estos materiales y en algunos casos falla.
- ❏ Impacto por factores ambientales, las condiciones del ambiente afectan a los sistemas RFID.
- ❏ Limitación en la lectura de etiquetas y el tiempo de lectura.
- ❏ La interferencia del hardware puede tener un impacto negativo si la configuración del hardware no se hace correctamente, por ejemplo una antena mal colocada
- ❏ La energía que obtiene la etiqueta depende mucho de la orientación de la antena y la distancia, esto afectara en la profundidad que la energía pueda alcanzar en una etiqueta.
- ❏ Tecnología inmadura, aun esta en desarrollo la tecnología, debido a su rápida evolución los cambios pueden afectar a algunos sistemas.[3]

IBM y la Asociación Europea para la Identificación Segura IDtrack hicieron un informe sobre algunas empresas que utilizan tecnología RFID, en la Figura 1.5.1 podemos observar algunas ventas y limitaciones que estas compañías encontraron.

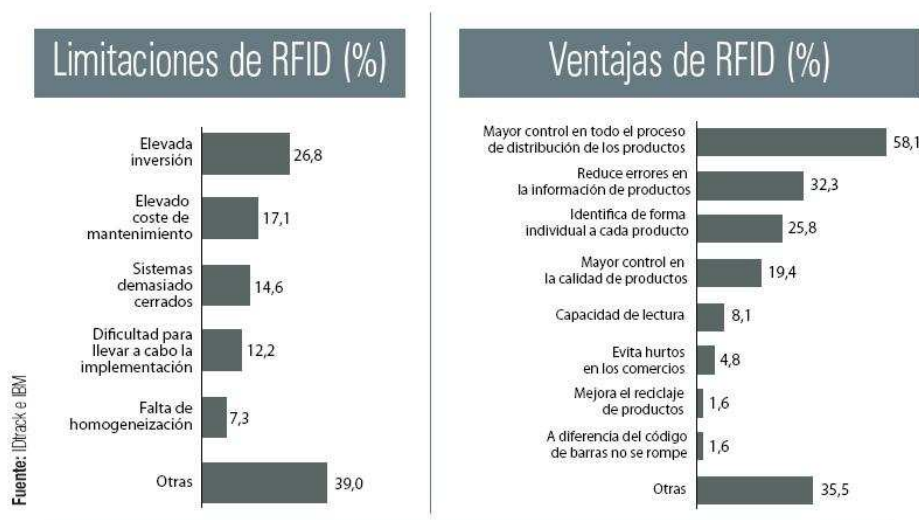


Figura 1.5.1 Ventajas y Limitaciones según los estudios de IBM

## **1.6 RFID Middleware**

El middleware de los sistemas RFID es un elemento importante ya que provee conectividad con los lectores mientras se encapsula la aplicación de la interfaz del dispositivo y las interconexiones, para bajar el tamaño de la información que utilizan las aplicaciones, los RFID necesitan filtrar las observaciones RFID provenientes de los lectores, una observación es cuando la antena busca y hace contacto con la etiqueta., para proveer una aplicación para el manejo de las observaciones.

Muchas implementaciones para los middleware de RFID, los eventos de aplicación de nivel (ALE) proveen una interfaz estándar para que los clientes obtengan la información ya filtrada de diversas fuentes como los lectores RFID u otros sensores.

## **1.7 Aplicaciones Actuales de las Etiquetas RFID**

La tecnología de Identificación por Radiofrecuencia (RFID) es un método de identificación automática sin contacto, es la tecnología más nueva y de más rápido crecimiento en el segmento de identificación automática en la industria. RFID permite identificación automática, localización y monitoreo de personas, objetos y animales en una infinidad de aplicaciones que van desde simple inventario hasta sistemas complejos de casetas de cobro en carreteras.

La tecnología RFID ha revolucionado la industria de la identificación automática ofreciendo avances significativos en comparación con sistemas tradicionales como código de barras, tarjetas de banda magnéticas y chips de contacto o proximidad. RFID ha resuelto problemas que anteriormente se pensaba que no podían tener solución.

Actualmente son muchos los sectores industriales que pueden beneficiarse de las ventajas de la tecnología de auto-identificación por radiofrecuencia. Algunas de sus aplicaciones de las etiquetas RFID son las siguientes:

- ❑ Control de calidad, producción y distribución.
- ❑ Localización y seguimiento de objetos.
- ❑ Control de accesos (ver figura 1.7.1).
- ❑ Identificación de materiales.
- ❑ Control de fechas de caducidad.
- ❑ Detección de falsificaciones.

- ☒ Almacenaje de datos.
- ☒ Automatización de los procesos de fabricación.
- ☒ Información al consumidor.
- ☒ Reducción de tiempo y coste de fabricación.
- ☒ Reducción de colas a la hora de pasar por caja.
- ☒ Identificación y localización de animales perdidos.
- ☒ Elaboración de censos de animales.
- ☒ Identificación y control de equipajes en los aeropuertos.
- ☒ Inventario automático.
- ☒ Medicina.



Figura 1.7.1 Un sistema RFID de control de acceso implementado en un peaje

Actualmente la tecnología RFID avanza muy rápido y los productos disponibles dentro del rango de las aplicaciones RFID se continúan expandiendo, algunos tipos de aplicaciones están ya bien hechos y se pueden usar comercialmente, otras aplicaciones están en proceso y tienen un prototipo ya creado. Muchos de los prototipos en proceso pueden ser potencialmente una aplicación para utilizarse comercialmente.

## 1.8 Aplicación de las etiquetas RFID en la Medicina

Los hospitales han implementado la tecnología de identificación inalámbrica (RFID), el objetivo es hacer más eficientes sus procesos de trabajo, reducir la pérdida de existencias caras, como instrumental quirúrgico y aparatos médicos, o proteger a pacientes con los que se corre el peligro de que se escapen de la clínica (ver figura 1.8.1).

En cuanto a la sanidad aún predominan proyectos RFID en estado piloto, ya que bajo el mandato del ahorro hay que tener en cuenta cada inversión. “La RFID no es un fin absoluto en sí. En realidad, la tecnología debe optimizar los procesos logísticos y de asistencia médica del hospital”, comenta Nils Lipprandt, Business Development Manager de TSystems [6]. Por este motivo, recomienda conceptos que consideren las aplicaciones RFID compaginadas con la construcción de otras infraestructuras. Por ejemplo, si una clínica planea establecer redes locales inalámbricas (WLAN) para utilizar ordenadores portátiles en las visitas o accesos móviles a Internet para los pacientes, es posible aplicar un proyecto RFID con un gasto que este al alcance del capital que se tenga.



Figura 1.8.1 Un sistema RFID implementado en la medicina

## **CAPÍTULO II**

### **Análisis y Diseño**

#### **Introducción**

Hace algunos años, los programadores no realizaban análisis profundos sobre el problema a resolver, con el tiempo han existido grandes cambios en aquellos sistemas, los cuales eran solamente algorítmicos donde todo el esfuerzo del desarrollo se centraba en la escritura de programas que realizaran algún tipo de cálculo.

Actualmente es necesario contar con el análisis y el diseño del sistema, es decir organizar el proceso del diseño de tal forma que los programadores, analistas y personas involucradas en el desarrollo del sistema lo entiendan, para poder realizar esto se utiliza el UML.

El UML (Unified Modeling Language) en español Lenguaje Unificado de Modelado es un modelo que une tres metodologías más difundidas (OMT, Bootch y OOSE), este lenguaje proporciona las herramientas necesarias para poder obtener el diseño del software, UML abarca todas las fases del ciclo de vida de un proyecto, soporta diferentes maneras de visualización dependiendo de quién tenga que interpretar. [7].

El Lenguaje UML está compuesto por diversos elementos gráficos que se unen para poder crear diagramas, en este capítulo, analizaremos el sistema con diversos diagramas UML para poder diseñar el sistema.

En este trabajo incluiremos bases de datos, las bases de datos es son un instrumento muy utilizado, en estos tiempos, el hecho que seamos programadores, analistas, informáticos, o cualquier otro título profesional que tenga que ver con el diseño, implica que nos tengamos que relacionar en algún momento de nuestras vidas con el diseño de Bases de Datos, en este capítulo también se hablará del diseño y modelado de las Bases de Datos.

#### **2.1 Bases de Datos**

Las Bases de Datos, son ahora un elemento indispensable en nuestra vida, es muy común encontrarnos con actividades que tengan que ver con Bases de Datos.

Un sistema de base de datos es esencialmente un sistema informatizado cuya finalidad general es recolectar información y permitir a los usuarios recuperar y actualizar esa información por medio de peticiones. La información en cuestión puede ser cualquier cosa que sea de importancia para

el individuo u organización; en otras palabras, todo lo que sea necesario para ayudarle en el proceso de su administración.

Una base de datos proporciona a los usuarios el acceso a datos, que pueden visualizar, ingresar o actualizar, en concordancia con los derechos de acceso que se les hayan otorgado. Se convierte más útil a medida que la cantidad de datos almacenados crece.

Una base de datos puede ser local, es decir que puede utilizarla sólo un usuario en un equipo, o puede ser distribuida, es decir que la información se almacena en equipos remotos y se puede acceder a ella a través de una red.

La principal ventaja de utilizar bases de datos es que múltiples usuarios pueden acceder a ellas al mismo tiempo.

Para poder administrar las bases de datos es necesario un sistema de administración para controlar tanto los datos como los usuarios. La administración de bases de datos se realiza con un sistema llamado DBMS (Database management system [Sistema de administración de bases de datos]). El DBMS es un conjunto de servicios (aplicaciones de software) para administrar bases de datos, que permite un fácil acceso a los datos, a su vez el acceso a la información por parte de múltiples usuarios y la manipulación de los datos encontrados en la base de datos (insertar, eliminar, editar) ver figura 2.1.1.

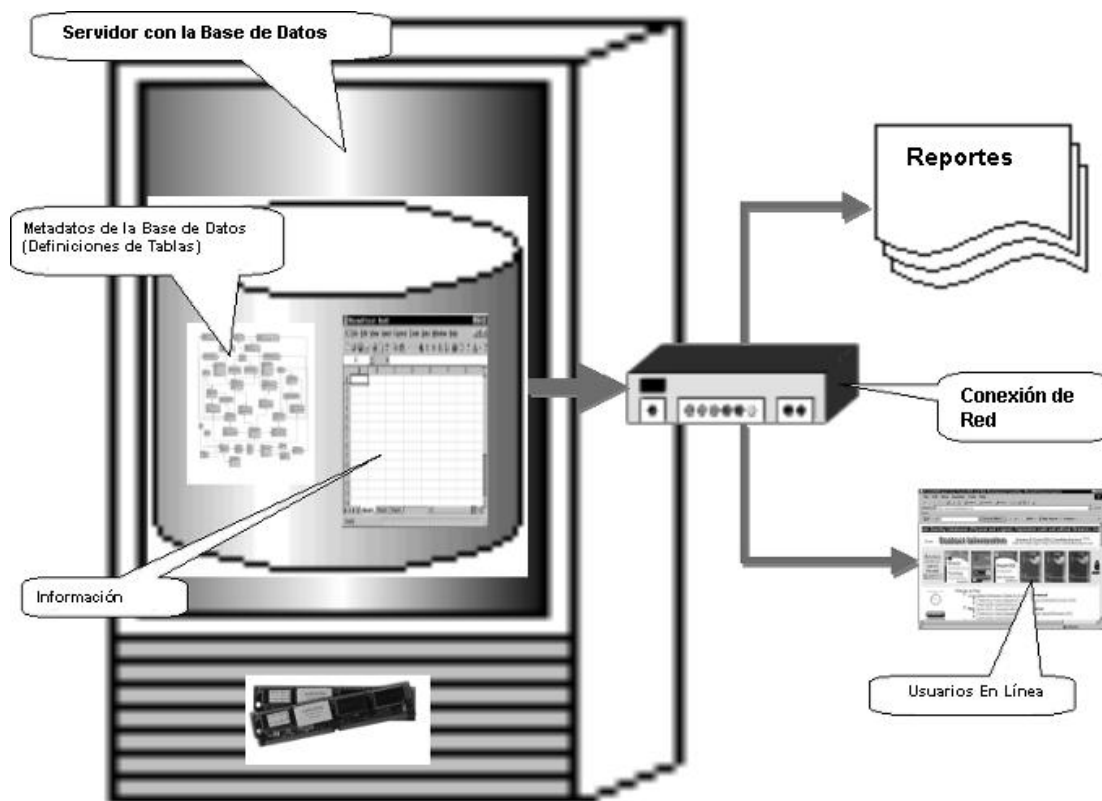


Figura 2.1.1 Esquema General del Funcionamiento de una Base de Datos.

Los sistemas de base de datos centralizados son aquellos que se ejecutan en un único servidor, en el cual las otras computadoras solo pueden acceder a su información, siempre y cuando el servidor este en funcionamiento. Tales sistemas comprenden el rango desde los sistemas de bases de datos monousuarios ejecutándose en computadoras personales hasta los sistemas de base de datos de alto rendimiento ejecutándose en grandes sistemas [8].

### 2.1.1 Modelado de Bases de Datos

El Modelado es la etapa en donde identificamos y “dibujamos” los conjuntos de datos que el Usuario requiere en un Sistema de información. El Diseño Lógico es la etapa donde transformamos ese modelo en un diseño relacional (asumiendo que es el más utilizado), independiente de la herramienta (DBMS) que vayamos a utilizar. El Diseño Físico es aquel en donde especificamos características propias del DBMS elegido, como tipos de dato concretos, parámetros por cada tabla, campo, relación, etc. Y la construcción, es en si el proceso de capturar los datos sobre datos para materializar el diseño relacional dentro de nuestro DBMS.

Existen tres tipos de modelado de datos los cuales son:

- Conceptual: muy general y abstracto, visión general del negocio/institución.
- Lógico: versión completa que incluye todos los detalles acerca de los datos.
- Físico: esquema que se implementara en un manejador de bases de datos (DBMS).

