

RESUMEN

IDENTIFICACIÓN DE PACIENTES CON TECNOLOGÍA RFID

por

Rafael Vázquez García

Asesor principal: Carlos Emilio Hernández Rentería

RESUMEN DE PROYECTO DE POSGRADO

Universidad de Montemorelos

Facultad de Ingeniería y Tecnología

Título: IDENTIFICACIÓN DE PACIENTES CON TECNOLOGÍA RFID

Nombre del investigador: Rafael Vázquez García

Asesor principal: Carlos Emilio Hernández Rentería, Maestría en Teleinformática

Fecha de terminación: Mayo de 2013

Problema

¿Cómo evitar la información equívoca en la identificación del paciente hospitalizado? ¿Qué tecnología puede ayudar a suplir y optimizar la forma tradicional de identificar pacientes?

Método

Existen protocolos y normas para la identificación de pacientes mediante el uso de brazaletes con un tablero de datos escritos a mano o un código de barras. El personal que atiende a cada paciente confirma la información del tablero de forma verbal. Sin embargo, identificar al paciente pueden presentarse problemas para hacer coincidir la información del tablero preguntando verbalmente ya que el paciente se puede encontrar indispuesto,

indiferente, o sedado, razones que aumentan el riesgo de una identificación equívoca y acarrear problemas como administración errónea del medicamento, intervenciones quirúrgicas innecesarias, etc. Como parte de esta indagaron y creación del proyecto, se investigaron diferentes tecnologías para la identificación de personas u objetos y se decidió utilizar la tecnología que disminuye el riesgo de información equivocada.

Conclusiones

Con base en los puntos expuestos anteriormente y la importancia de la correcta identificación del paciente, es necesaria una herramienta que optimice esta tarea y que sea inmune a los problemas que se pueden presentar en la forma tradicional de identificar a un paciente.

Universidad de Morelos
Facultad de Ingeniería y Tecnología

IDENTIFICACIÓN DE PACIENTES CON TECNOLOGÍA RFID

Proyecto
presentado en cumplimiento parcial
de los requisitos para el grado de
Maestría en Ciencias Computacionales

por

Rafael Vázquez García

Mayo de 2013



UNIVERSIDAD DE MONTEMORELOS

FACULTAD DE INGENIERÍA Y TECNOLOGÍA
MAESTRIA EN CIENCIAS COMPUTACIONALES

ACTA DE DEFENSA DE TESIS / PROYECTO

En la ciudad de Montemorelos, Nuevo León, siendo las 16:00 horas del día 28 de Mayo del 2013
se reunieron en la Aula Audiovisual de Sistemas los miembros del Comité Examinador de la recepción
de grado de Maestría en Ciencias Computacionales con acentuación en Redes
del alumno Rafel Vázquez García presentada en la modalidad de Proyecto Tesis

Dicho Comité Examinador estuvo integrado por las siguientes personas:

Director de Protocolo.:	<u>Dra. Raquel B. de Korniejczuk</u>	Firma	
Asesor principal:	<u>M.T. Carlos E. Hernández Rentería</u>	Firma	
Asesor:	<u>M.C. Alejandro Garcia Mendoza</u>	Firma	
Asesor:	<u>M.C. Saulo Hernández Osoria</u>	Firma	
Examinador Externo:	<u>M.C. Andrés Carballo M.</u>	Firma	

Se procedió a examinar al sustentante y siendo las 16:52 horas, el dictamen fue:

- Aprobado sin modificaciones
- Aprobado con modificaciones menores
- Aprobado con modificaciones mayores
- No aprobado

DEDICATORIA

A mi mamá, Gladys García, por su amor y apoyo incondicional.

A mi papá, Marco Vázquez, por sus enseñanzas y formación. Gracias a su ejemplo, soy una persona de bien.

A mis amigos Felipe, Abel, Mitzi, Omar, Evi y mi novia Rubí, porque con ellos he vivido las mayores experiencias de mi vida, y juntos hemos aprendido lo que no se aprende en las aulas.

A mi asesor principal, Carlos Hernández, por su guía y por su apoyo a la realización de este proyecto. Por su amistad.

Al profesor Saulo Hernández por su constante asesoría y amistad.

TABLA DE CONTENIDO

LISTA DE FIGURAS	vi
Capítulo	
I. INTRODUCCIÓN	1
Antecedentes	1
Antecedentes de la identificación de pacientes	1
Antecedentes de la tecnología RFID en sector salud	2
Justificación	6
Definición del problema	7
Declaración del problema	7
Objetivos	7
Limitaciones	8
Delimitaciones	8
Materiales, métodos y técnica	8
Definición de términos	9
II. MARCO TEÓRICO	11
Identificación de pacientes	11
Tecnologías	14
¿Qué es y cómo funciona RFID?	14
Componentes de un sistema RFID	15
Etiquetas RFID	17
Estándares y protocolos	19
Arduino	24
¿Qué es Arduino?	24
Arduino software	26
Arduino hardware	26
Arduino Ethernet	27
PHP	28
MySQL	30
Red de datos TCP/IP	30
Capa física	32
Capa de conexión	33
Modelo de redes jerárquicas	33

Capa de acceso	34
Capa de distribución	34
Capa núcleo	35
LAN	35
VLAN	36
III. METODOLOGÍA	38
Características del proyecto	38
Planificación	39
Modelo de tres capas para el sistema RFID	40
Capa de infraestructura	42
Requerimientos de la red	43
Capa de hardware RFID	44
Arduino Ethernet + RFID	45
Adquiriendo los datos de las etiquetas RFID	45
Configuración IP Arduino	48
Cliente web incrustado	49
Enviando la información al servidor web	51
Recibiendo información desde Arduino	52
Archivos PHP y conexión con la base de datos	53
Capa de aplicaciones RFID	53
Consulta de la identidad del paciente por medio del portal web	54
IV. CONCLUSIONES	60
Resumen	60
Conclusiones	61
Resultados	62
V. TRABAJOS FUTUROS	64
Apéndice	
A. LECTOR RFID & CLIENTE WEB	65
B. ARDUINO.PHP	68
LISTA DE REFERENCIAS	70

LISTA DE FIGURAS

1.	Diseño de pulseras en HONADOMANI.	13
2.	Componentes de un sistema RIFD.	15
3.	Etiqueta RFID con antenas helicoidales para 13.56 MHz.	16
4.	Lectura de etiquetas.	17
5.	Componentes de una etiqueta RFID.	19
6.	Ejemplo de espectro ensanchado por salto de frecuencia.	20
7.	Lector RFID con protocolo EM4001.	21
8.	Modulaciones SSB-ASK, PR-ASK.	22
9.	Mecanismo antilisión EPCglobal Class 1 Gen 2.	23
10.	Interacción de sensores.	26
11.	Placa Arduino UNO.	27
12.	Placa Arduino Ethernet.	28
13.	Proceso al visitar una página con PHP.	29
14.	Modelo de capas TCP/IP.	31
15.	Encapsulación.	32
16.	Subcapas de acceso a la red.	32
17.	Modelo de redes jerárquicas.	34
18.	Topología logica LAN.	36

19.	Topología logica VLAN.	37
20.	Modelo de tres capas del sistema RFID.	41
21.	Modelo jerárquico y modelo de tres capas del sistema RFID.	41
22.	protocolos 802.2 y 802.3.	43
23.	Parámetros para comunicación en la red en el lector RFID.	43
24.	Red jerárquica con VLAN.	44
25.	Interrogatorio RFID.	46
26.	Diagrama de pines para la comunicación entre el interrogador RFID y la placa electrónica Arduino.	46
27.	Programación para la comunicación del interrogador y la placa Arduino.	48
28.	Configuración de IP de la placa Arduino.	49
29.	Configuración de un cliente web para el envío de información al servidor web.	50
30.	Monitor serial de la placa electrónica Arduino mostrando la identidad del tag.	50
31.	Red TCP/IP para la comunicación entre el cliente web en Arduino y el servidor web.	51
32.	Captura de tramas entre en cliente web y el servidor web.	52
33.	Contenido del stream de datos.	52
34.	Log de servicio en el software de servidor web.	53
35.	Archivo Arduino2.php.	54
36.	Consulta para depositar el valor del tag en la base de datos.	54
37.	Diagrama de usuario y privilegios.	55
38.	Proceso de depósito de la identidad en la tabla relacionada.	56

39.	Consulta de la identidad iniciando sesión.	57
40.	Selección de lector RIFD.	57
41.	Portal web para la identificación del paciente mostrando la ultima lectura de interrogador RFID accediendo desde una computadora.	58
42.	Acceso al portal web desde un dispositivo móvil.	59

CAPÍTULO I

INTRODUCCIÓN

Antecedentes

Antecedentes de la identificación de pacientes

La identificación de pacientes es un procedimiento que permite a los profesionales de la salud tener la certeza de la identidad del paciente al que se está atendiendo, identificar al paciente es la primera norma en la asistencia segura. Es responsabilidad de todo profesional sanitario identificar al paciente siguiendo las políticas estándares establecidas y no debe trasladar a nadie ni comenzar ningún procedimiento sin seguir el protocolo de identificación, ni siquiera si el profesional conoce al paciente. La identificación debe realizarse en el momento del TRIAGE (método de la medicina de emergencias y desastres para la selección y clasificación de los pacientes basándose en las prioridades de atención) en el servicio de urgencias, al ingreso a una unidad de hospitalización o en la sala de partos. Se debe informar al paciente y la familia sobre la importancia de la identificación (Nebrada, 1998).