En cuanto a los modelos conceptuales existen distintos tipos como lo son:

Basados en registros

- Jerárquico: datos en registros, relacionados con apuntadores y organizados como colecciones de árboles
- Redes: datos en registros relacionados por apuntadores y organizados en gráficas arbitrarias
- Relacional: datos en tablas relacionados por el contenido de ciertas columnas

Basados en objetos

- Orientado a objetos: datos como instancias de objetos (incluyendo sus métodos)
- Entidad-relación: datos organizados en conjuntos interrelacionados de objetos (entidades) con atributos asociados

Generalmente se usa el modelo de Entidad-Relación para el modelado y diseño lógico de bases de datos, en donde a través del reconocimiento de entidades débiles, entidades fuertes, relaciones entre entidades y su semántica se va creando la base de datos, para después, analizarla y poder llegar al diseño lógico. En nuestro caso, utilizaremos el modelo Entidad-Relación.

### **2.1.2 Modelo Entidad – Relación**

El modelo entidad Relación es el modelo conceptual más utilizado para el diseño conceptual de base de datos, fue introducido por Peter Chen en el año de 1976. El modelo entidad relación está formado por un conjunto de conceptos que permiten describir la realidad mediante un conjunto de representaciones gráficas y lingüísticas.

Inicialmente el modelo Entidad Relación, solo incluía los conceptos de entidad relación y atributos, más tarde se añadieron algunos otros conceptos, los cuales veremos mas adelante.

#### **2.1.2.1 Conceptos del Modelo Entidad-Relación**

##### **Entidad**

Cualquier tipo de objeto o concepto sobre el que se recoge información: cosa, persona, concepto abstracto o suceso. Por ejemplo: coches, casas, empleados, clientes, empresas, oficios, diseños de

productos, conciertos, excursiones, etc. Las entidades se representan gráficamente mediante rectángulos y su nombre aparece en el interior. Un nombre de entidad sólo puede aparecer una vez en el esquema conceptual.

Hay dos tipos de entidades: fuertes y débiles. Una entidad débil es una entidad cuya existencia depende de la existencia de otra entidad. Una entidad fuerte es una entidad que no es dependiente de otra entidad.

### **Relación**

Es una correspondencia o asociación entre dos o más entidades. Cada relación tiene un nombre que describe su función. Las relaciones se representan gráficamente mediante rombos y su nombre aparece en el interior.

Las entidades que están involucradas en una determinada relación se denominan entidades participantes. El número de participantes en una relación es lo que se denomina grado de la relación. Por lo tanto, una relación en la que participan dos entidades es una relación binaria; si son tres las entidades participantes, la relación es ternaria; etc.

Una relación recursiva es una relación donde la misma entidad participa más de una vez en la relación con distintos papeles. El nombre de estos papeles es importante para determinar la función de cada participación.

La cardinalidad con la que una entidad participa en una relación especifica el número mínimo y el número máximo de correspondencias en las que puede tomar parte cada ocurrencia de dicha entidad. La participación de una entidad en una relación es obligatoria (total) si la existencia de cada una de sus ocurrencias requiere la existencia de, al menos, una ocurrencia de la otra entidad participante. Si no, la participación es opcional (parcial). Las reglas que definen la cardinalidad de las relaciones son las reglas de negocio.

### **Atributo**

Es una característica de interés o un hecho sobre una entidad o sobre una relación. Los atributos representan las propiedades básicas de las entidades y de las relaciones. Toda la información extensiva es almacenada por los atributos. Gráficamente, se representan mediante bolitas que cuelgan de las entidades o relaciones a las que pertenecen.

Cada atributo tiene un conjunto de valores asociados denominado dominio. El dominio define todos los valores posibles que puede tomar un atributo. Puede haber varios atributos definidos sobre un mismo dominio.

Los atributos pueden ser simples o compuestos. Un atributo simple es un atributo que tiene un solo componente, que no se puede dividir en partes más pequeñas que tengan un significado propio. Un atributo compuesto es un atributo con varios componentes, cada uno con un significado por sí mismo. Un grupo de atributos se representa mediante un atributo compuesto cuando tienen afinidad en cuanto a su significado, o en cuanto a su uso. Un atributo compuesto se representa gráficamente mediante un óvalo.

### **Identificador**

Un identificador de una entidad es un atributo o conjunto de atributos que determina de modo único cada ocurrencia de esa entidad. Un identificador de una entidad debe cumplir dos condiciones:

1. No pueden existir dos ocurrencias de la entidad con el mismo valor del identificador.
2. Si se omite cualquier atributo del identificador, la condición anterior deja de cumplirse.

Toda entidad tiene al menos un identificador y puede tener varios identificadores alternativos. Las relaciones no tienen identificadores [8].

### **Jerarquía de generalización**

Una entidad  $E$  es una generalización de un grupo de entidades  $E^1, E^2, \dots, E^m$ , si cada ocurrencia de cada una de esas entidades es también una ocurrencia de  $E$ . Todas las propiedades de la entidad genérica  $E$  son heredadas por las subentidades.

Cada jerarquía es total o parcial, y exclusiva o superpuesta. Una jerarquía es total si cada ocurrencia de la entidad genérica corresponde al menos con una ocurrencia de alguna subentidad. Es parcial si existe alguna ocurrencia de la entidad genérica que no corresponde con ninguna ocurrencia de ninguna subentidad. Una jerarquía es exclusiva si cada ocurrencia de la entidad genérica corresponde, como mucho, con una ocurrencia de una sola de las subentidades. Es superpuesta si existe alguna ocurrencia de la entidad genérica que corresponde a ocurrencias de dos o más subentidades diferentes.

Un subconjunto es un caso particular de generalización con una sola entidad como subentidad.

Un subconjunto siempre es una jerarquía parcial y exclusiva.

### **2.1.3 Diagrama Entidad-Relación (E-R)**

Denominado por sus siglas como: E-R; Este modelo representa a la realidad a través de un esquema gráfico empleando los terminología de entidades, que son objetos que existen y son

los elementos principales que se identifican en el problema a resolver con el diagramado y se distinguen de otros por sus características particulares denominadas atributos, el enlace que rige la unión de las entidades esta representada por la relación del modelo [9]

### 2.1.3.1 Notación del Diagrama E-R

Anteriormente, ya se había hablado acerca de los conceptos que se deben tomar en cuenta para el modelado, lo cuales son

- ❖ Entidad ( entity )
- ❖ Atributo ( attribute )
- ❖ Dominio ( values set )
- ❖ Relación ( relationship )

En la Figura 2.1.2 podemos observar como se representa cada uno de estos de forma gráfica.

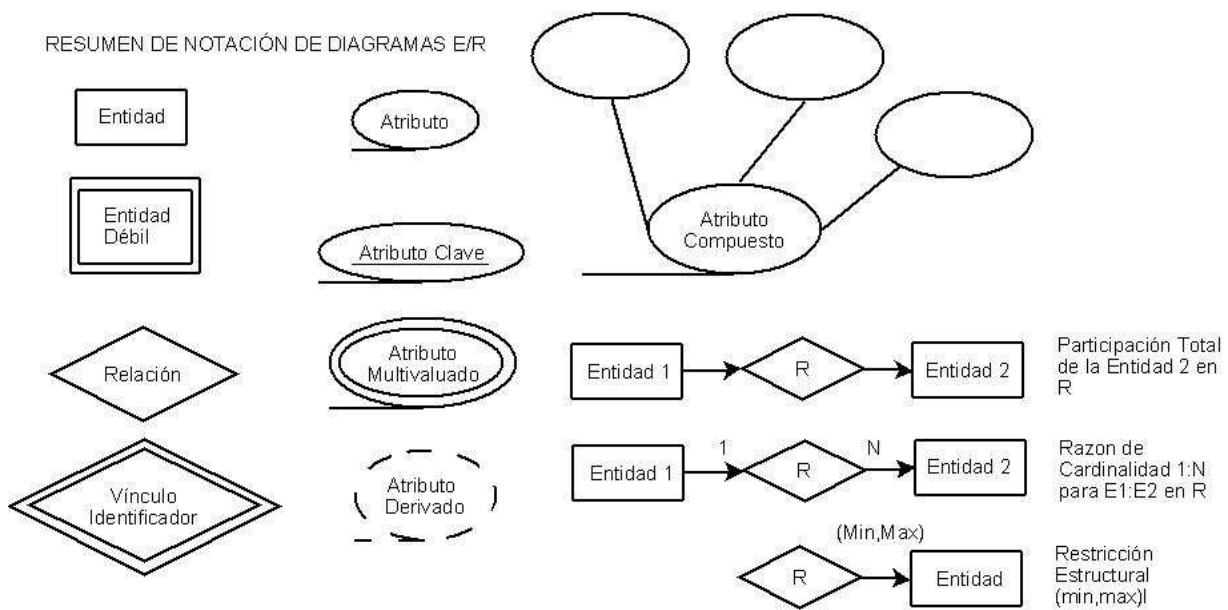


Figura 2.1.2 Notación para diagramas E-R [9]

### 2.1.4 Diagrama Entidad/Relación Para El Sistema De Base De Datos De Archivos Clínicos

Cabe recordar que el diagrama Entidad relación es una herramienta de modelado de sistemas, que se concentra en los datos almacenados en el sistema y las relaciones entre éstos.

El diagrama Entidad Relación (E-R) para el Sistema de Base de Datos de Archivos que Clínicos se muestra en la figura 2.1.3 consta de seis entidades: Médico, Paciente, Expediente, Signos

Vitales y Estudios de Laboratorio, estas cuatro últimas entidades son subentidades de la entidad paciente.

La relación que tienen las entidades paciente y medico es de muchos a muchos (n-m), es decir, muchos médicos pueden tener muchos pacientes, así como muchos pacientes pueden tener muchos médicos.

Los atributos de la entidad médico son: id\_medico, especialidad, contraseña, y nombre completo, el cual es un atributo compuesto por nombre y apellidos.

Los atributos de la entidad paciente son: id\_paciente, id\_expediente, id\_medico.

Los identificadores de cada relación son: id\_medico en la entidad médico, id\_paciente en la entidad paciente.

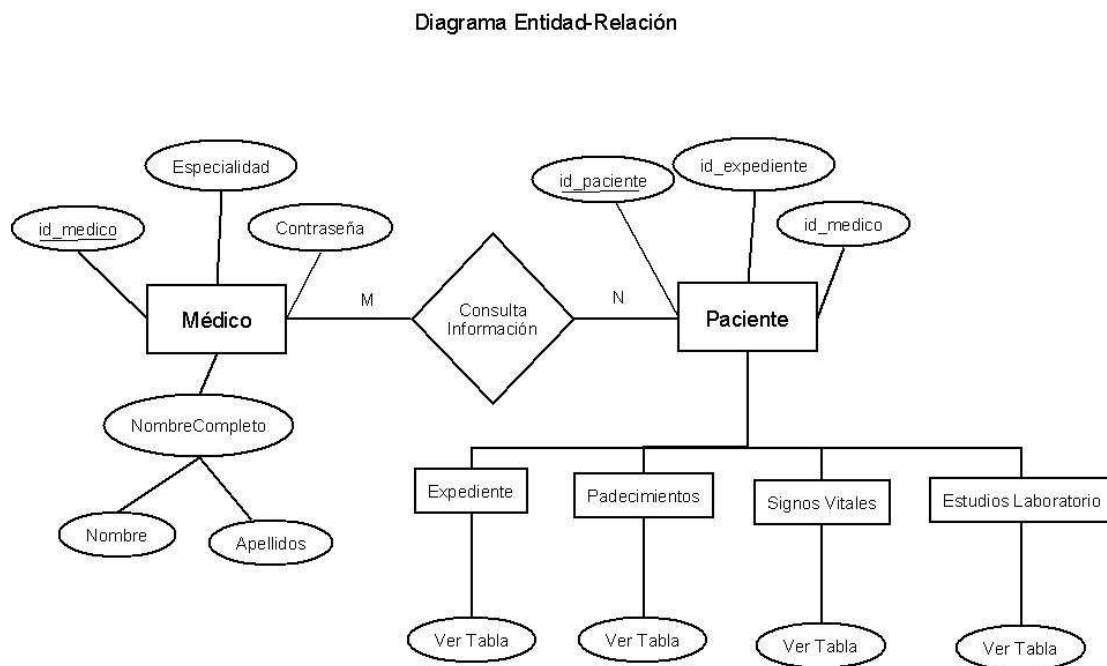


Figura 2.1.3 Diagrama Entidad Relación del Sistema Propuesto

Los siguientes atributos que se muestran en la siguiente tabla pertenecen a las entidades mencionadas en el diagrama.

<b>Expediente</b>	<b>Padecimientos</b>	<b>Signos Vitales</b>	<b>Estudios de Laboratorio</b>
id_paciente	id_paciente	id_paciente	id_paciente
<u>id_expediente</u>	<u>id_padecimiento</u>	<u>id_signos_vitales</u>	<u>id_estudios_laboratorio</u>
Nom_paciente	Alergias	fecha_actual	id_bioquimica_clinica
fecha_nacimiento	enfermedad_cronica	temperatura	id_uroanalisis
Edad	operaciones_realizadas	Pulso	id_hormonas
inicio_menstruacion	Tratamientos	ritmo_cardiaco	id_hematologia
inicio_actividad_sexual	-transtornos	Presion	id_coagulacion
parejas_sexuales	fecha_ultima_operacion	frecuencia respiratoria	id_microbiologia Parasitologia
Peso	fecha_ultimo_tratamiento		id_serologia
estatura			id_marcadores_tumorales
antecedentes_familiares			

Figura 2.1.4 Tabla con los atributos de las entidades del Modelo E-R

## 2.2 UML (Lenguaje Unificado de Modelado)

UML es un lenguaje visual para el modelado y la comunicación entre sistemas a través del uso de diagramas y texto de ayuda, como sabemos UML abarca tres aspectos, el lenguaje, el modelado y el unificar.

Cuando hablamos de lenguajes en UML nos referimos a que es un lenguaje para construir, visualizar y documentar los conceptos de un proceso intensivo a la hora de crear un sistema, esto se enfoca en incluir los pasos a seguir para producir, mantener o dar al sistema los requerimientos que necesite.

El modelado es la representación de un caso, el modelado captura las ideas acerca del tema o caso que se este diseñando, sin un modelo es muy difícil para los miembros de un equipo tener el conocimiento en común de los requerimientos del sistema, así mismo es difícil considerar

cambios cuando el sistema se desarrolla si no se tiene un modelo. Es muy importante enfocarse en obtener la información relevante requerida para entender el problema.

El término unificado se refiere al hecho que el “Object Management Group (OMG) “, es una industria reconocida la cual trabajando en conjunto con IBM crearon el estándar UML.

El propósito general del UML puede ser aplicado a través del desarrollo del sistema, esto ayuda a conocer todos los procesos y requerimientos posibles para implementar el sistema.

Para tener diferentes vistas del modelo, en UML se definen los siguientes diagramas gráficos:

- Class Diagram (Diagrama de clases)
- Use Case Diagram (Diagrama de casos de uso)
- Diagramas de comportamiento:
  - Statechart Diagram (Diagrama de estados)
  - Activity Diagram (Diagrama de actividades)
- Diagramas de interacción
  - Sequence Diagram (Diagrama de secuencia)

Estos diagramas proporcionan múltiples perspectivas del sistema bajo análisis. El modelo integra estos aspectos de forma que se puede construir un sistema autoconsistente. Estos diagramas, junto con la documentación de soporte, es lo primero que ve el diseñador o desarrollador de Software. [10]

### **2.2.1 Diagrama de Clases**

Un diagrama de clases sirve para visualizar las relaciones entre las clases que involucran el sistema, las cuales pueden ser asociativas, de herencia, de uso.

Un diagrama de clases esta compuesto por los siguientes elementos:

- Clase: atributos, métodos y visibilidad.
- Relaciones: Herencia, Composición, Agregación, Asociación y Uso.

Los diagramas de Clase son más útiles para ilustrar relaciones entre clases e interfaces, las generalizaciones, agregaciones, y asociaciones son importantes al reflejar herencias, composición o uso, y conexiones respectivamente.

#### **Clase**

Es la unidad básica que encapsula toda la información de un Objeto (un objeto es una instancia de una clase). A través de ella podemos modelar el entorno en estudio (una Casa, un Auto, una Cuenta Corriente, etc.).

En UML, una clase es representada por un rectángulo que posee tres divisiones ver figura 2.2.1:

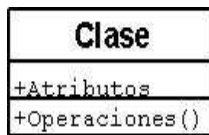


Figura 2.2.1 Representación gráfica en UML de una clase

En donde la parte:

- ❖ **Superior:** Contiene el nombre de la Clase
- ❖ **Intermedio:** Contiene los atributos (o variables de instancia) que caracterizan a la Clase (pueden ser privadas, protegidas o publicas).
- ❖ **Inferior:** Contiene los métodos u operaciones, los cuales son la forma como interactúa el objeto con su entorno (dependiendo de la visibilidad: privadas, protegidas o públicas).

### Relaciones

En UML, la cardinalidad de las relaciones indica el grado y nivel de dependencia, se anotan en cada extremo de la relación y éstas pueden ser:

- uno o muchos: 1..\* (1..n)
- 0 o muchos: 0..\* (0..n)
- número fijo: m (m denota el número).

### Herencia (Especialización/Generalización)

Una generalización se usa para indicar herencia, es decir, indica que una subclase hereda los métodos y atributos especificados por una Super Clase, por ende la Subclase además de poseer sus propios métodos y atributos, poseerá las características y atributos visibles de la Super Clase (publica y protegida).

### Agregación

Para modelar objetos complejos, no bastan los tipos de datos básicos que proveen los lenguajes: enteros, reales y secuencias de caracteres. Cuando se requiere componer objetos que son instancias de clases definidas por el desarrollador de la aplicación, tenemos dos posibilidades:

➤ Por Valor: Es un tipo de relación estática, en donde el tiempo de vida del objeto incluido esta condicionado por el tiempo de vida del que lo incluye. A esta relación se le llama llamada Composición (el Objeto base se construye a partir del objeto incluido, es decir, es "parte/todo").

➤ Por Referencia: Es un tipo de relación dinámica, en donde el tiempo de vida del objeto incluido es independiente del que lo incluye. A esta relación se le llama Agregación (el objeto base utiliza al incluido para su funcionamiento).

## Asociación

Una asociación implica que dos elementos del modelo tienen una relación – usualmente implementada como una variable de instancia de una clase. Este conector puede incluir roles nombrados en cada extremo, cardinalidad, dirección y restricciones. Una asociación es el tipo de relación general entre elementos. Para más de dos elementos, un elemento de la caja de herramientas de representación diagonal también se puede usar. Cuando se genera código para diagramas de clase, las asociaciones se convierten en variables de instancia en la clase de destino. [10]

### 2.2.1.1 Notación del Diagrama De Clases

El diagrama de Clase muestra los bloques de construcción de cualquier sistema orientado a objetos. Los diagramas de clases describen la vista estática del modelo o parte del modelo, describiendo que atributos y comportamientos tienen en lugar de detallar los métodos para realizar operaciones, en la siguiente figura 2.2.2 se vera como se representa gráficamente cada uno de los elementos mencionados con anterioridad.

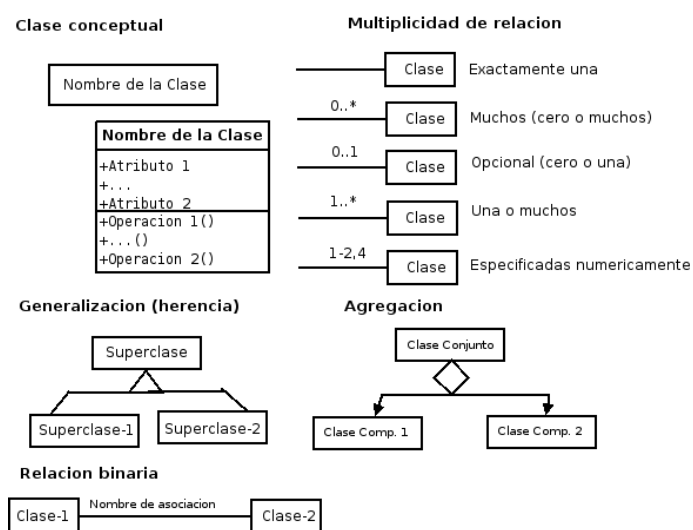


Figura 2.2.2 Notación Del Diagrama de Clases

### 2.2.1.2 Diagrama De Clases Para El Sistema De Base De Datos De Archivos Clínicos

El diagrama de clases del sistema propuesto se puede ver en la figura 2.2.3, este consta de 2 clases principales y 11 subclases.

La clase médico tiene como atributos:

- id\_medico: es el número de trabajador, el cual será nuestra llave primaria de la clase medico, es tipo varchar (30).
- nom\_medico: es el nombre del médico trabajador de tipo varchar (30).
- especialidad: en que área esta especializado el medico, por ejemplo: ginecología, medicina interna, oncología, etc. Es de tipo varchar (30).
- contraseña: el código que utilizará para poder ingresar al sistema también de tipo varchar (30).

La clase médico tiene como operaciones:

- crearMedico: esta operación se encargará de registrar a un nuevo medico ingresando su número de trabajador y su nombre.
- acceder: esta operación recibirá dos parámetros, el nombre del medico y su contraseña, esto servirá para poder acceder al sistema.
- autenticar: es muy importante la seguridad en un sistema, es por eso que se creará esta operación, verificará que los datos del médico sean correctos.
- ObtenerExpediente: servirá para poder buscar el expediente del paciente con el identificador.

La clase paciente tiene como atributos:

- id\_paciente: es el identificador, la llave primaria de la clase paciente, la cual es tipo varchar (30).
- id\_expediente: es la clave de nuestra subclase expediente, es tipo varchar (30).
- id\_signos\_vitales: es la clave de nuestra subclase signos vitales es tipo varchar (30).
- id\_padecimientos: es la clave de nuestra subclase padecimientos es tipo varchar (30).
- id\_estudios\_laboratorio: es la clave de nuestra subclase resultados de estudios de laboratorio es tipo varchar (30).

La clase paciente tiene como operaciones:

- crearPaciente: esta operación se encargará de registrar a un nuevo paciente, en este caso solo se creará el id\_paciente, ya que esta clase tiene cuatro subclases que contendrán la información del paciente.
- busqueda: esta operación podrá buscar el expediente general del paciente ya sea por sus padecimientos, los resultados de los análisis de laboratorio, o sus signos vitales.
- mostrar: esta operación podrá mostrar el expediente general del paciente ya sea por sus padecimientos, los resultados de los análisis de laboratorio, o sus signos vitales

En el caso de las subclases Padecimientos, Expedientes, Estudios de Laboratorio y Signos Vitales cada una tendrá por herencia el atributo id\_paciente, para que de esta manera el usuario pueda buscarlos por nombre, fecha, estudios, etc.

En general en las subclases, las operaciones que todas estas tienen son:

- ingresar: esta operación, nos permitirá añadir nuevos registros o información a los registros ya guardados en la base de datos.
- modificar: esta operación, nos permitirá modificar registros ya guardados en la base de datos.
- buscar: esta operación, nos permitirá buscar registros específicos ya guardados en la base de datos.
- Consultar: esta operación, nos permitirá consultar todos los registros ya guardados en la base de datos.

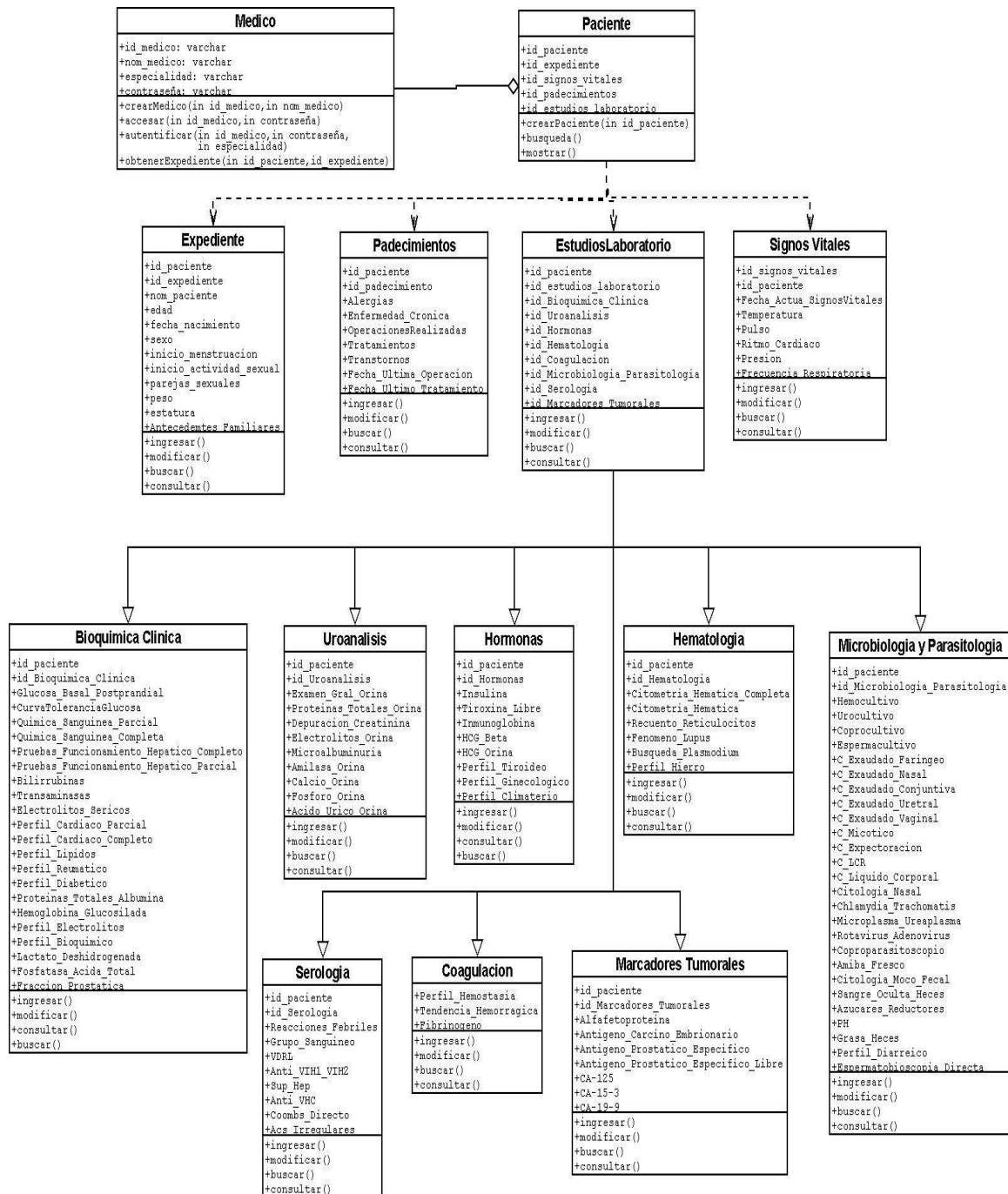


Figura 2.2.3 Diagrama De Clases Para El Sistema Propuesto

## 2.2.2 Diagrama de Casos de Uso

Un caso de uso es una descripción de las acciones de un sistema desde el punto de vista del usuario, se dice que es una técnica de aciertos y errores para obtener los requerimientos del

sistema desde el punto de vista del usuario, esto es importante si la finalidad es crear un sistema que pueda ser utilizado por la gente en general.

En el diagrama de casos de uso se representa también el sistema como una caja rectangular con el nombre en su interior. Los casos de uso están en el interior de la caja del sistema, y los actores fuera, y cada actor está unido a los casos de uso en los que participa mediante una línea.

### **Elementos**

Los elementos que pueden aparecer en un Diagrama de Casos de Uso son: actores, casos de uso y relaciones entre casos de uso.

### **Actores**

Un actor es algo con comportamiento, como una persona (identificada por un rol), un sistema informatizado u organización, y que realiza algún tipo de interacción con el sistema. Se representa mediante una figura humana dibujada con palotes. Esta representación sirve tanto para actores que son personas como para otro tipo de actores.

### **Casos de Uso**

Un caso de uso es una descripción de la secuencia de interacciones que se producen entre un actor y el sistema, cuando el actor usa el sistema para llevar a cabo una tarea específica. Expresa una unidad coherente de funcionalidad, y se representa en el Diagrama de Casos de Uso mediante una elipse con el nombre del caso de uso en su interior. El nombre del caso de uso debe reflejar la tarea específica que el actor desea llevar a cabo usando el sistema.