Los elementos utilizados para identificar al paciente pueden ser diversos, siendo la pulsera el más común y universal. En la actualidad la pulsera está confeccionada con una etiqueta identificativa con los datos del paciente, pulseras con un código de barras o tarjetas identificativas.

Existe un método en donde el paciente es identificado a través de una banda impresa colocada en el muñeca o el tobillo del paciente, lleva el nombre y el número de identificación interno del hospital y el nombre del médico que indico su internación. Las bandas tienen colores codificados que indican alergias u otros problemas. La tarjeta de identificación del paciente se usa para marcar todos los papeles (historia clínica, órdenes, estudios, etc.) y deben de coincidir con el brazalete. Esta tarjeta debe estar fijada a la historia clínica durante el traslado y debe permanecer con el paciente cada vez que se le traslade fuera de alguna unidad.

La forma adecuada de identificar al paciente es la siguiente: (a) Examinar la banda de identificación. Comparar el nombre y el número con los que Figuran en la historia clínica, (b) comparando el nombre completo, evitando llamar al paciente por su nombre hasta que él mismo confirme quién es y (c) pedir al paciente que le diga qué procedimiento le van a realizar (Fuller, 2007).

Antecedentes de la tecnología RFID en sector salud

La identificación por radio frecuencia (RFID), en ocasiones llamada identificación automática (Auto ID), tiene alrededor de 50 años de estar en uso. El primer despliegue de esta tecnología fue por la Royal Air Force (RAF), durante la segunda guerra mundial, como forma de distinguir a sus aviones de los aviones enemigos. La tecnología RFID permite almacenamiento inalámbrico y recolección automática de datos (Fanberg, 2004). Posterior a la guerra entre los años 50 y 60 se inició la investigación y el desarrollo de aplicaciones para esta tecnología, pero fue hasta 1970 que Marco Cardullo presentó la primera patente

de una etiqueta RFID activa con una memoria regrabable (Landt, 2001). En la actualidad la tecnología RFID tiene muchas aplicaciones en las áreas de automatización e industria. Actualmente se ha orientando a la optimización de diferentes procesos como (a) control de acceso, (b) gestión de activos, (c) cadenas de suministro, (d) control de la producción y (e) calidad (Corrales, Rivas y Salichs, 2007).

La tecnología RFID posee un amplio campo de aplicación y puede aportar un valor agregado a muchos sectores de la industria. Un estudio realizado en 2006 por la Universidad Politécnica de Milán la cual realizó un análisis de la capacidad y alcance de las diferentes aplicaciones que puede tener RFID, obtuvo como conclusión que más de la mitad de las aplicaciones (60 %) están situadas en el sector de servicios, incluyendo el sector sanitario y de salud (Portillo, Bermejo y Bernardos, 2008). Las principales aplicaciones que tiene la tecnología RFID en el sector salud son (a) rastreo, (b) identificación y (c) monitoreo. La aplicación de rastreo puede ser usada en equipos, medicamentos, implantes, flujo de pacientes, mantenimiento de equipos y disponibilidad de camas y cuartos. En un caso práctico la farmacéutica Pfizer gasto 5 millones de dólares en etiquetado RFID del medicamento Viagra para evitar la falsificación (FDA, 2005).

Otro ejemplo es en el Hospital Oncológico Costa, el paciente entra en la unidad para recibir una medicación de forma ambulatoria y este tipo de atención no requiere hospitalización, pero intervienen varios servicios hospitalarios. Uno de estos servicios es farmacia, donde se preparan los fármacos a administrar y otro es el banco de sangre donde llegan las muestras de sangre para realizar los análisis a los pacientes que necesitarán una transfusión y así proveerlos de bolsas de sangre. Entran en juego factores críticos como

la administración segura de medicamentos (orden de aplicación, caducidad, fecha programada, etc.) y la seguridad en el momento que se realiza una transfusión. Se implementó un sistema de identificación, mediante el uso de pulseras RFID para mejorar los procesos de administración de medicamento y transfusión de sangre a los pacientes. Para que la duración de los procesos de identificación de pacientes, administración de medicamentos y análisis sanguíneo sean reducidos, también se ha optado por identificar tanto los medicamentos que vienen de farmacia como las muestras que viajan hacia el banco de sangre mediante la tecnología RFID en sustitución de la lectura de códigos de barras. En el proceso, las lecturas de los distintos dispositivos son realizadas por el personal sanitario mediante dispositivos de mano tipo PDAs provistos de los lectores RFID necesarios. El sistema incorpora una aplicación que va guiando al profesional en la administración de la medicación para cumplir de forma segura con una identificación correcta de (a) paciente, (b) medicamento, (c) dosis, (d) vía y (f) horario. En todo momento, el acceso a datos del paciente se hace de forma segura al sistema de información del hospital evitando errores (Gutierrez, 2008).

Otro caso de aplicación en el sector salud tuvo lugar en el Hospital Greenville del Centro Médico Universitario de Carolina del Sur, se implementó el etiquetado RFID en sondas quirúrgicas que tienen un costo individual de miles de dolares, con la finalidad de que no fueran desechadas. A causa de su pequeño tamaño, en ocasiones se perdían en las sabanas sucias de las camas o artículos desechables que se eliminaban después de los procedimientos quirúrgicos y eran arrojadas a la basura. Para resolver el problema, se instaló un lector RFID en el vestíbulo del hospital por donde pasaban todos los materiales

desechables y la ropa sucia antes de ser depositados en la basura. Al poner en funcionamiento el sistema, se detectó una sonda de \$17.000 dólares que se encontraba escondida en el fondo de un carro lleno de artículos sucios en camino a ser desechados (Bacheldor, 2009).

Los dispositivos médicos portátiles representan un recurso sumamente importante para la asistencia sanitaria, el equipo de seguimiento de uso de identificación por radiofrecuencia proporciona una solución prometedora para los problemas encontrados en la localización de equipos portátiles. En el Hospital Royal Alexandra en Paisley, Reino Unido, se llevó a cabo una prueba del uso de la tecnología RFID que implicó la instalación temporal de tres lectores y etiquetas activas para diferentes dispositivos médicos. Los lectores activos y el sistema de computadoras estaban vinculados con una red de datos a la medida. Se pusieron a prueba las etiquetas y lectores de dos fabricantes diferentes. Debido a que la carcasa de un dispositivo médico interfiere con la señal recibida por una etiqueta, se encontraron dificultades de fiabilidad en las pruebas de la tecnología del primer fabricante. Se obtuvieron mejores resultados cuando se utilizó tecnología del segundo fabricante con una tasa de error global de 12,3%, ya que fueron diseñadas específicamente para ser usadas en la superficie de equipos médicos (Britton, 2007).

Un caso más es el estudio de tiempos y movimientos que se llevó a cabo con las enfermeras de 36 unidades hospitalarias con unidades médico-quirúrgicas, donde se realizó un monitoreo y registro de actividades para saber en qué consumían más tiempo de su jornada laboral. El monitoreo de las enfermeras se realizó con dispositivos PDA (Personal Digital Assistant) y lectores RFID. El análisis de monitoreo demostró áreas específicas pa-

ra mejorar: (a) documentación, (b) administración de medicamentos y (c) coordinación de la atención. Los cambios en la forma de realizar las tareas cotidianas, el uso de tecnología, procesos y la organización de la unidad, demostraron un cambio en el trabajo de las enfermeras (Hendrich, Chow, Skierczynski y Lu, 2008). Existen dificultades al implementar RFID en el sector salud, una de ellas durante la instalación de cableado en construcciones muy antiguas y la instalación de otras tecnologías en paredes y techos de metal que pueden afectar las señales. El ruido electrónico debe ser identificado y filtrado (Kumar, Livermont y McKewan, 2009). En la mayoría de las implementaciones hospitalarias de sistemas con lectores RFID se opta operar en HF (High Frequency), o alta frecuencia, pues ésta presenta una mejor tolerancia a la presencia de metales y líquidos. Su rango de cobertura y su pequeño tamaño facilitan el empotrado en los equipos y monitorear su ubicación. Si se desea, en cambio, hacer un seguimiento y monitoreo del personal y bienes, o se necesita conocer su ubicación en cada momento e incluso un historial de ubicaciones, se recomienda ampliamente utilizar etiquetas RFID activas que posean mayor alcance (Portillo, Bermejo y Bernardos, 2008).

Justificación

Este proyecto de investigación tiene la intención de proponer un sistema para la identificación del paciente utilizando tecnología RFID, junto con un portal web de fácil acceso desde cualquier dispositivo que se conecte a una red de datos y un explorador web. Todas las operaciones que se realizan en este sistema se encapsulan en capas y se dividen en módulos para explicar su funcionamiento.

Gracias a las bondades de la tecnología RFID se pretende dar solución a las situaciones en las que el paciente pueda impedir su identificación, al evitar la necesidad de una línea de vista entre el dispositivo identificador y el dispositivo lector. Se pretende eliminar la confirmación verbal o visual de un brazalete con información para la identificación del paciente, la obtención de estos datos deben ser inequívocos y podrán ser consultados por medio de un portal web de fácil acceso desde cualquier dispositivo que se conecte a internet y cuente con un explorador web.

Definición del problema

Con la finalidad de identificar a un paciente de manera inequívoca y consultar esta información, el presente proyecto busca diseñar, desarrollar e implementar conceptualmente un sistema de identificación de pacientes con tecnología RFID y un portal web, además de que el costo total del mismo sea accesible y tenga la capacidad de integrarse con otros proyectos.

Declaración del problema

El presente proyecto propone un sistema para la identificación de pacientes con un software de consulta y dispositivos periféricos para corroborar la identidad de un paciente utilizando la tecnología RFID como mecanismo de identificación.

Objetivos

Diseñar y desarrollar un lector de RFID con una interfaz Ethernet para comunicarse en una red de datos TCP/IP. Además de la implementación de un portal web

para la consulta de la identidad de los pacientes, con capacidad de acceso desde cualquier dispositivo móvil y exploradores web.

Limitaciones

El presente estudio tendrá las siguientes limitaciones:

1. El lector RFID cuenta con un alcance de lectura corto.
2. El protocolo que utiliza el lector RFID para leer etiquetas permite sólo leer una etiqueta por vez.
3. Por los inconvenientes que implicaba comprar los componentes electrónicos en el extranjero y la falta de recursos económicos sólo se compraron componentes dentro del territorio nacional. Por consiguiente los lectores no cuentan con la robustez deseada.

Delimitaciones

Con el fin de alcanzar una solución oportuna y práctica al problema planteado, la investigación se delimitó de la siguiente manera:

1. No se creará un dispositivo final con el lector RFID donde todos los componentes del lector se encuentren en un solo circuito impreso (PCB). Sólo se creará un prototipo funcional.

Materiales, métodos y técnica

Los materiales con los que se trabajó en este proyecto fueron placa electrónica Arduino Ethernet, la cual contiene los circuitos para procesar las señales; un convertidor usb a serial, para programar la placa electrónica Arduino desde la computadora; lector

de RFID ID-20, el cual utiliza para leer el contenido de las etiquetas de RFID; pulseras, tarjetas y cápsulas de vidrio RFID, que contiene códigos únicos para su identificación; cables de red, se emplean para establecer una comunicación entre los dispositivos finales; sistema operativo Ubuntu server, con servicios de web, PHP, MySQL como base de datos y log de servicios.

Métodos: (a) estudio de tecnologías, (b) desarrollo de un lector de RFID que deposite la información por medio de una red TCP/IP a un servidor local o remoto, (c) desarrollo de un portal web y (d) prueba concepto.

Definición de términos

Se presentan a continuación las definiciones de los términos que serán utilizados en este documento:

1. RFID: es un sistema de almacenamiento y recuperación de datos remoto que usa dispositivos denominados etiquetas, tarjetas, transpondedores o tags RFID. El propósito fundamental de la tecnología RFID es transmitir la identidad de un objeto (similar a un número de serie único) mediante ondas de radio.

2. Tag RFID: un tag tiene la capacidad de almacenar tanto la información de identificación como información adicional.

3. Lector RFID: compuesto por una antena, un transceptor y un decodificador. El lector envía periódicamente señales para ver si hay alguna etiqueta en sus inmediaciones. Cuando capta la señal de una etiqueta (la cual contiene la información de identificación de ésta), extrae la información y se la pasa al subsistema de procesamiento de datos.

4. TRIAGE: es un método de la medicina de emergencias y desastres para la selección y clasificación de los pacientes basándose en las prioridades de atención, privilegiando la posibilidad de supervivencia, de acuerdo a las necesidades terapéuticas y los recursos disponibles.

5. Identificación de pacientes: es la acción que asegura y garantiza que se presta la atención a quién corresponde.

6. Red de datos TCP/IP: red basada en el modelo TCP/IP, describe un conjunto de guías generales de diseño e implementación de protocolos de red específicos para permitir que un equipo pueda comunicarse en una red. TCP/IP provee conectividad de extremo a extremo especificando cómo los datos deberían ser (a) formateados, (b) direccionados, (c) transmitidos, (d) enrutados y (e) recibidos por el destinatario.

7. LAN: es la interconexión de una o varias computadoras y periféricos. Antiguamente su extensión estaba limitada físicamente.

8. VLAN: es una subred IP separada de manera lógica. Las VLAN permiten que redes de IP y subredes múltiples existan en la misma red conmutada.

9. Estándar 802.3: describe las funciones de la subcapa MAC y de la capa física.

10. Estándar 802.11: comúnmente llamada Wi-Fi, es un sistema por contención que utiliza un proceso de acceso al medio de acceso múltiple con detección de portadora y prevención de colisiones (CSMA/CA).

CAPÍTULO II

MARCO TEÓRICO

En este capítulo se presentan los conceptos de este proyecto, y los temas fundamentales necesarios para su elaboración. Se mencionan algunas metodologías para la identificación de pacientes, el funcionamiento de RFID, así como la forma en que la información adquirida desde las etiquetas es depositada en una base de datos y cómo es consultada la identidad del paciente a través de un portal web.

Identificación de pacientes

La identificación del paciente es un proceso sumamente importante en diferentes tareas tales como (a) la correcta administración de medicamentos, (b) intervenciones quirúrgicas, (c) transfusión de sangre y (d) traslado de camas. En los párrafos siguientes se describe la importancia de la identificación correcta del paciente en diferentes tareas hospitalarias.

Uno de los puntos de admisión de un paciente a hospitalizar es el departamento de urgencias, los pacientes que acuden a este departamento deben tener un registro lo más adecuado posible dentro de las limitaciones y características que presentan estas admisiones, es necesario que exista un registro centralizado y que se garantice una correcta identificación del paciente consultando esta información desde un registro central o creando

un registro nuevo con número de historia clínica en caso de ser la primera vez que se requiere asistencia. Cabe destacar que la identificación se realiza previo a la asistencia, siempre que sea posible y que las condiciones del paciente lo permitan. En los casos en que no sea posible, se articularán medidas para registrar al paciente de forma provisional y se irán complementando sus datos a medida que se vayan disponiendo de ellos (Tamayo, 2004).

Otra de las tareas importantes es la administración de medicamentos, procedimiento que se repite con mayor frecuencia a lo largo de la jornada de trabajo de una enfermera. Su correcta administración no es fácil como en un principio podría parecer, exige máxima precaución y la seguridad de una técnica correcta, desde la identificación del paciente hasta la última de las anotaciones en las hojas de medicación. Por ello dicho procedimiento es una de las responsabilidades más problemáticas en enfermería (Ramos y Aznar, 1994).

Todas las tareas hospitalarias son importantes pues de ellas depende la salud de los pacientes, cabe destacar la importancia de identificar correctamente a un paciente en las situaciones de petición y recogida de muestras hemoderivadas. Toda transfusión debe quedar registrada en las hojas preparadas para ello, que se incluirán en la historia clínica del paciente. Estas hojas deben contar con todos los datos de identificación del paciente más el número de la bolsa, la fecha, la hora y la firma legible de la enfermera que realiza la transfusión (de la Torre, Martín y Arribas, 1996).

Existen diversos sistemas y protocolos para la identificación de pacientes, cada uno se adapta a las necesidades, capacidades y recursos propios de las instituciones que los emplean. La mayoría de éstos concuerdan en la existencia de datos inequívocos para la

identificación, como son (a) nombre y apellidos, (b) fecha de nacimiento y (c) número de historial clínico o expediente médico del paciente.

Desde el año 2008, en el Hospital Nacional Docente Madre “Niño San Bartolomé” (HONADOMANI), en Lima Perú, se implementó un programa que consiste en objetivos, líneas de acción, estrategias y metodologías, que buscan mejorar el sistema de identificación de los pacientes hospitalizados en la institución y así reducir los posibles errores en sus tareas hospitalarias. Los instrumentos que utilizan para la identificación de un paciente son pulseras de identificación de material de polipropileno en colores y blanco, las cuales utilizan impresión térmica directa (en negro) con los datos inequívocos del paciente, además de información adicional (Montanez, 2008). En la Figura 1 se muestra la información que contiene la pulsera.

Para este sistema de identificación del paciente (ID) por pulsera, se desarrollo un software que se integra con la red de datos del hospital, lo que facilita la identificación de pacientes. Los pasos para la implementación de la pulsera según el HONADOMANI son: (a) se registra a los pacientes en los puntos de acceso como urgencias, (b) se imprime la pulsera y el personal de salud informa al usuario del servicio o responsable acerca de la finalidad del sistema de identificación del paciente, (c) la pulsera de identificación se



Figura 1. Diseño de pulseras en HONADOMANI.

coloca en la muñeca derecha y de no ser posible en la izquierda; si a pesar de ello no es posible, se procede a colocar en el tobillo derecho y si no es posible, en el tobillo izquierdo. La verificación es realizada por el personal de salud antes de realizar cualquier práctica o procedimiento médico, verificando la identidad del paciente.