### **Relaciones entre Casos de Uso**

Un caso de uso, en principio, debería describir una tarea que tiene un sentido completo para el usuario. Sin embargo, hay ocasiones en las que es útil describir una interacción con un alcance menor como caso de uso. La razón para utilizar estos casos de uso no completos en algunos casos, es mejorar la comunicación en el equipo de desarrollo, el manejo de la documentación de casos de uso. [10]

#### **2.2.2.1 Notación del Diagrama De Casos de Uso**

Hay que recordar que los diagramas de casos de uso sirven para facilitar la comunicación con los futuros usuarios del sistema, y con el cliente, y resultan especialmente útiles para determinar las características necesarias que tendrá el sistema, ver figura 2.2.4.

Las descripciones de casos de uso son reseñas textuales del caso de uso. Normalmente tienen el formato de una nota o un documento relacionado de alguna manera con el caso de uso, y explica los procesos o actividades que tienen lugar en el caso de uso.

Notación del Diagrama de Casos de Uso

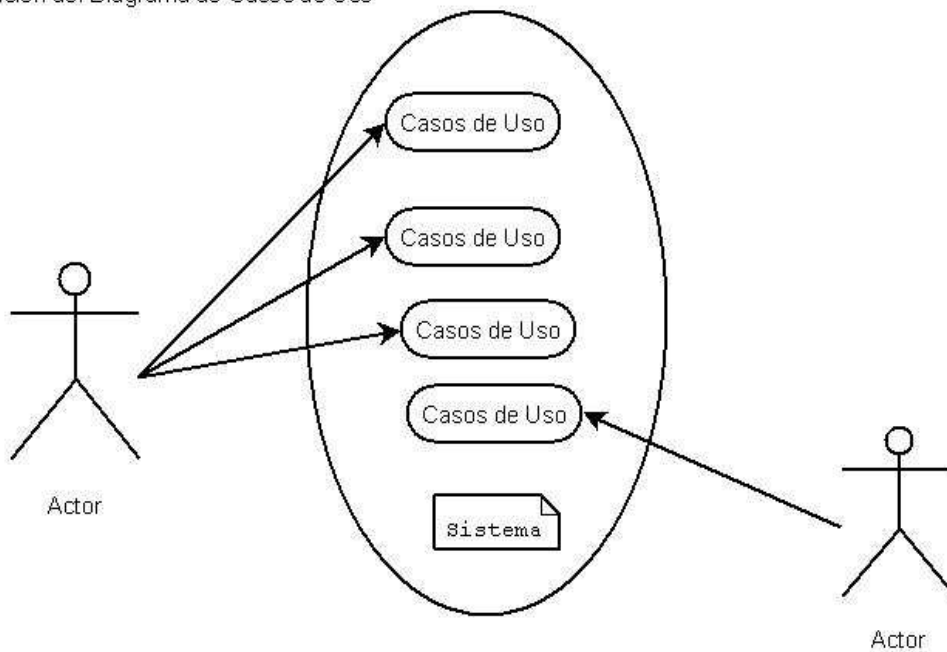


Figura 2.2.4 Notación del Diagrama de Casos de Uso

### 2.2.2.2 Diagrama De Casos de Uso Para El Sistema De Base De Datos De Archivos Clínicos

En el diagrama de casos de usos del sistema de base de datos de archivos clínicos que observamos en la figura 2.2.5, muestra claramente como funciona el sistema, hay dos actores que pueden hacer uso del sistema, el médico y el administrador del sistema, la función del administrador del sistema será registrar los médicos trabajadores del hospital y del funcionamiento correcto de las etiquetas RFID, a su vez se encargará de cualquier fallo que se presente en la comunicación de las etiquetas RFID con la base de datos.

El medico podrá ingresar al sistema una vez ya registrado, él se encargará del registro de los pacientes, de los cuales podrá ver su expediente, podrá ingresar nueva información, modificar información ya guardada y podrá salir del sistema.

Cabe mencionar que los archivos clínicos solo se consultarán por los médicos designados, nadie más tiene acceso al expediente.

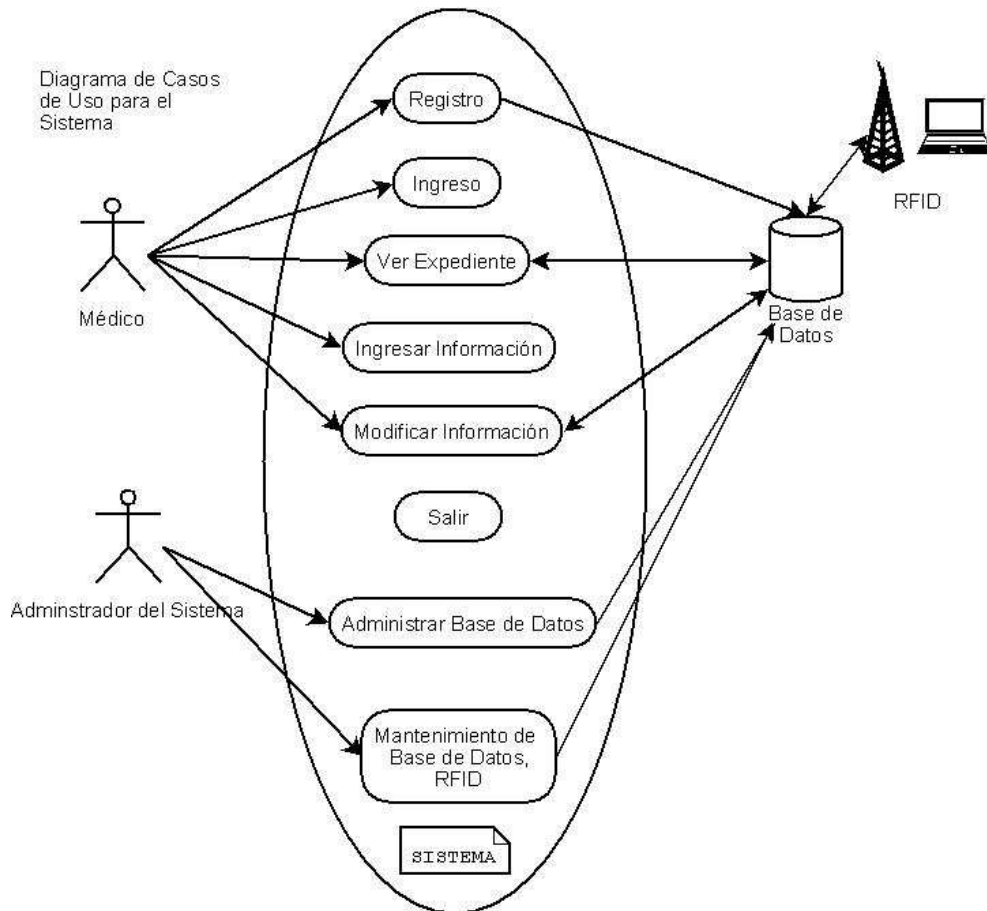


Figura 2.2.5 Diagrama de Casos de Uso para el Sistema Propuesto

### 2.2.3 Diagrama de Estados

Los diagramas de estado muestran los diferentes estados de un objeto durante su vida, y los estímulos que provocan los cambios de estado en un objeto.

Los diagramas de estado ven a los objetos como máquinas de estado o autómatas finitos que pueden estar en un conjunto de estados finitos y que pueden cambiar su estado a través de un estímulo perteneciente a un conjunto finito.

#### Estado

Los estados son los objetos de los diagramas de estado. Un estado pertenece a exactamente una clase y representa un resumen de los valores y atributos que puede tener la clase. Un estado UML

describe el estado interno de un objeto de una clase particular. Los cambios en los atributos de un objeto deben estar representados por estados, sino únicamente aquellos cambios que pueden afectar significativamente a la forma de funcionamiento del objeto

Hay dos tipos especiales de estados: inicio y fin. Son especiales en el sentido de que no hay ningún evento que pueda devolver a un objeto a su estado de inicio, y de la misma forma no hay ningún evento que pueda sacar a un objeto de su estado de fin.

Una transición entre estados representa un cambio de un estado origen a un estado sucesor destino que podría ser el mismo que el estado origen, dicho cambio de estado puede ir acompañado de alguna acción. Las acciones se asocian a las transiciones y se considera que ocurren de forma rápida y no interrumpible. Por otro lado, las actividades se asocian a los estados pudiendo consumir más tiempo, dicha actividad puede verse interrumpida por la ocurrencia de algún evento. [10]

### **2.2.3.1 Notación del Diagrama De Estados**

Los diagramas de estados abarcan todos los mensajes que un objeto puede enviar o recibir. En un diagrama de estados, un escenario representa un camino dentro del diagrama. Dado que generalmente el intervalo entre dos envíos de mensajes representa un estado, se pueden utilizar los diagramas de secuencia para buscar los diferentes estados de un objeto.

En notación UML, un estado se representa mediante un rectángulo con los bordes redondeados, que puede tener tres compartimentos: uno para el nombre, otro para el valor característico de los atributos del objeto en ese estado y otro para las acciones que se realizan al entrar, salir o estar en un estado. Los estados iniciales y finales se representan mediante los símbolos de la figura 2.2.6.

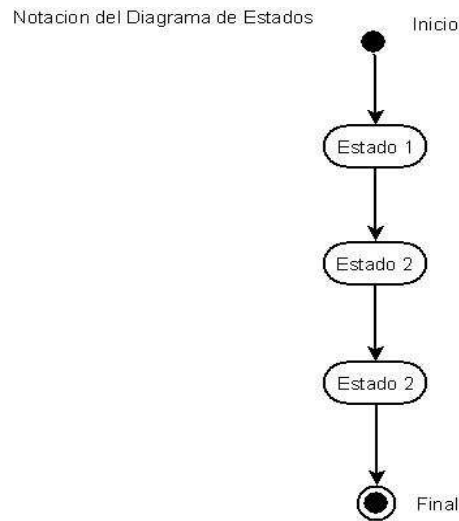


Figura 2.2.6 Notación del Diagrama de Estados

### 2.2.3.2 Diagrama De Estados Para El Sistema De Base De Datos De Archivos Clínicos

En la figura 2.2.7 podemos ver el diagrama de estados del sistema propuesto, en general tenemos que el usuario manda a solicitar la información, en ese momento la etiqueta RFID va a estar transmitiendo la información la cual será recibida y almacenada en la base de datos, después de eso el usuario podrá consultar la información obtenida.



Figura 2.2.7 Diagrama de estados del sistema propuesto

## 2.2.4 Diagrama de Secuencia

Los diagramas de secuencia muestran el intercambio de mensajes (es decir la forma en que se invocan) en un momento dado. Los diagramas de secuencia ponen especial énfasis en el orden y el momento en que se envían los mensajes a los objetos.

En los diagramas de secuencia, los objetos están representados por líneas intermitentes verticales, con el nombre del objeto en la parte más alta. La línea de tiempo también es vertical, incrementándose hacia abajo, de forma que los mensajes son enviados de un objeto a otro en forma de flechas con los nombres de la operación y los parámetros. [11]

### 2.2.4.1 Notación del Diagrama De Secuencia

Los diagramas de clase de UML no incluyen flujo de mensajes entre clases, es por esto que los diagramas de secuencia se deben crear en paralelo con los diagramas de clases.

Los componentes principales de un diagrama de secuencia son ver figura 2.2.8:

- Objeto: Se representa con un rectángulo que contiene el nombre y la clase del objeto en un formato *nombreObjeto: nombreClase*.
- Enlaces: Un enlace es una instancia de una asociación en un diagrama de clases. Se representa como una línea continua que une a dos objetos, acompañada por un número que indica el orden dentro de la interacción.
- Flujo de mensajes: Expresa el envío de un mensaje. Se representa mediante una flecha dirigida cerca de un enlace.
- Marcadores de creación y destrucción de objetos: Puede mostrarse en la gráfica qué objetos son creados y destruidos, agregando una restricción con la palabra *new* o *delete* respectivamente.

Objeto compuesto: Es una representación alternativa de un objeto y sus atributos. En esta representación se muestran los objetos contenidos dentro del rectángulo que representa al objeto que los contiene. [11]

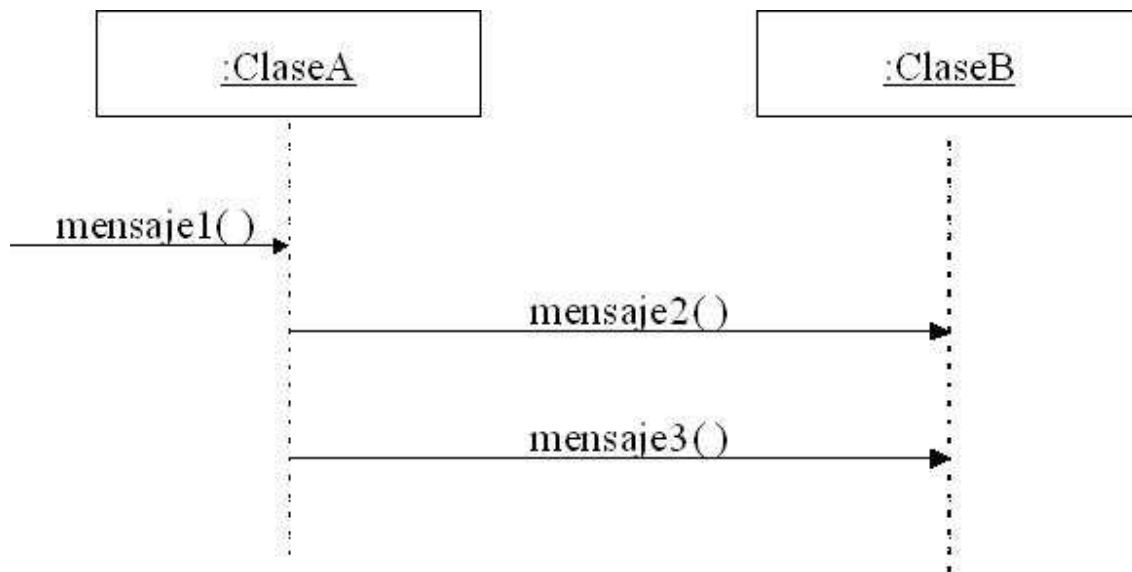


Figura 2.2.8 Notación de un diagrama de Secuencia

#### 2.2.4.2 Diagrama De Secuencia Para El Sistema De Base De Datos De Archivos Clínicos

El diagrama de secuencia de este sistema (ver figura 2.2.9) consta de dos clases principales medico y paciente, cuatro subclases que pertenecen a la clase paciente, en este caso las llamamos expediente general por ser información del paciente de diversas índoles (expediente, resultados de laboratorio, padecimientos, signos vitales), el usuario, en este caso el medico, ingresará al sistema, verificará que realmente es el medico e ingresara el identificador del paciente, de esta manera consulta la información que esta almacenada en la base de datos, la cual recibe lo datos desde las etiquetas RFID e inmediatamente después almacena la información, el usuario después de visualizar la información saldrá del sistema sin problema alguno.