La identificación de pacientes mediante el uso de brazaletes funciona ya en 19 hospitales de la Comunidad de Valencia, España, esto permite conocer en todo momento la identidad del paciente, con lo que se garantiza la seguridad clínica de todo aquel que ingrese en el hospital. Se trata de un sistema de gran utilidad, recomendado por la Organización Mundial de la Salud y por la Fundación Avedis Donabedian a través de su centro de investigación para la seguridad de los pacientes (Montanez, 2008).

Tecnologías

¿Qué es y cómo funciona RFID?

La identificación por radio RFID (Radio Frequency IDentification) es un sistema de almacenamiento y recuperación de datos remoto que usa un circuito con una antena y una pequeña memoria llamada etiqueta RFID. El propósito fundamental de la tecnología RFID es transmitir la identidad de un objeto (similar a un número de serie único) mediante ondas de radio. La tecnología RFID se agrupa dentro de las denominadas Auto ID (identificación automática) (Thornton et al., 2006).

Las etiquetas RFID son unos dispositivos pequeños que pueden ser adheridos o incorporados a un producto, un animal o una persona. Contienen antenas que les permiten recibir y responder a peticiones por radiofrecuencia desde un lector RFID. Existen dos

tipos de etiquetas, las etiquetas pasivas que no necesitan alimentación eléctrica interna, y las etiquetas activas que requieren el uso de alimentación eléctrica. (Thornton et al., 2006).

Componentes de un sistema RFID

RFID forma parte de las tecnologías inalámbricas y es usada para identificar personas u objetos etiquetados. Como se muestra en la Figura 2 los elementos básicos de un sistema RFID: (a) Tag (transponder), compuesto por un chip semiconductor, una antena y en ocasiones una batería, (b) lector (interrogador), compuesto por una antena, un módulo electrónico de radiofrecuencia y un módulo control, (c) controlador (host), comúnmente se usa una PC o estación de trabajo ejecutando un software de base de datos y control (middleware) (Thornton et al., 2006).

Básicamente, RFID captura e identifica automáticamente la información contenida en etiquetas, tags o transpondedores. En el momento en que estas etiquetas entran en el área de cobertura de un lector RFID, éste envía una señal para que transmita la información almacenada en la memoria. La principal ventaja de este sistema es la forma



Figura 2. Componentes de un sistema RFID.

en la que se obtienen los datos contenidos en la etiqueta mediante radio frecuencia sin la necesidad de contacto físico o una línea de vista entre el dispositivo lector RFID y las etiquetas (Portillo, Bermejo y Bernardos, 2008).

Las etiquetas RFID tienen una amplia variedad de propiedades que son importantes para usos específicos. Una de sus propiedades esenciales es la frecuencia de operación que se extiende desde 125 kHz (baja frecuencia) a 5,8 GHz (microondas). Las etiquetas que trabajan en la frecuencia de 13,56MHz (HF), mostrada en la Figura 3, o frecuencias del rango 860-960 MHz, también son conocidas como UHF (Ultra High Frequency) o ultra alta frecuencia. EPCglobal es un estándar que rige las reglas para identificar productos de forma electrónica mediante un código, que trabaja en la banda UHF, importante en las aplicaciones de logística. Las etiquetas que trabajan en la banda HF son utilizadas principalmente para identificación de piezas u objetos, mientras que las etiquetas que trabajan en UHF son utilizadas con fines logísticos (Hansen y Gillert, 2008).

Los principales parámetros de rendimiento en etiquetas RFID son: (a) El rango de lectura, (b) la velocidad de transmisión de datos y (c) la sensibilidad de los objetos circundantes.

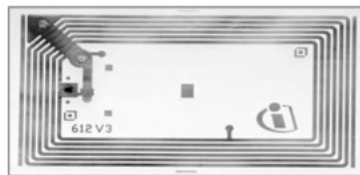


Figura 3. Etiqueta RFID con antenas helicoidales para 13.56 MHz.

Etiquetas RFID

El rango de lectura (véase Figura 4), la capacidad para resistir interferencias ambientales y la idoneidad en las etiquetas RFID depende en mayor parte de (a) la frecuencia de funcionamiento, (b) la intensidad del campo (alimentación) de la antena del lector, (c) la orientación de la etiqueta con respecto a la antena del lector y (d) la forma y tamaño de la antena de la etiqueta. La frecuencia como protocolo de transmisión, incluyendo al protocolo de anticollisión, son decisivos para la velocidad de transmisión de datos y la lectura de una gran cantidad de etiquetas a la vez. Una situación de saturación es cuando varios objetos con etiquetas RFID pasan a través del campo de una antena al mismo tiempo. El campo electromagnético entre la antena del lector y la antena de la etiqueta se llama “interfaz aérea” y las reglas que rigen el intercambio de datos se denomina “protocolo de interfaz aérea”. Todo esto está regido por el estándar EPCglobal Generación 2 (Hansen y Gillert, 2008).

Los transpondedores o etiquetas son clasificados en términos de su capacidad de almacenamiento y los datos almacenados pueden ser modificados (reescritura). Existen

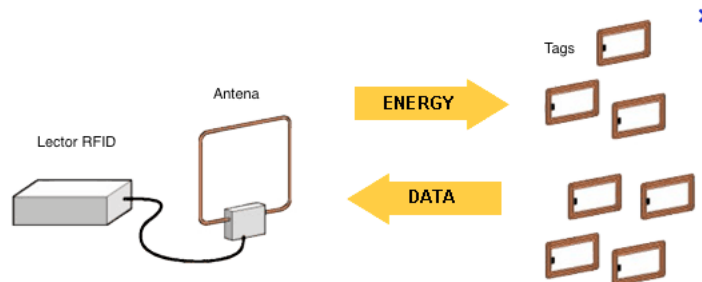


Figura 4. Lectura de etiquetas.

cuatro tipos de transpondedores: (a) Etiqueta de sólo lectura, (b) etiqueta de sólo lectura con un identificador único, (c) etiqueta de una sola escritura, muchas lecturas y (d) etiqueta de lectura-escritura.

Las etiquetas de sólo lectura, es el tipo más sencillo de etiquetas, almacenan exactamente un bit y no contiene un chip. Estas etiquetas de un solo bit son usadas en productos de venta al por menor para evitar el robo. Después de la operación de pago se eliminan en el punto de venta y si permanecen adheridas a los productos, son detectadas por las antenas en las salidas de los establecimientos y activan una alarma en caso de robo. Las etiquetas no contienen información sobre el artículo etiquetado.

Las etiquetas de sólo lectura, con un identificador único, son etiquetas que se les asigna un número de serie único durante su fabricación. Estos números se utilizan para identificar un producto u objeto y no se pueden modificar. Se utiliza una base de datos para asociar el número del artículo con el ID que se encuentra dentro de la etiqueta. Estas son las etiquetas más económicas ya que se pueden volver a utilizar debido a su flexibilidad o método de asociación.

Las etiquetas de una sola escritura, muchas lecturas, pueden ser codificadas una vez con información no modificable que posteriormente se puede leer cuantas veces sea necesario. Se utilizan para almacenar números de artículos y números de serie tales como EPC.

Las etiqueta de lectura-escritura, los datos pueden ser escritos y modificados, se pueden almacenar datos de usuario, instrucciones de manipulación o datos de un proceso. Los datos también se pueden cifrar.

El componente central en una etiqueta RFID es el chip que puede almacenar la información. Se encuentra en el orden de milímetros cuadrados y esencialmente contiene tres partes: (a) Módulo de radiofrecuencia para la generación de la señal y potencia en la obtención de un campo magnético, (b) unidad de control para el procesamiento de órdenes recibidas y (c) unidad de almacenamiento (véase Figura 5).

Estándares y protocolos

La especificación EPCglobal es un conjunto de normas interrelacionadas para las interfaces de hardware, software y datos junto con los servicios básicos que tienen un objetivo común: mejorar la cadena de suministro a través del uso de Códigos de Productos Electrónicos (EPC)(Ahson e Ilias, 2008).

La red EPCglobal consiste en estándares que especifican elementos de hardware y software. Los detalles de las especificaciones de hardware del protocolo de interfaz aérea, la comunicación entre el lector y las etiquetas, son conocidas como UHF Clase 1 Generación-2 v1.0.9.

Las especificaciones del protocolo Class 1 Generación 2 detallan la comunicación entre las etiquetas y el lector RFID, contiene las especificaciones de la interfaz aérea.

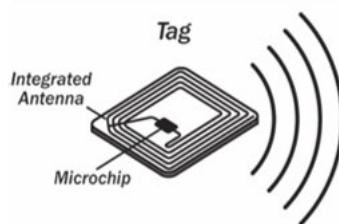


Figura 5. Componentes de una etiqueta RFID.

La comunicación entre lectores y etiquetas se basa en principios de “Reader Talks First”, donde el lector emite las instrucciones para la comunicación de una población de etiquetas que se encuentran dentro del campo RF del lector. El objetivo del protocolo es separar un etiqueta de un entorno de múltiples etiquetas para leer su identidad u otra información almacenada (Ahson e Ilias, 2008).

La frecuencia de transmisión de los lectores RFID opera entre 860 y 960 MHz de acuerdo a las regulaciones locales. Existen tres bandas de interés: (a) Región 1, EU y África, típicamente 865-868 MHz, 2 W frecuencia Ágil, LBT (b) región 2, EU y Canadá, 902-928 MHz, 4 W EIRP espectro ensanchado por salto de frecuencia (veáse Figura 6) y (c) región 3, Asia.

El protocolo EM4001 (veáse Figura 7) rige la comunicación en etiquetas pasivas de corto alcance, utiliza la codificación Manchester y trabaja a 125 kHz.

Las etiquetas deben de entender tres esquemas de modulación diferentes, uno de ellos será seleccionado por el lector y enviado a las etiquetas que están en el campo de RF del lector, como parámetro para establecer la comunicación con las etiquetas. El lector selecciona el esquema de modulación más adecuado basado en sus propios parámetros de

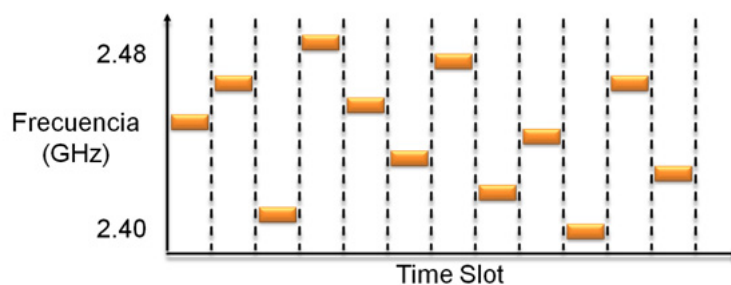


Figura 6. Ejemplo de espectro ensanchado por salto de frecuencia.

funcionamiento, tales como (a) entorno regulatorio, (b) ruido percibido y (c) preferencias establecidas por el integrado de sistema. La Figura 8 muestra los tres diferentes esquemas de modulación (a) Double Sideband Amplitude Shift Keying DSB-ASK, (b) Single Sideband Amplitude Shift Keying SSB-ASK y (c) Phase Reversal Amplitude Shift Keying PR-ASK.

Los comandos que establecen la comunicación con etiquetas también definen los parámetros de comunicación para la respuesta de la etiqueta. Una vez más, estas decisiones son tomadas por el lector en base a las normativas locales, la aplicación requerida y el rendimiento de ruido de las técnicas de comunicación anteriores. Las etiquetas pueden responder a las órdenes del lector utilizando uno de los esquemas de modulación de retrodispersión de etiquetas, ya sea ASK o modulación PSK, seleccionando el formato por el proveedor de etiqueta. Los lectores tienen capacidad demoduladora para cualquiera de los tipos de modulación. Las etiquetas pueden ser controladas por el lector para codificar los datos de retrodispersión, ya sea como banda base FM0 o modulación Miller (Ahson e Ilias, 2008).

La especificación de detalles de los comandos del lector y las respuestas de la eti-

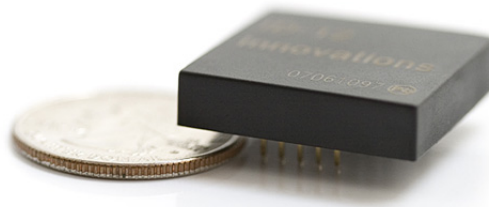


Figura 7. Lector RFID con protocolo EM4001.

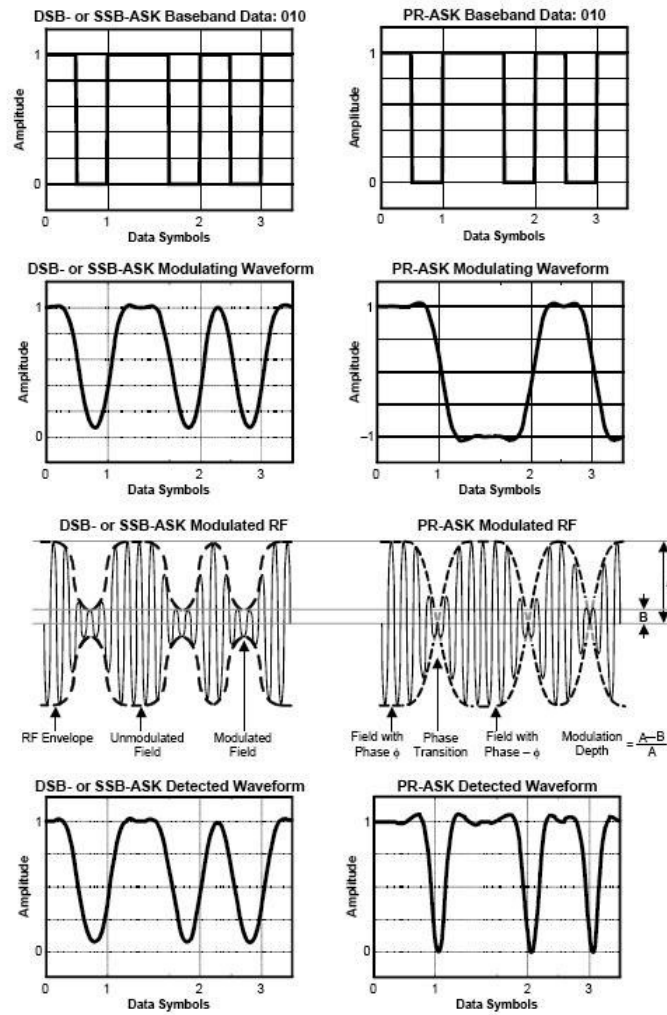


Figura 8. Modulaciones SSB-ASK, PR-ASK.

queta a los comandos se ocupan para la lectura o escritura de datos en la etiqueta. Las etiquetas son identificadas en una población de varias etiquetas por el protocolo anticoli- sión, llamado protocolo Q. Este es un protocolo de respuesta basada en ALOHA, en el cual las etiquetas reciben parámetros anticoli- sión del lector y seleccionan un período al azar para responder. Este período de respuesta se llama ranura, donde en una ronda existen varias ranuras. El tamaño de la ronda (valor Q) es provisto por el lector a las ranuras y

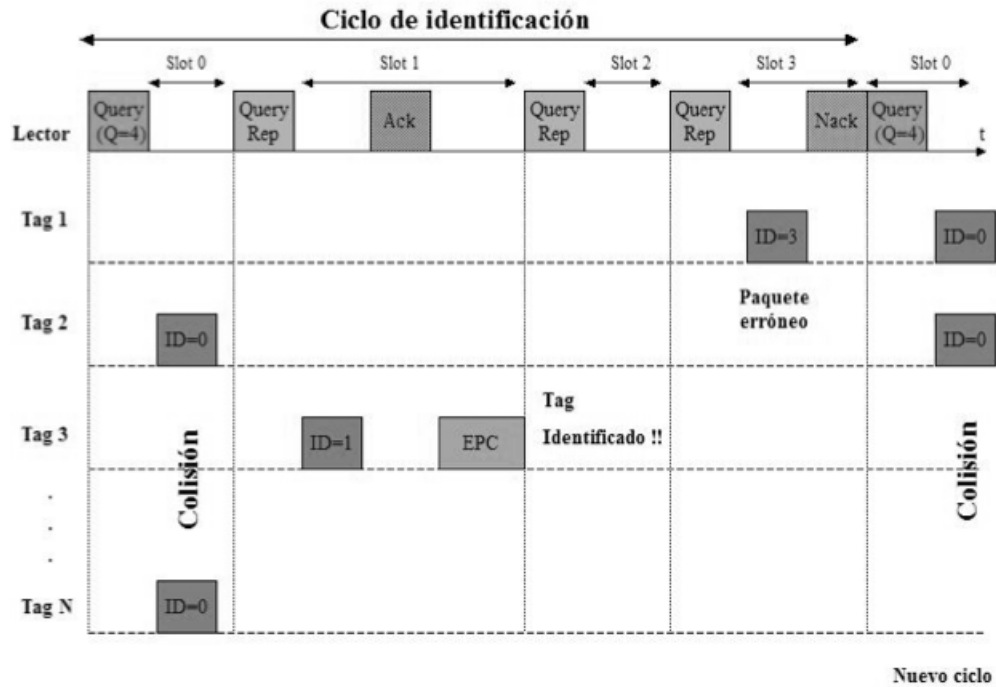


Figura 9. Mecanismo antilisión EPCglobal Class 1 Gen 2.

son numeradas desde 0 a $2^Q - 1$. La Figura 9 muestra el esquema para manejar colisiones en las etiquetas. Este es un valor aleatorio de 16 bits seleccionado por la etiqueta para esta sesión de comunicación (Ahson y Ilyas, 2008).