El usuario administrador solo logrará registrar médicos en la base de datos y salir del sistema.

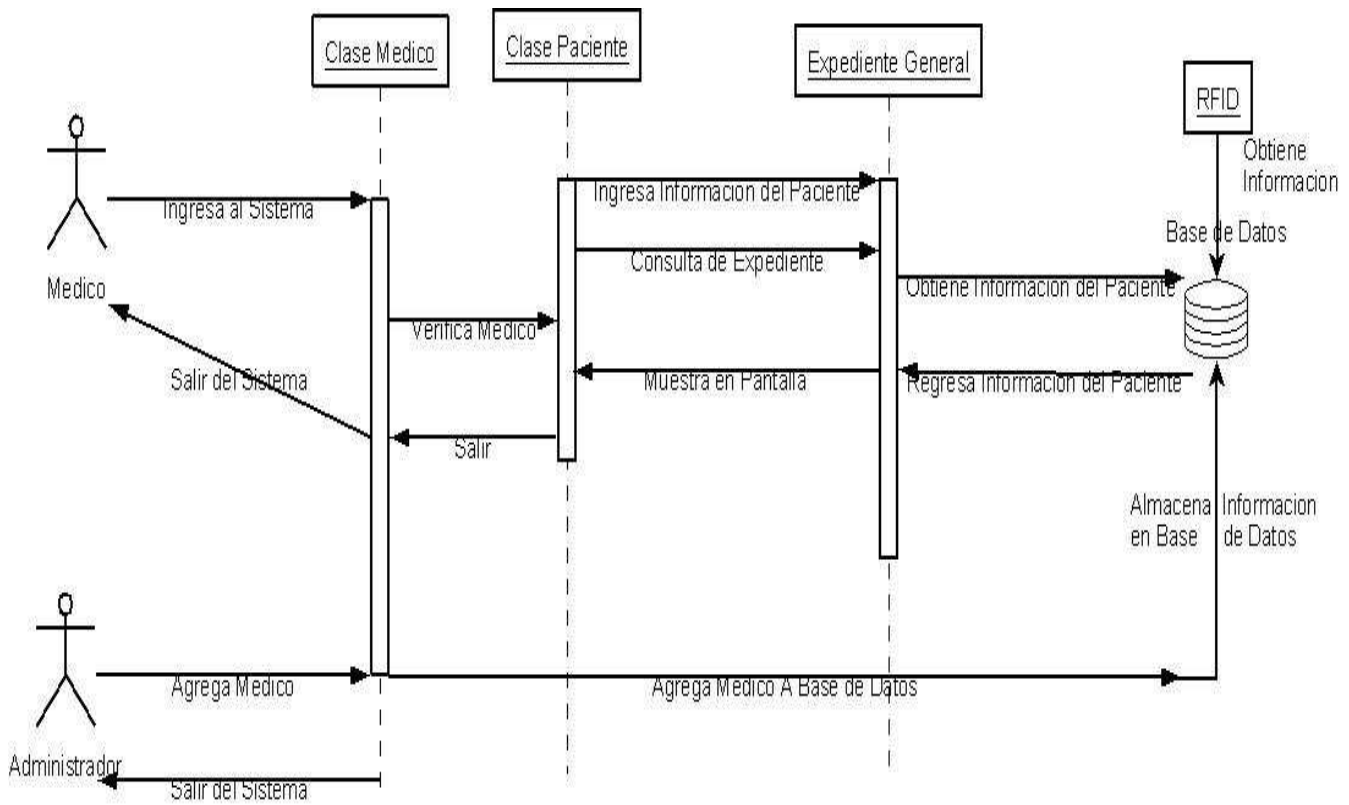


Figura 2.2.9 Diagrama de Secuencia para el Sistema Propuesto

## **CAPÍTULO III**

### **Implementación**

#### **Introducción**

Los lenguajes de programación son lenguajes creados por el ser humano para poder comunicarse con las computadoras. Se puede decir que un lenguaje de programación es el conjunto de símbolos y palabras que permiten al usuario de una computadora darle instrucciones y órdenes para que la computadora los ejecute.

En este capítulo se hablará del software que sirve para conectar la computadora con la etiqueta, poderla grabar, leer y borrar, dicho software está en el lenguaje de Programación llamado C# del cual se hablará más adelante.

#### **3.1 Lenguajes de Programación**

Los lenguajes de programación son herramientas que nos permiten crear programas y software, es decir, ellos facilitan la tarea de programación, los lenguajes de programación representan en forma simbólica y en manera de un texto los códigos que podrán ser leídos por una persona.

Los procesadores del lenguaje son los programas que permiten el tratamiento de la información en forma de texto, representada en los lenguajes de programación simbólicos, existen lenguajes de programación que utilizan compilador.

La ejecución de un programa con compilador requiere de dos etapas:

- 1) Traducir el programa simbólico a código máquina
- 2) Ejecución y procesamiento de los datos.

Otros lenguajes de programación utilizan un programa intérprete o traductor, el cual analiza directamente la descripción simbólica del programa fuente y realiza las instrucciones dadas. El intérprete en los lenguajes de programación simula una máquina virtual, donde el lenguaje de máquina es similar al lenguaje fuente. La ventaja del proceso intérprete es que no necesita de dos fases para ejecutar el programa, sin embargo su inconveniente es que la velocidad de ejecución es más lenta ya que debe analizar e interpretar las instrucciones contenidas en el programa fuente.

En general hay variedad en cuanto se habla de lenguajes de programación:

**El Lenguaje Máquina:** es el lenguaje de programación que entiende directamente la computadora o máquina. Este lenguaje de programación utiliza el alfabeto binario, es decir, el 0 y el 1. Con estos dos únicos dígitos, conocidos como bits, forma lo que se conoce como cadenas binarias

(combinaciones de ceros y unos) son con las que se escriben las instrucciones que el microprocesador de la computadora entiende nuestra peticiones. Fue el primer lenguaje de programación. Este tipo de lenguaje de programación dejó de utilizarse por su gran dificultad y por la facilidad para cometer errores.

Lenguajes de programación de bajo nivel: Son mucho más fáciles de utilizar que el lenguaje máquina, pero dependen mucho de la máquina o computadora como sucedía con el lenguaje máquina. El lenguaje ensamblador fue el primer lenguaje de programación que trató de sustituir el lenguaje máquina por otro mucho más parecido al de los seres humanos.

Lenguajes de programación de alto nivel: Este tipo de lenguajes de programación son independientes de la máquina, lo podemos usar en cualquier computador con muy pocas modificaciones o sin ellas, son muy similares al lenguaje humano, pero precisan de un programa interprete o compilador que traduzca este lenguaje de programación de alto nivel a uno de bajo nivel como el lenguaje de máquina que la computadora pueda entender.

Los lenguajes de programación de alto nivel son más fáciles de aprender porque se usan palabras o comandos del lenguaje natural. Existen muchos lenguajes de programación de alto nivel por eso es difícil clasificarlos, no obstante se pueden clasificar desde el punto de vista del uso de los programas y la filosofía de su creación es la siguiente:

- Lenguajes de programación imperativos: el Cobol, Pascal, C y Ada, etc.
- Lenguajes de programación declarativos: el Lisp y el Prolog.
- Lenguajes de programación orientados a objetos: el Smalltalk y el C++, C#, Java
- Lenguajes de programación orientados al problema: son aquellos lenguajes específicos para gestión.

Lenguajes de programación naturales: son los nuevos lenguajes que pretenden aproximar el diseño y la construcción de programas al lenguaje de las personas. [12]

### **3.2 Programación Orientada a Objetos**

La Programación Orientada a Objetos (también conocida como P.O.O. o O.O.P., por sus siglas en inglés) es un paradigma o modelo de programación. En otras palabras no es un lenguaje específico, o una tecnología, sino una manera de programar, una forma de plantearse la

programación. No es el único modelo existente, ya que tenemos la programación estructurada y algunas otras más, pero se ha constituido en una de las formas de programar más populares e incluso muchos de los lenguajes que usamos hoy día lo soportan o están diseñados bajo ese modelo (PHP, JAVA, entre otros).

Lo que caracteriza a la Programación Orientada a Objetos es que intenta llevar al mundo del código lo mismo que encontramos en El Mundo Real. El modelo que la POO intenta seguir para estructurar un sistema es que cada objeto pertenece a una clase para un ejemplo más claro ver figura 3.2.1.

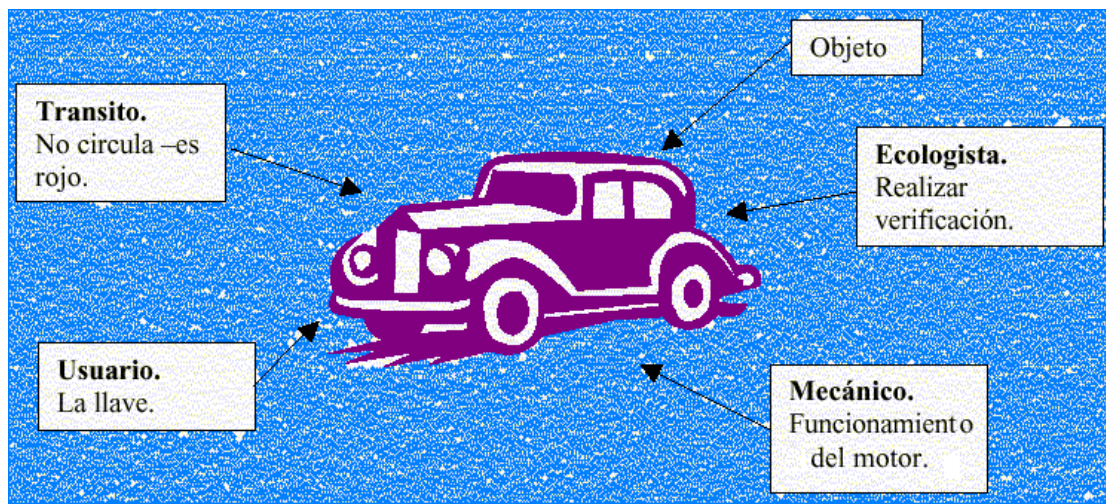


Figura 3.2.1 Ejemplo del funcionamiento POO con un objeto del Mundo Real

Un objeto es un elemento independiente de un programa de computadora, que representa un grupo asociado de características y está diseñado para realizar tareas específicas. A los objetos también se les conoce como instancias.

Cada objeto tiene un papel específico en un programa, y todos los objetos pueden funcionar con otros objetos en maneras definidas.

Una clase es una plantilla que se utiliza para crear múltiples objetos con características similares. Las clases engloban todas las características de un conjunto particular de objetos. Cuando se escribe un programa en un lenguaje orientado a objetos, no se definen objetos individuales, sino que se definen clases de objetos.

Las características de las instancias en la Programación Orientada a Objetos son:

1. Modularización: descomponer la aplicación en partes más pequeñas.
2. Abstracción.
3. Encapsulación: ocultación de información.
4. Jerarquización: relaciones de herencia o generalización (es\_un) y relaciones de contenido o agregación (tiene\_un).
5. Polimorfismo: capacidad de referirse a objetos de clases distintas en una jerarquía utilizando el mismo elemento de programa para realizar la misma operación, pero de formas distintas. [13]

### 3.3 CSharp (C#)

Microsoft.NET es el conjunto de nuevas tecnologías en las que Microsoft ha estado trabajando durante los últimos años con el objetivo de obtener una plataforma sencilla y potente para distribuir el software en forma de servicios que puedan ser suministrados remotamente y que puedan comunicarse y combinarse unos con otros de manera totalmente independiente de la plataforma, lenguaje de programación y modelo de componentes con los que hayan sido desarrollados. Ésta es la llamada plataforma .NET.

Su sintaxis básica deriva de C/C++ y utiliza el modelo de objetos de la plataforma.NET el cual es similar al de Java aunque incluye mejoras derivadas de otros lenguajes (entre ellos Delphi).

En el caso del lenguaje C#, la intención original de los creadores del lenguaje fue llamarlo "C en re bemol", para denotar belleza e inteligencia en el diseño de tal lenguaje. Debido a que se encontró parecido el símbolo "sharp" o "numeral" en español, al correspondiente de Do sostenido mayor, el nombre del lenguaje quedó como C#.

Aunque C# forma parte de la plataforma.NET, ésta es una interfaz de programación de aplicaciones (**API**); mientras que C# es un lenguaje de programación independiente diseñado para generar programas sobre dicha plataforma.

El Common Language Runtime (CLR) es el núcleo de la plataforma .NET, el cual es el motor encargado de gestionar la ejecución de las aplicaciones para ella desarrolladas y a las que ofrece numerosos servicios que simplifican su desarrollo y favorecen su fiabilidad y seguridad.

La Librería de Clase Base (BCL) es una librería incluida en el .NET Framework formada por cientos de tipos de datos que permiten acceder a los servicios ofrecidos por el CLR y a las funcionalidades más frecuentemente usadas a la hora de escribir programas.

En resumen, C# es un lenguaje de programación que toma las mejores características de lenguajes preexistentes como Visual Basic, Java o C++ y las combina en uno solo. El hecho de ser relativamente reciente no implica que sea inmaduro, pues Microsoft ha escrito la mayor parte de la BCL usándolo, por lo que su compilador es el más depurado y optimizado de los incluidos en el .NET Framework SDK

Básicamente una aplicación en C# puede verse como un conjunto de uno o más archivos de código fuente con las instrucciones necesarias para que la aplicación funcione como se desea y que son pasados al compilador para que genere un ejecutable.

Cada uno de estos ficheros no es más que un fichero de texto plano escrito usando caracteres Unicode y siguiendo la sintaxis propia de C#. [15]

### **3.4 Implementación**

La implementación del Sistema de Base de Datos de Archivos Clínicos utilizando etiquetas RFID se programó en el lenguaje C#, utilizando como IDE SharpDevelopment 3.0 el cual es un IDE de código libre.