Los comandos anticollisión incluyen comandos de selección, en donde las etiquetas son elegidas para estar en una ronda en base a los comandos que dicta cuáles bits serán utilizados para la selección. Existen complejos criterios de selección que pueden ser desarrollados por la sucesión de comandos de selección. Existen cuatro bancos de memoria definidos: (a) Memoria reservada: Contiene las contraseñas de acceso, (b) Memoria EPC: Contiene un CRC-16, protocolo de control y el código EPC; identifica el objeto que llevará la etiqueta adherida y (c) Memoria TID: Contiene 8 bit ISO/IEC y suficiente información

para identificar los comandos personalizados y/o características opcionales que soporta una etiqueta.

Arduino

¿Qué es Arduino?

La placa electrónica Arduino es una fusión de tres elementos fundamentales: (a) hardware, (b) software y (c) la comunidad de personas que día a día contribuyen al crecimiento de esta plataforma (Oxer y Blemings, 2009).

El entorno Arduino ha sido diseñado para ser fácil de usar para las personas practicantes que no tienen experiencia de software o electrónica. Con Arduino, es posible crear objetos que pueden responder y/o controlar (a) la luz, (c) el sonido, (d) el tacto y (c) el movimiento. La placa electrónica Arduino se ha utilizado para crear una increíble variedad de cosas, incluyendo instrumentos musicales, robots, esculturas de luz, juegos interactivos, muebles, e incluso ropa interactiva (Margolis, 2011).

La placa electrónica Arduino se utiliza en muchos programas educativos en todo el mundo, especialmente por los diseñadores y artistas que quieren crear fácilmente prototipos pero no necesitan comprender a profundidad de los detalles técnicos detrás de sus creaciones. Debido a que está diseñado para ser utilizado por personas no técnicas, el software incluye una variedad de códigos de ejemplo que demuestran sus usos y aplicaciones. A pesar de que es fácil de usar, el hardware subyacente Arduino funciona en el mismo nivel de especialización que los ingenieros emplean para construir los dispositivos integrados. Las personas que ya están trabajando con los microcontroladores también son atraídos

por Arduino, debido a sus capacidades de desarrollo ágiles y su facilidad para la rápida implementación de las ideas (Margolis, 2011).

La placa electrónica Arduino es un pequeño microcontrolador con un conector USB que se puede conectar con una computadora, y con una serie de conectores que le permiten cablearse hasta electrónica externa, como (a) motores, (b) relevadores, (c) sensores de luz, (d) láser de diodos, (e) altavoces, y (f) micrófonos. Puede alimentarse a través de la conexión USB del ordenador o de una batería de 9V. Puede ser controlada desde el equipo, o programada por la computadora y luego desconectada para funcionar de manera independiente (Monk, 2010).

En el pasado, la electrónica requería trabajar con ingenieros especializados, construir pequeños componentes y tener amplios conocimientos. Pero los microcontroladores han evolucionado a dispositivos más baratos y fáciles de usar, siendo mucho más accesibles a aficionados, artistas e ingenieros de otras ramas (Martinez, 2012).

La placa electrónica Arduino proporciona la infraestructura para que todo el mundo sea capaz de construir sus propios objetos interactivos. Las placas se pueden adquirir o construir a partir de los diagramas, y un IDE (entorno de desarrollo integrado) liberado como código abierto, lo que hace a esta plataforma la ideal para personas con pocos conocimientos hasta para estudiantes y profesionales con un alto grado de especialización (Martinez, 2012).

Arduino software

Los programas de software, llamado sketches, son creados en un ordenador con el IDE para Arduino. El IDE permite escribir y editar código, convertirlo en instrucciones que el hardware de Arduino entiende. El IDE también transfiere las instrucciones a la placa Arduino, un proceso llamado uploading o carga de instrucciones (Margolis, 2011).

Arduino hardware

La placa Arduino es donde se ejecuta el código que se escribe. La placa Arduino sólo puede controlar y responder a la electricidad, por lo que los componentes específicos están conectados a él para que pueda interactuar con el mundo real. Estos componentes pueden ser sensores que convierten algún aspecto del mundo físico a la electrónica para que la placa pueda detectarlo, o actuadores que obtienen electricidad de la placa y la convierten en algo que interviene en el mundo (véase Figura 10). Algunos ejemplos de sensores son (a) interruptores, (b) acelerómetros y (c) sensores de distancia o de ultrasonido. Los actuadores son dispositivos tales como (a) las luces LED, (b) altavoces y (c) motores. Hay una gran

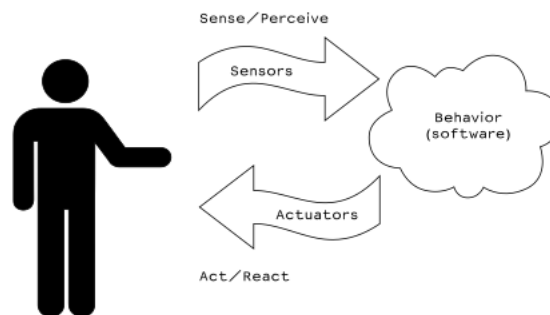


Figura 10. Interacción de sensores.

variedad de placas oficiales que pueden utilizarse con el software de Arduino y tableros de una gama amplia de Arduino compatible (Margolis, 2011). El dispositivo Arduino mas popular entre la comunidad de desarrolladores y estudiantes es Arduino UNO (véase Figura 11).

Arduino Ethernet

El Arduino Ethernet es una placa electrónica basada en el ATmega328. Cuenta con (a) 14 pines de entradas/salidas digitales, (b) 6 pines de entradas analógicas, (c) un oscilador de cristal de 16 MHz, (d) una conexión RJ45, (e) un conector de alimentación, (f) una cabecera ICSP (programador serial en circuito) y (g) un botón de reset (véase Figura 12). Los conectores 10, 11, 12 y 13 están reservados para el módulo Ethernet, esto reduce el número de pines disponibles a 9, con 4 disponibles como salidas PWM. La placa Arduino Ethernet se diferencia de otras placas ya que no tiene interconstruido un convertidor USB a Serial pero tiene una interfaz Ethernet. Cuenta con un lector de tarjetas microSD, que



Figura 11. Placa Arduino UNO.

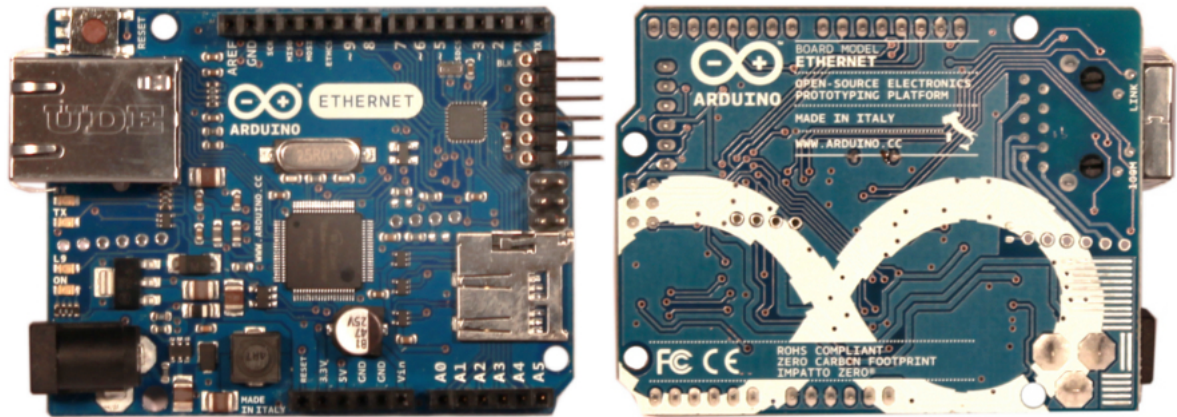


Figura 12. Placa Arduino Ethernet.

puede ser usado para almacenar archivos para servir a través de la red, se puede acceder a través de la Biblioteca SD (Comunidad, s.f.).

PHP

PHP es un lenguaje de alto nivel que se ejecuta en un servidor. Se ejecuta en el servidor donde se encuentran almacenadas las páginas web, al contrario de otros lenguajes que son ejecutados en el propio navegador. La principal ventaja, al ejecutarse el código en el servidor, es que todas las páginas tendrán la posibilidad de ser visitadas desde cualquier dispositivo independientemente del navegador web que se utilice. Por otro lado, la desventaja de que se ejecute el código en el navegador web, es que no todos los navegadores tienen la capacidad de soportar y entender todo el código, lo que se refleja en errores al mostrar el resultado al visitar una página web (Puertas, 2010).

PHP es un lenguaje de programación gratuito, esto significa que cualquier persona

puede utilizarlo sin costo alguno en comparación con otros lenguajes donde se necesitar comprar licencias de software (Puertas, 2010).

PHP es un lenguaje interpretado del lado del servidor y se caracteriza por (a) su potencia, (b) versatilidad, (c) robustez y (d) modularidad. Los programas escritos en PHP son incrustados directamente en HTML y ejecutados por el servidor web a través de un intérprete, antes de transferir al cliente lo que ha solicitado en formato de código HTML puro (véase Figura 13). PHP es un lenguaje multiplataforma: se ejecuta sobre la mayoría de los servidores web y tiene la capacidad para trabajar con más de 20 tipos de bases de datos. Al ser un lenguaje de programación inicialmente concebido para entornos Unix, es en estos donde mejor se pueden explotar todas sus capacidades. En comparación con otras tecnologías similares, PHP demuestra ser más veloz independientemente de la plataforma en la que se esté ejecutando. PHP ofrece un facilidad de conexión y es uno de los lenguajes

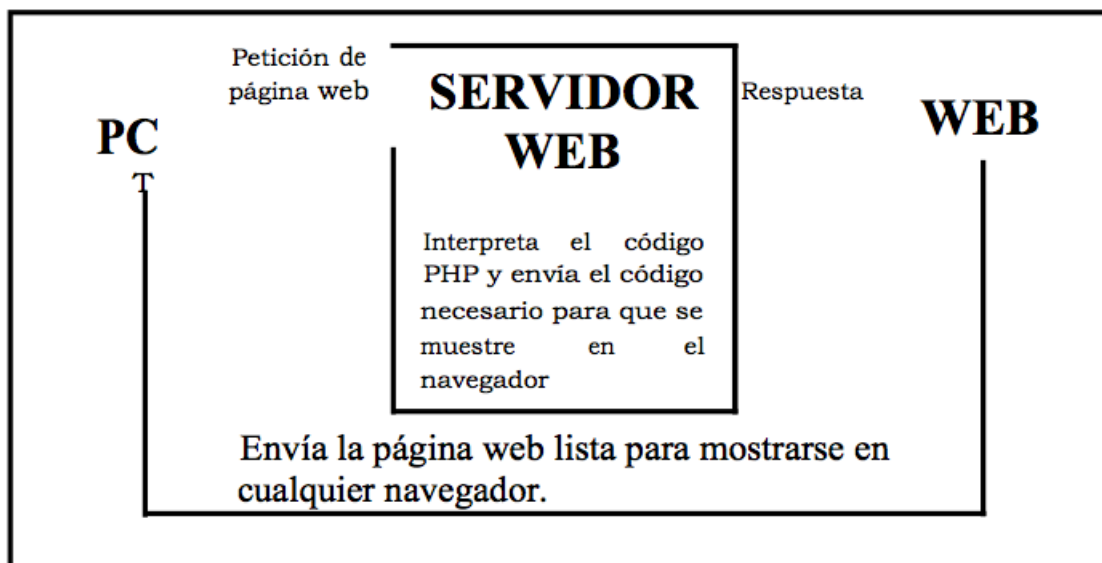


Figura 13. Proceso al visitar una página con PHP.

más utilizados para la generación de páginas dinámicas, no solo personales sino también de empresas y organizaciones (Cobo, 2005).

MySQL

MySQL es un sistema de administración de bases de datos relacionales (SGBDR) rápido, robusto y fácil de usar. Se adapta a la administración de datos en un entorno de red, especialmente en arquitecturas cliente/servidor. Proporciona muchas herramientas y es compatible con muchos lenguajes de programación. Este servidor de bases de datos es interrogable mediante SQL (Structured Query Language), el lenguaje estándar más popular para consultar bases de datos. SQL permite configurar los datos muy fácilmente (Thibaud, 2006).

MySQL es una base de datos que soporta el lenguaje SQL y la conexión de varios usuarios. La principal ventaja, al igual que PHP, reside en que es una base de datos gratuita (Puertas, 2010).

Red de datos TCP/IP

Una red de datos es un conjunto de redes digitales utilizadas para enviar datos entre computadoras, basa su funcionamiento en el modelo TCP/IP que se divide en capas para comunicaciones de internetworks y se conoce con el nombre de modelo de Internet. Define cuatro categorías de funciones que deben tener lugar para que las comunicaciones sean exitosas, como se muestra en la Figura 14.

El modelo TCP/IP describe la funcionalidad de los protocolos que forman la suite de protocolos TCP/IP. Estos protocolos se implementan en dispositivo emisor y receptor,

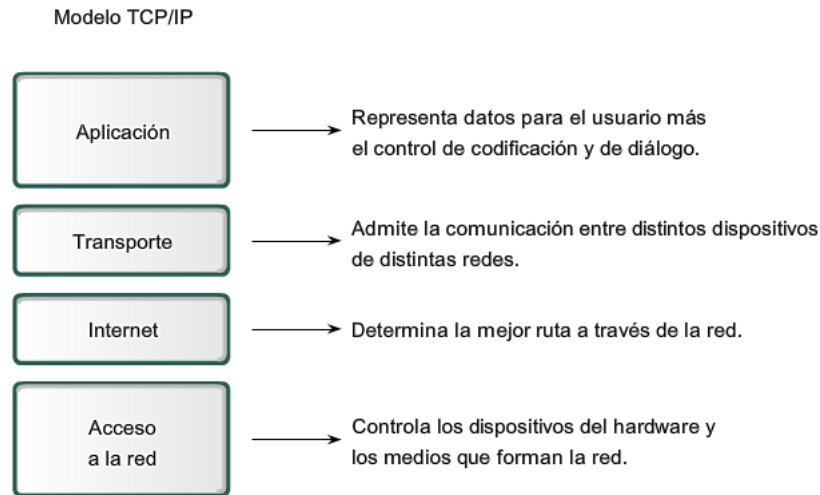


Figura 14. Modelo de capas TCP/IP.

interactúan para proporcionar la entrega de aplicaciones de extremo a extremo a través de una red. La información que se genera en las capas superiores, baja por las capas del protocolo y se transmite por los medios de red. Esto comúnmente se conoce como proceso de encapsulación como se muestra la Figura 15. La forma que adopta una sección de datos en cualquier capa se denomina Unidad de Datos del Protocolo (PDU). Durante la encapsulación, cada capa encapsula las PDU que recibe de la capa inferior de acuerdo con el protocolo que se utiliza. En cada etapa del proceso, una PDU tiene un nombre distinto para reflejar su nuevo aspecto (Odom, 2011).

La capa de acceso a la red se divide en dos subcapas, la capa física y la capa de conexión (véase Figura 16). Estas capas tienen un rol sumamente importante, pues gracias a ellas la información proveniente de las capas superiores hasta este punto software pasa a los componentes de hardware y la información es puesta en el medio de transmisión (Atelin y Dordoigne, 2007).

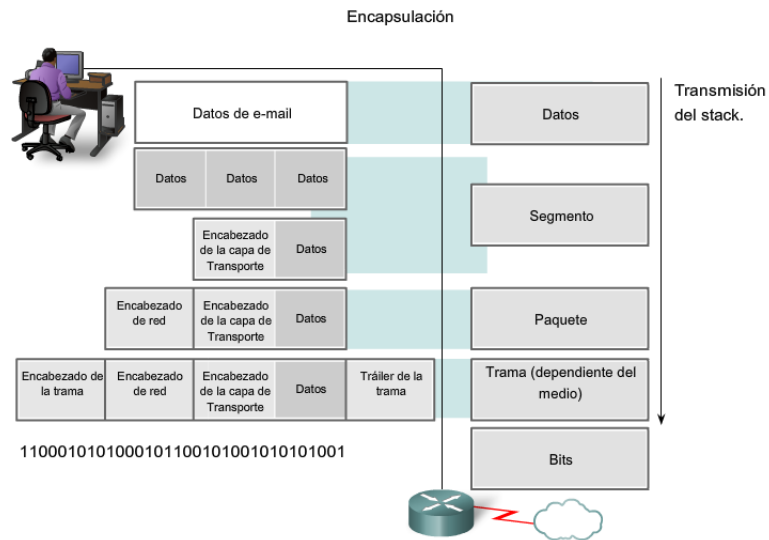


Figura 15. Encapsulación.



Figura 16. Subcapas de acceso a la red.

Capa física

Su papel es la transmisión de bit a bit a través del soporte de señales eléctricas, electromagnéticas o luminosas que cifran datos numéricos entre el emisor y el receptor (0 ó 1). En esta capa trabaja el protocolo Ethernet 802.3.

Capa de conexión

En esta capa los datos numéricos se transforman en señal, los bits de datos se organizan en tramas, se crea un encabezado en que se pueden identificar al emisor y al destinatario por su dirección física.

En esta capa se añade un código de redundancia cíclico (CRC – Cyclic Redundancy Code) que permite detectar algunos problemas de transmisión. Así el destinatario de una trama recalcula la suya y la compara con la que se transmitió. Si hay alguna diferencia, se rechaza. En esta capa trabaja el protocolo Ethernet 802.3 y 802.2.

Gracias al soporte de protocolos 802.2 y 802.3 es que la capa de acceso a la red cuenta con capacidad para diferentes medios de transmisión como cable de cobre, fibra óptica y Wireless.

Modelo de redes jerárquicas

Las redes jerárquicas implican la división de la red en capas independientes. Cada capa cumple funciones específicas que definen su rol dentro de la red general. La separación de las diferentes funciones existentes en una red hace que el diseño de la red se vuelva modular y esto facilita la escalabilidad y el rendimiento. El modelo de diseño jerárquico típico se separa en tres capas: (a) capa de acceso, (b) capa de distribución y (c) capa núcleo (véase Figura 17). En comparación con otros diseños de redes, una red jerárquica se administra y expande con más facilidad y los problemas se resuelven con mayor rapidez (Doherty, Anderson y Maggiora, 2008).