Un IDE es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica GUI. Los IDEs pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes. El lenguaje Visual Basic, por ejemplo, puede ser usado dentro de las aplicaciones de Microsoft Office, lo que hace posible escribir sentencias Visual Basic en forma de macros para Microsoft Word[16].

El sistema consta de un menú con dos opciones, en una de ellas el usuario podrá abrir el archivo Excel que contendrá la información almacenada, debido al uso que se le da a Excel se optó por utilizarlo para evitar que el usuario (Médico) tenga que estar copiando toda la base de datos de salir del sistema ver figura 3.4.1, en la opción de conexión tendremos un menú desplegable que mostrara la configuración del puerto serial, y 3 opciones de conexión/desconexión ver figura 3.4.2, la conexión de la antena es por el puerto serial.

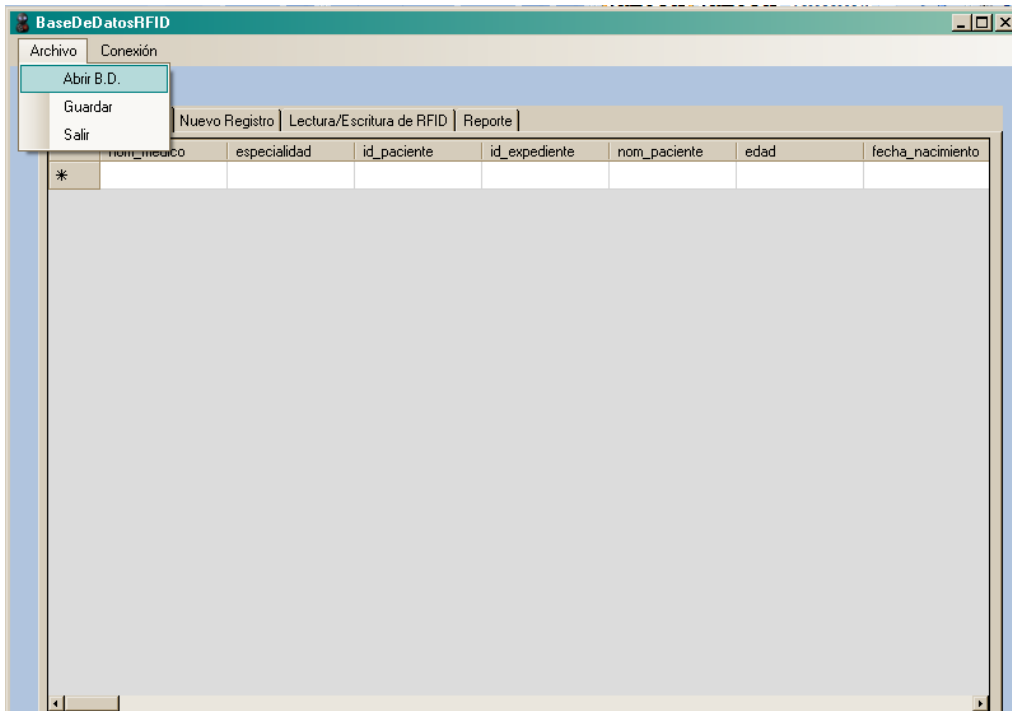


Figura 3.4.1 Cargando la Base de Datos

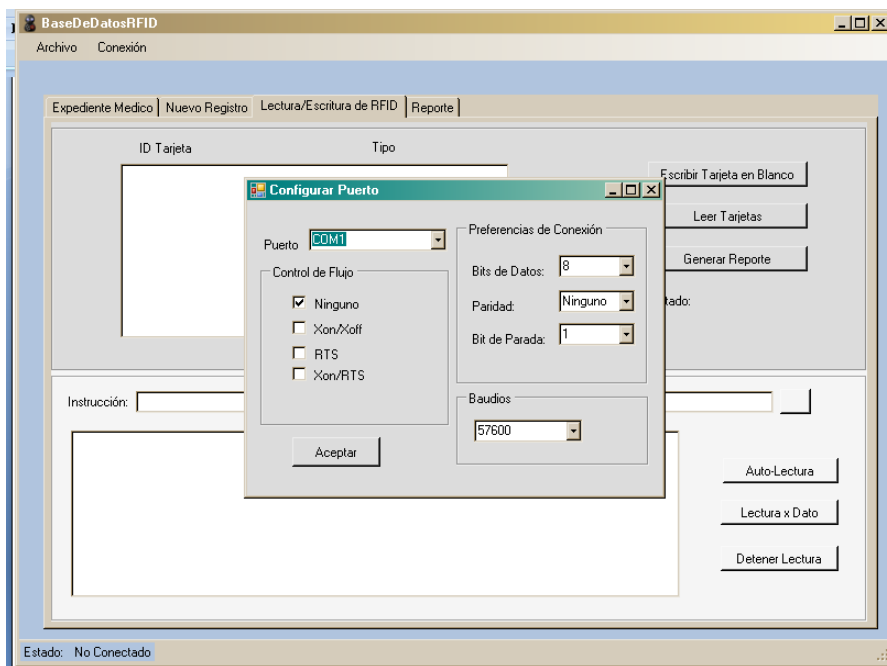


Figura 3.4.2 Configuración del Puerto Serial

El programa tiene la opción de llenar cada uno de los campos necesarios para poder formar el expediente del paciente, este se divide 13 pestañas, las cuales dan la opción de llenar el tipo de análisis que sea mandado por el médico ver figura 3.4.3.

The screenshot shows a software window titled 'BaseDeDatosRFID'. At the top, there is a menu bar with 'Archivo' and 'Conexión'. Below the menu bar is a tabbed interface with four tabs: 'Expediente Medico', 'Nuevo Registro', 'Lectura/Escritura de RFID', and 'Reporte'. The 'Expediente Medico' tab is active and contains 13 sub-tabs: 'Expediente', 'Padecimientos', 'Signos Vitales', 'E.L.: Bioquímica Clínica', 'E.L.: Uroanálisis', 'E.L.: Hematología', 'E.L.: Hormonas', 'E.L.: Serología', 'E.L.: Coagulación', 'E.L.: Marcadores Tumorales', 'Microbiología / Parasitología', and 'E.L.: Otros Análisis'. The main area of the window is a form for entering patient information. On the left side, there are several text labels followed by input fields: 'Número de Paciente:', 'Edad:', 'Sexo:', 'Fecha de Nacimiento:', 'Si es Mujer, Inicio Menstruación:', 'Inicio Actividad Sexual:', 'Parejas Sexuales:', 'Peso:', and 'Estatura:'. On the right side, there is a label 'Antecedentes Familiares:' followed by a large, empty rectangular text area. At the bottom center of the form area, there is a button labeled 'Guardar Registro'. At the bottom left of the window, there is a status bar that reads 'Estado: No Conectado'.

Figura 3.4.3 Llenado de información del paciente

Para poder escribir en las etiquetas RFID es necesario tener el lector conectado al puerto serial, cuando el lector esté conectado se podrá observar en la barra de estado del programa, esta nos dirá si se tiene o no conexión con el puerto serial .

Dentro del sistema se le añadió una pestaña la cual maneja la lectura de las tarjetas, ya que se debe tener certeza que las etiquetas están al alcance de la antena o se encuentren en su rango, con esta pestaña obtendremos la información, la cual si la queremos guardar, nos generará un reporte el cual será mostrado en la siguiente pestaña, el reporte puede ser o no guardado ver figura 3.4.4.

Se pueden tener diversas formas de lectura de las etiquetas RFID como lo es la auto-lectura, esto quiere decir que automáticamente cuando detecte la etiqueta cerca del rango la mostrara en la pantalla dándonos su identificación de etiqueta y el tipo que se está usando.

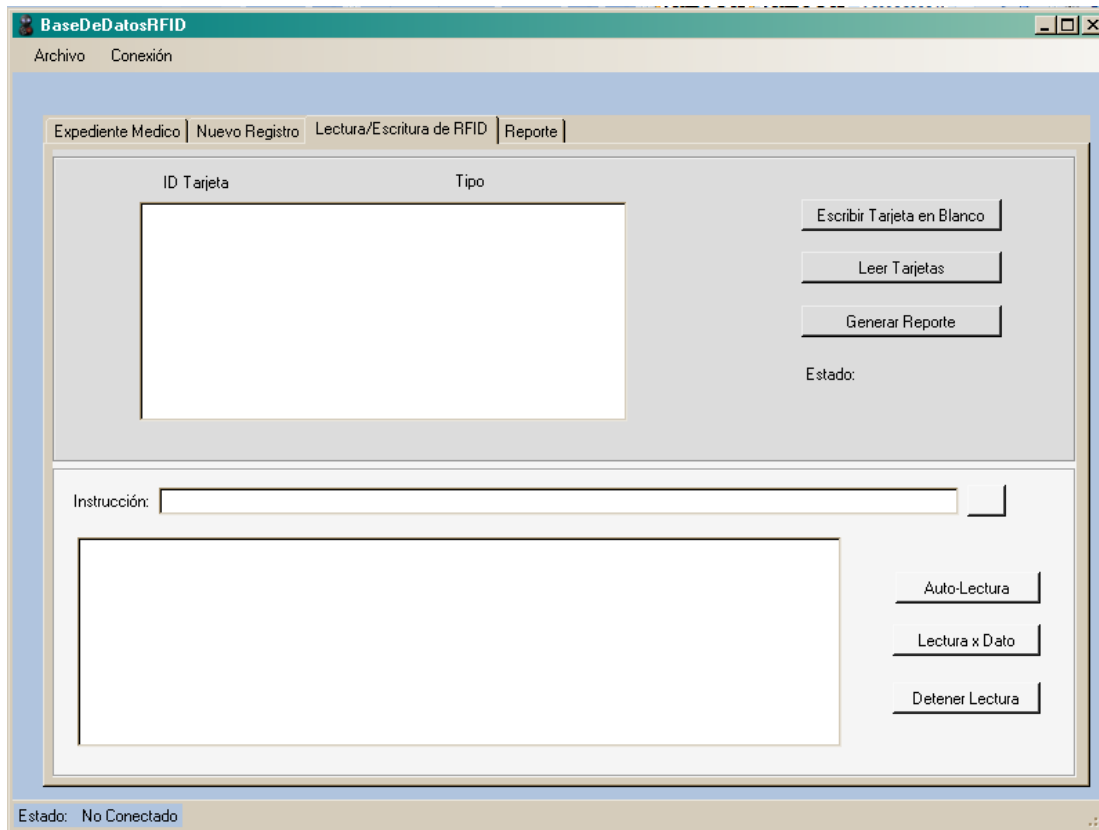


Figura 3.4.4 Interfaz para lectura de etiquetas RFID

La conectividad con la antena se hace por medio del sistema RFID (numero de serial 005558) como se ve en la figura 3.4.5 utilizando una antena de SAMSys , esta se conecta al puerto serial por medio de un adaptador para puerto serial (PL-2303 USB-SERIAL Driver), el cual se muestra en la figura 3.4.6, este nos permite utilizar equipos mas modernos que no cuenten con el puerto serial, la figura 3.4.7 se ve el tipo de etiquetas que se utilizan que son ECP-Gen02, Las etiquetas EPC son mejor conocidas como “Electronic Product code”, son etiquetas pasivas que nos permiten escribir 4 veces por segundo y leer 300 etiquetas por segundo, las EPC-Gen02 son aun más rápidas pueden escribir 7 veces por segundo y leer hasta 1000 , contiene seguridad, se le administra una contraseña y esta se puede utilizar para controlar la lectura o escritura en la tarjeta. [17][18]



Figura 3.4.5 Sistema RFID, “Low-Power UHF Reader Module”



Figura 3.4.6 Adaptador PL-2303 (USB-SERIAL Driver)



Figura 3.4.7 Etiqueta ECP- Gen02

Por ultimo hablemos de la privacidad de la información, para esto, el médico deberá estar registrado en la base de datos para poder acceder al sistema como podemos observar en la figura 3.4.8, esto es muy importante ya que la privacidad es un punto clave en los expediente de cualquier tipo, si el médico usuario no esta registrado no podrá acceder directamente al sistema, una vez ya hecho esto el usuario accederá al programa, y podrá hacer el tipo de operaciones que desee.

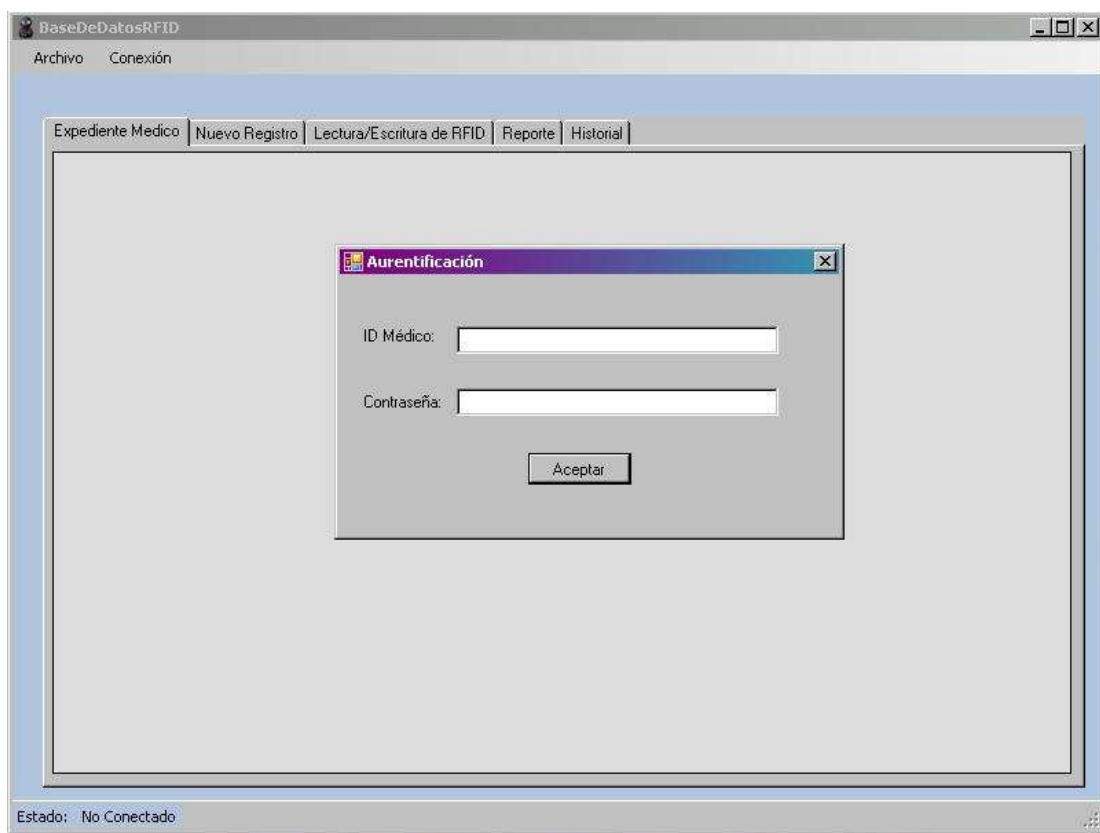


Figura 3.4.8 Ingreso al Sistema Base de Datos de Archivos Clínicos

## CAPÍTULO IV

### Pruebas del Sistema

#### Introducción

Las pruebas se realizan a lo largo del desarrollo del sistema y no simplemente al final. Esto significa sacar a la luz problemas no conocidos y no demostrar la perfección de programas manuales o equipo, también debe ser probado el sistema trabajando con un todo. Esto incluye probar las interfaces entre subsistemas, la corrección de la salida y la utilidad y comprensibilidad de la documentación de la salida del sistema. En este capítulo veremos los resultados obtenidos después de implementar el sistema.[18][19]

#### 4.1 Pruebas del Sistema

Para comenzar tenemos el acceso al sistema, en las pruebas se realizaron 3 diferentes formas de acceder al sistema, en la figura 4.1.1 se muestra como se accede con éxito al sistema, es decir, que el médico esté registrado en la base de datos, la figura 4.1.2 muestra el acceso al sistema cuando se le da un id medico o contraseña incorrecta, este mandará mensaje de error el cual avisa que no se encuentra en la base de datos y por ultimo será el caso cuando el usuario no escribe nada y nos manda un mensaje de error , ver figura 4.1.3 .

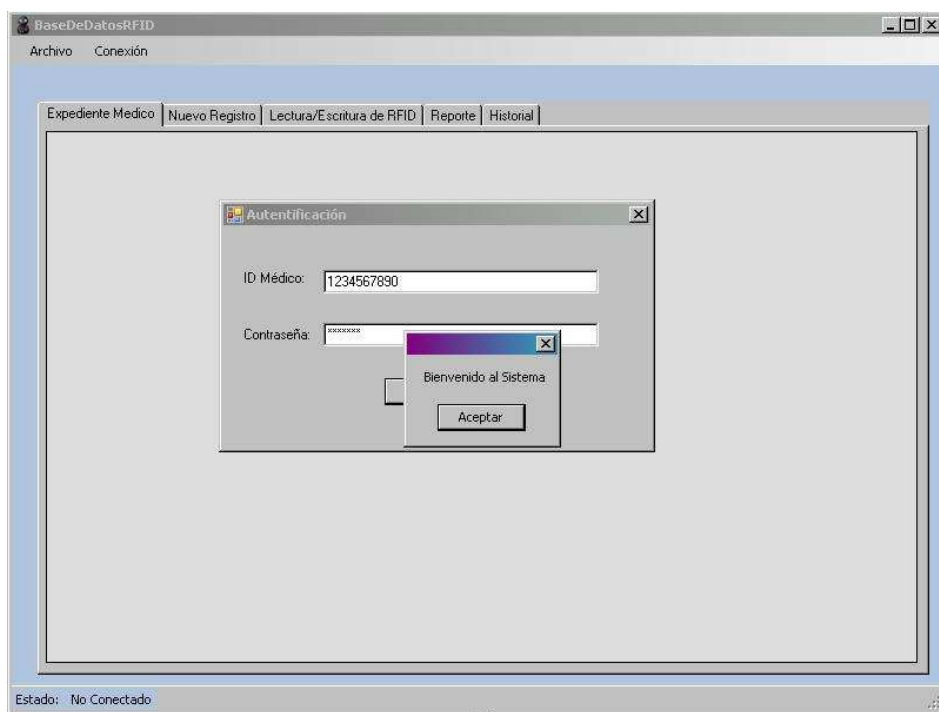


Figura 4.1.1 Acceso al Sistema con Éxito

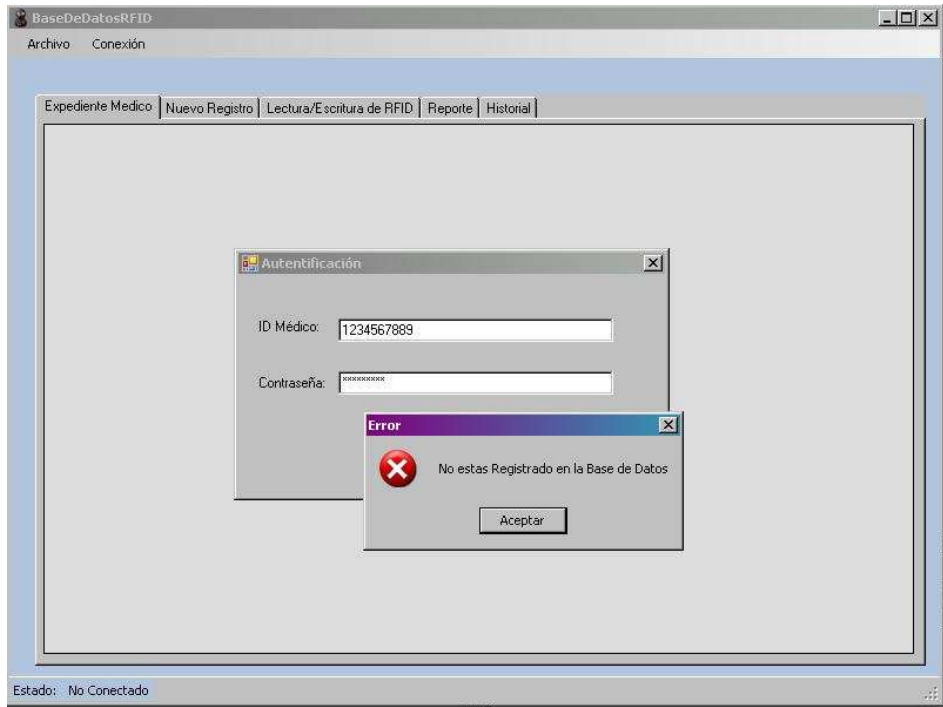


Figura 4.1.2 Acceso al Sistema con Datos Erróneos

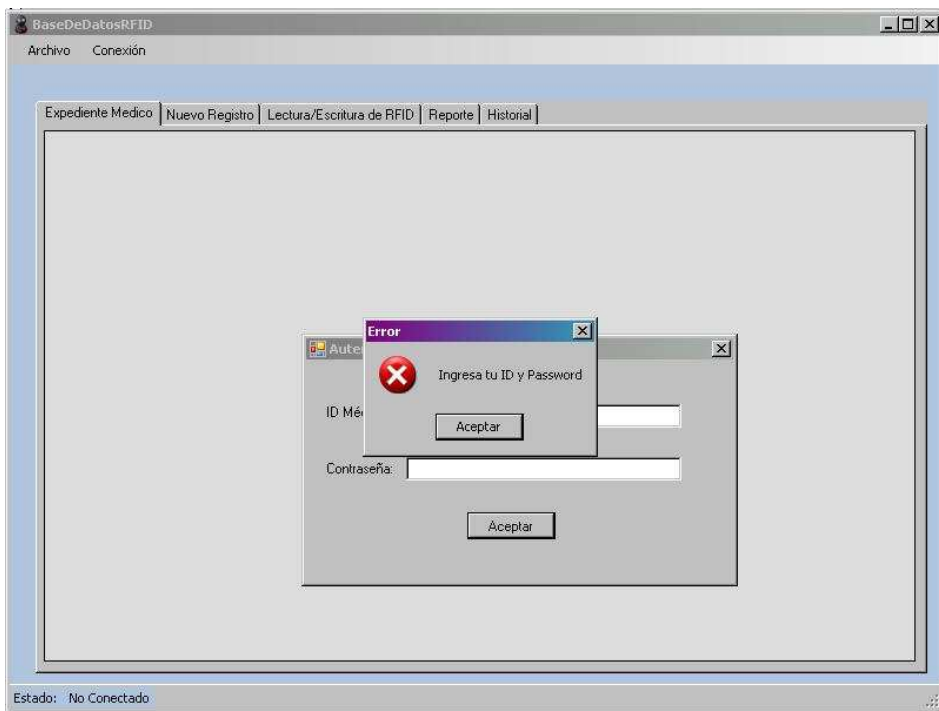


Figura 4.1.3 Acceso al Sistema sin Ingresar Datos

Después de acceder al sistema proseguimos a configurar puerto, en la computadora que se uso el adaptador USB-serial se utilizo el puerto COM 2, ya que era el único disponible, ver figura 4.1.4, una vez que esta configurado nuestro puerto se hizo la prueba para conectar al sistema y fue todo un éxito como lo podemos observar en la figura 4.1.5.

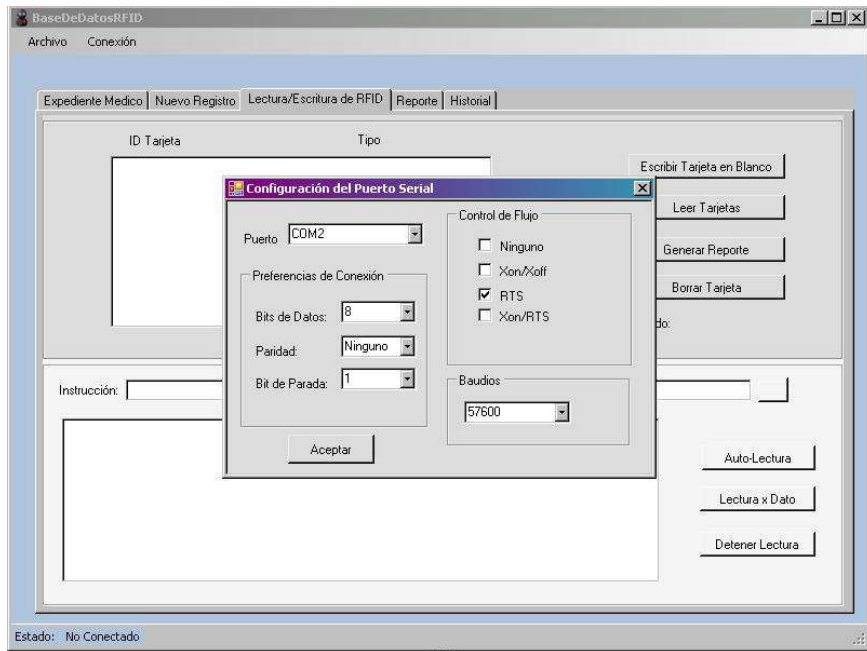


Figura 4.1.4 Configuración del Puerto Serial

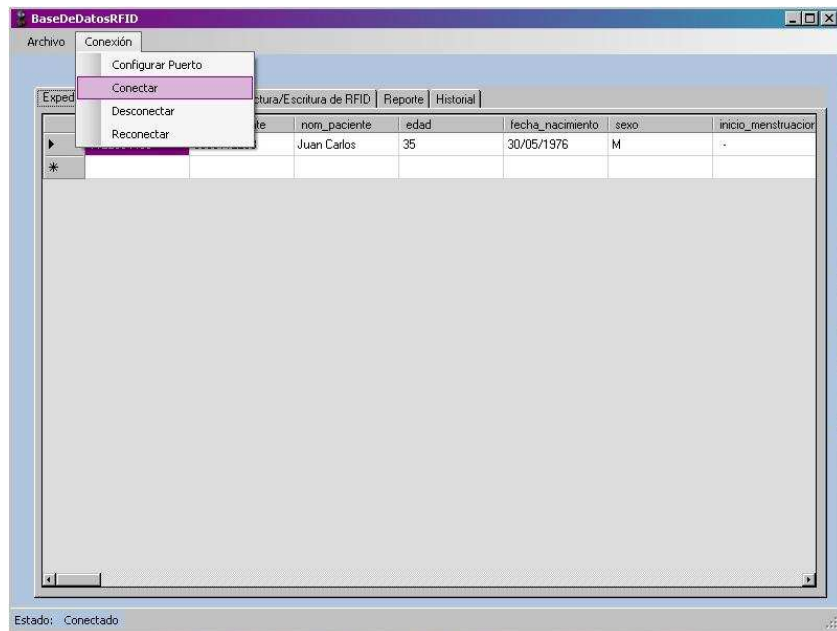


Figura 4.1.5 Conectando con el Puerto Serial

Para poder escribir información en la tarjeta es necesario primero guardar la información en la base de datos, en la pestaña de nuevo registro se da de alta y automáticamente nos aparecerá en el apartado de expediente médico como podemos ver en las figuras 4.1.6 ,4.1.7 y 4.1.8.

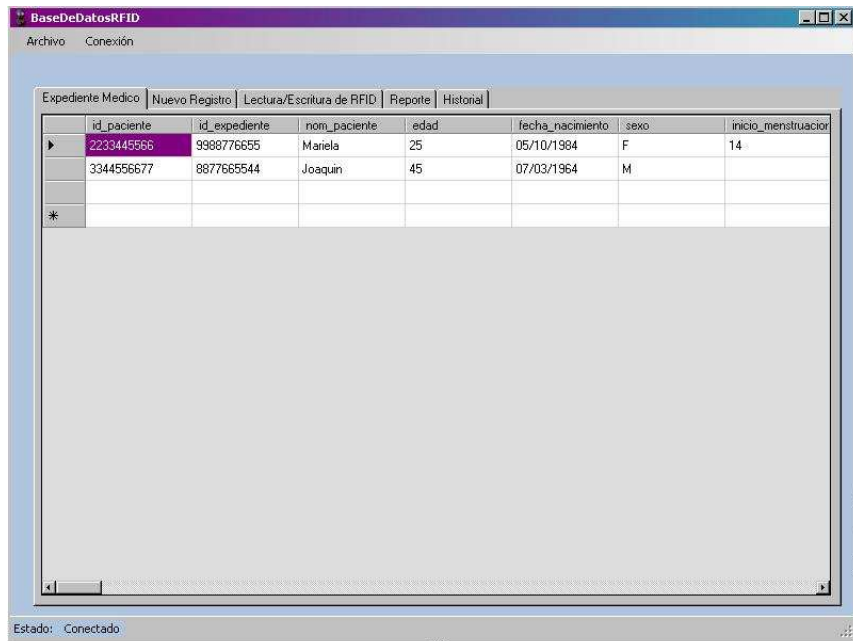


Figura 4.1.6 Registros del sistema sin aun ingresar un registro nuevo

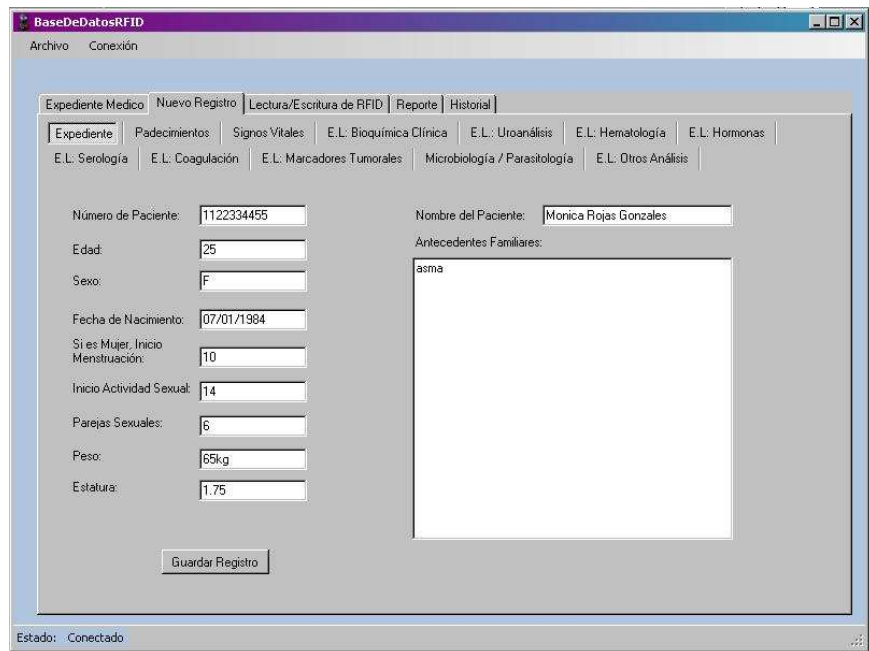


Figura 4.1.7 Ingresando un registro al sistema

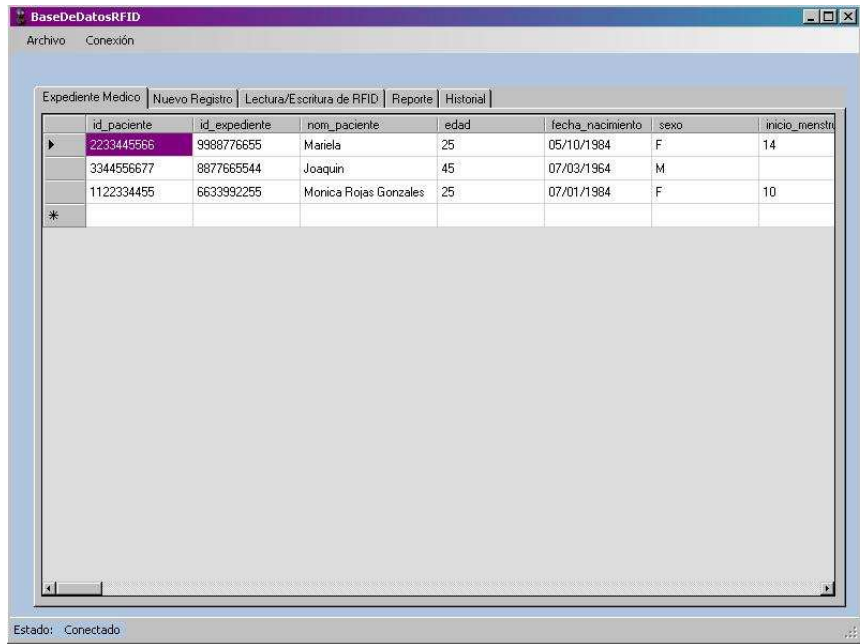


Figura 4.1.8 Registro ingresado con éxito

Ahora se procederá a grabar la información en la etiqueta pero es importante que primero se lea la etiqueta como se observa en la figura 4.1.9, recordemos que cada etiqueta tiene un identificador único, y ese identificador único es el que aparece en la figuras anterior mente mencionada.

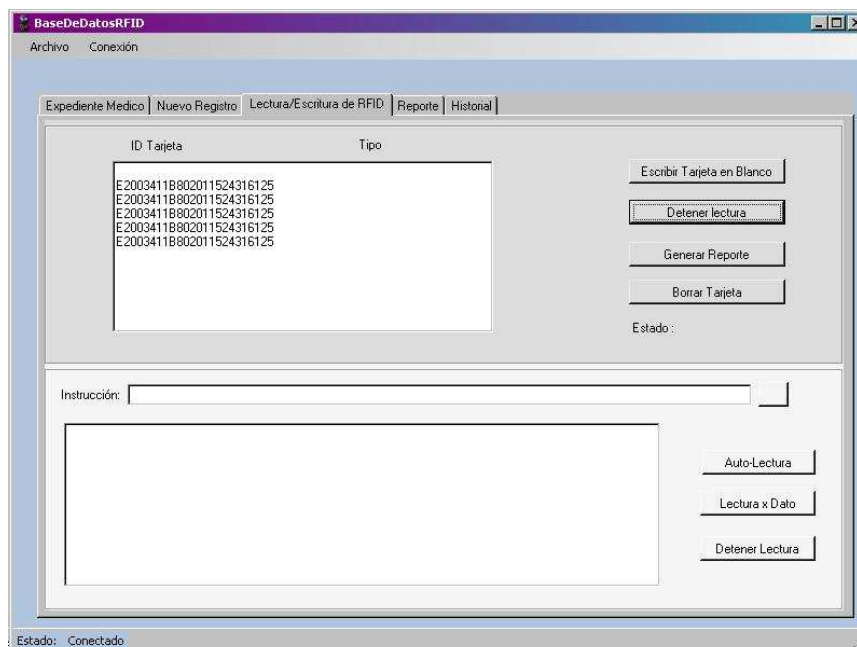


Figura 4.1.9 Lectura de las etiquetas RFID

Finalmente se procede a grabar la etiqueta con el registro anteriormente llenado, si recuerdan nuestro numero de paciente es el 1122334455 y es la información que se esta grabando en la etiqueta como en la figura 4.1.10, después de eso se finaliza la lectura de las etiquetas 4.1.11.

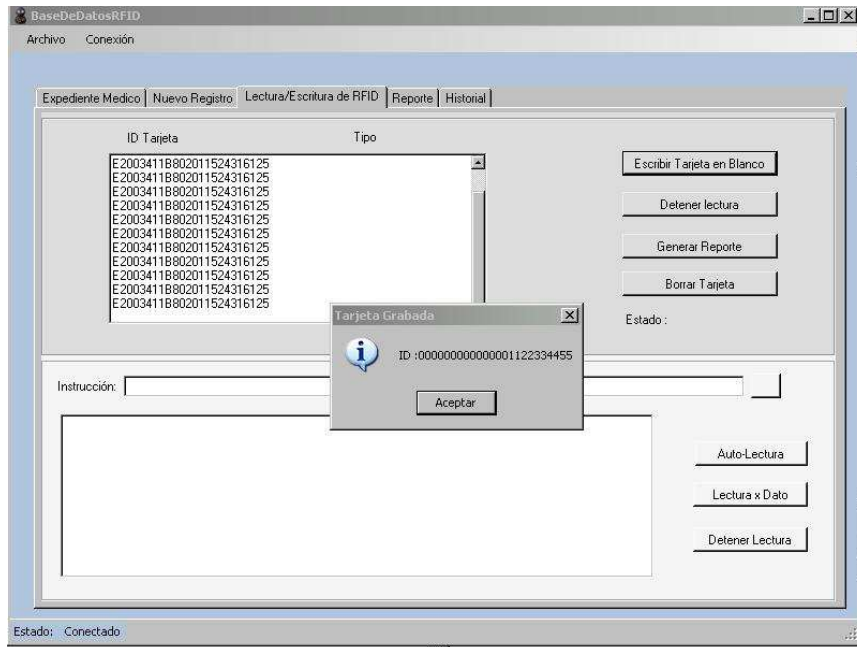


Figura 4.1.10 Información Grabada con éxito en la etiqueta

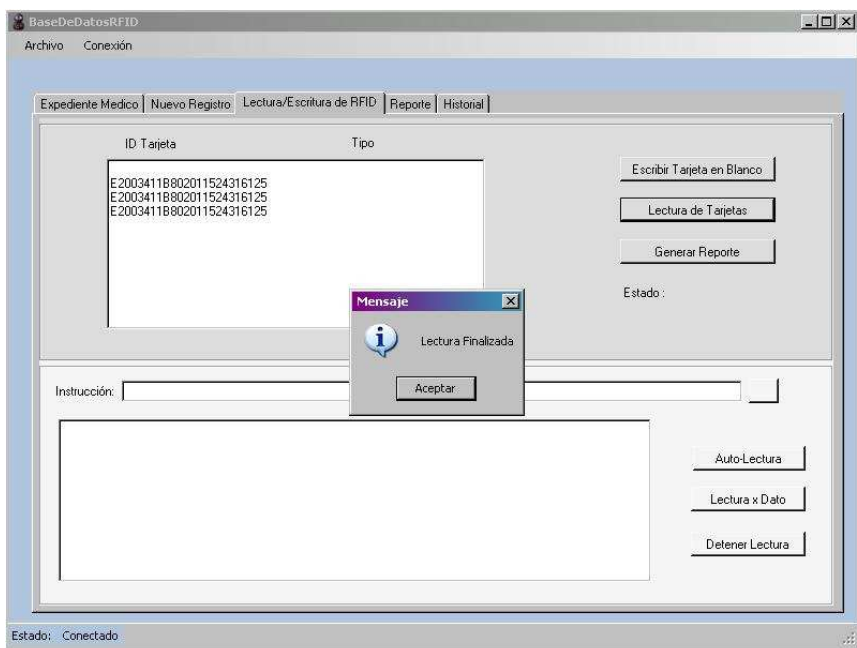


Figura 4.1.11 Finalización de la lectura de la etiqueta

## CONCLUSIONES

La realización del sistema fue satisfactoria, se desarrolló un sistema haciendo uso de la tecnología RFID y las bases de datos centralizadas para ayudar al manejo de la información que utilizan los médicos en los hospitales.

Los principales beneficios de este proyecto hacia los hospitales, son la actualización y optimización del manejo de los archivos clínicos, si hablamos de un medio electrónico, con la tecnología que hoy en día se puede encontrar que es posible obtener una reducción en costos.

Una reducción de costos es el ahorro de tiempo en cuanto a la consulta de expedientes por parte de los médicos, ahorro en gastos para la administración del hospital, ya que toda la información se tendrá en la base de datos lo cual hace que ya no se tenga que gastar en papel, otro punto importante de mencionar es la organización, el médico podrá saber de una manera sencilla que es lo que sucede con el paciente con solo acceder a su expediente.

Otro beneficio importante es tener un monitoreo constante del paciente en tiempo real, cabe mencionar que el sistema es una fuente totalmente segura en cuanto a la privacidad y calidad de la información que se esta maneja.

En general el uso de la tecnología RFID es de gran utilidad para el diferentes áreas como lo puede ser el área de la salud, en esta área, los médicos o los encargados de recopilar la información en el hospital en cuestión tendrán control de la información y los pacientes, también se puede ver como una solución esencial del cuidado médico para los pacientes, incluyendo pacientes en estado grave de salud en salas de urgencia.

## BIBLIOGRAFIA

1. <http://www.rfidjournals.com>
2. **Klaus Finkenzeller**, “*RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification*”, Ed. John Wiley & Sons, Año 2003.
3. **Syed Ahson, Mohammad Ilyas**, “*RFID Handbook: Applications, Technology, Security, and Privacy*”, Ed. CRC Press, London New York, Año 2008, Páginas 13-15.
4. **Bill Glover and Himanshu Bhatt**, “*RFID Essentials*”, Ed. O'Reilly, Año 2006, 1-19.
5. **Sandip Lahiri**, “*RFID Sourcebook*”, Ed. Prentice Hall, Massachusetts E.E.U.U, Año 2005.
6. <http://www.e-health-insider.com/>
7. **Alarcón Raúl**, “*Diseño Orientado a Objetos con UML*”, Ed. Grupo Eidos, Año 2002.
8. **Powell Gavin**, “*Database Design*”, Ed. Wiley Publishing, Año 2006
9. **Elmarsí Ramez A., Shamkant B. Navathe**, “*Fundamentos de Sistemas de Base de Datos*”, 3ra Edición, Ed. Addison Wesley, Año 2002.
10. **Sinan Si Alhir**, “*Learning UML*”, Ed. O'Reilly, Año Julio 2003.
11. **Schmuller Joseph**, “*Aprendiendo UML en 24 Horas*”, Ed. Prentice Hall, Año 2006. 13.
12. Barker Jacquie, Palmer Grant, “*Beginning C objects: From Concept to Code*”.Ed..Apres, Año 2004.
13. **Kenneth C. Louden**, “*Programming Languages: Principles and Practice*”, Ed. Thomson Brooks/cole, Año 2003, 2a Edición
14. **Enrique Alba Torres Ríos, Andrés Rubio del Río**, “*Programación Orientada a Objetos con C++*”.
15. **Ferguson, Patterson, Beres** “*La Biblia de C#*”, Ed. Anaya, Año 2003
16. [http://es.wikipedia.org/wiki/Ambiente\\_integrado\\_de\\_desarrollo](http://es.wikipedia.org/wiki/Ambiente_integrado_de_desarrollo)
17. [http://www.idtechex.com/research/articles/an\\_introduction\\_to\\_epc\\_gen\\_2\\_tags\\_00000168.asp](http://www.idtechex.com/research/articles/an_introduction_to_epc_gen_2_tags_00000168.asp)
18. SAMSys, “*User Guides, Low Power UHF Reader Module*”, SAMSys, 2005, 1ra Edición.
19. SAMSys, “*Reference Guide, Low Power UHF Reader Module*”, SAMSys, 2005, 1ra Edición.