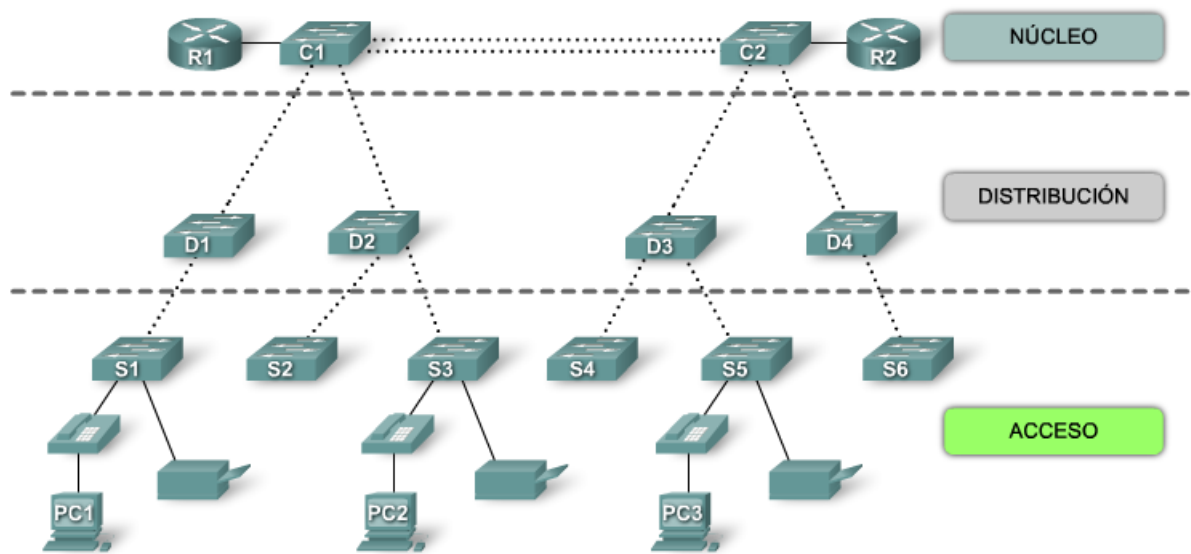


Figura 17. Modelo de redes jerárquicas.

Capa de acceso

La capa de acceso hace interfaz con dispositivos finales como las PC, impresoras y teléfonos IP, para proveer acceso al resto de la red. Esta capa de acceso puede incluir routers, switches, puentes, hubs y puntos de acceso inalámbricos. El propósito principal de la capa de acceso es aportar un medio de conexión de los dispositivos a la red y controlar qué dispositivos pueden comunicarse en la red (Doherty et al., 2008).

Capa de distribución

La capa de distribución agrega los datos recibidos de los switches de la capa de acceso antes de que se transmitan a la capa núcleo para el enrutamiento hacia su destino final. La capa de distribución controla el flujo de tráfico de la red con el uso de políticas y traza los dominios de broadcast al realizar el enrutamiento de las funciones entre las

LAN virtuales (VLAN) definidas en la capa de acceso. Las VLAN permiten al usuario segmentar el tráfico sobre un switch en subredes separadas. Por ejemplo, en una universidad el usuario podría separar el tráfico según se trate de profesores, estudiantes y huéspedes. Normalmente, los switches de la capa de distribución son dispositivos que presentan disponibilidad y redundancia altas para asegurar la fiabilidad (Doherty et al., 2008).

Capa núcleo

La capa núcleo en el diseño jerárquico es la capa backbone de alta velocidad de la internetwork. La capa núcleo es esencial para la interconectividad entre los dispositivos de la capa de distribución, por lo tanto, es importante que el núcleo sea sumamente disponible y redundante. El área del núcleo también puede conectarse a los recursos de Internet. El núcleo agrega el tráfico de todos los dispositivos de la capa de distribución, por lo tanto debe poder reenviar grandes cantidades de datos rápidamente (Doherty et al., 2008).

LAN

Una red de área local (LAN – Local Area Network, véase Figura 18) es un sistema de comunicaciones constituido por un hardware (cableado, terminales, servidores, etc.) y un software (acceso al medio, gestión de recursos de intercomunicación, etc) que se distribuyen por una extensión limitada (planta, edificio, grupo de edificios) en el que existen una serie de recursos compatibles, a los que tienen acceso los usuarios para compartir información (Solsona, Moya y Calero, 2006).

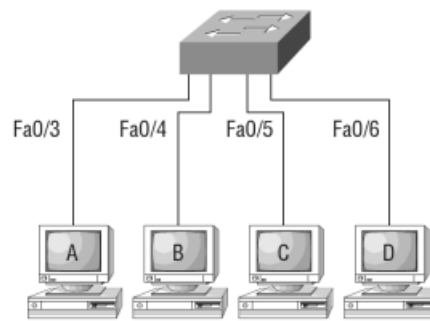


Figura 18. Topología lógica LAN.

VLAN

Una VLAN es una subred IP separada de manera lógica. Las VLAN permiten que redes de IP y subredes múltiples existan en la misma red conmutada. Una VLAN permite que un administrador de red cree grupos de dispositivos conectados a la red de manera lógica que actúan como si estuvieran en su propia red independiente, incluso si comparten una infraestructura común con otras VLAN (Lammle, 2008).

Mediante VLAN podemos disponer de una red conmutada que esté segmentada lógicamente según las aplicaciones, los protocolos y las funciones de los usuarios, sin que importe dónde están situados físicamente estos usuarios (véase Figura 19). Así pues, podríamos definir que cada puerto de un Switch forma parte de una VLAN distinta, si es necesario (Griera y Ordinas, 2009).

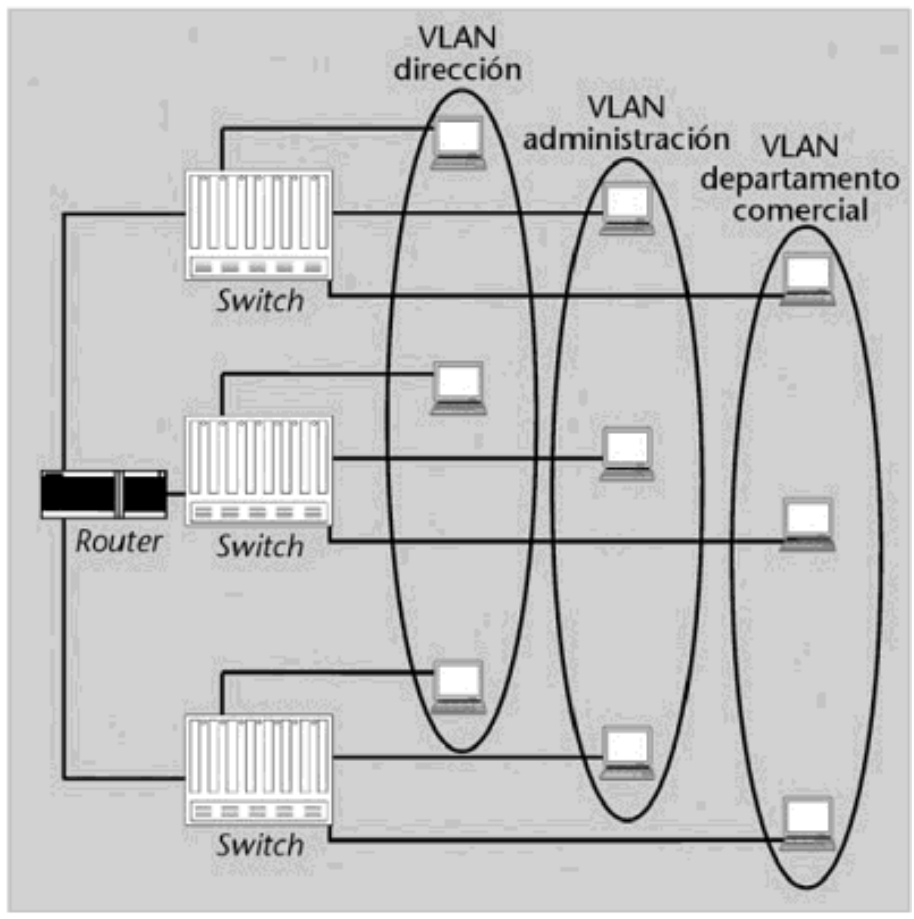


Figura 19. Topología logica VLAN.

CAPÍTULO III

METODOLOGÍA

Características del proyecto

En este capítulo se describe la metodología y los procesos utilizados para la creación del sistema de identificación del paciente. Se muestran las diferentes etapas, problemas y retos encontrados en el proceso así como las soluciones que se aplicaron.

La tecnología RFID ha sido utilizada por varios hospitales alrededor del mundo para la localización de pacientes, reducción en tiempo y monitorización de enfermeras. Se han encontrado múltiples ventajas en la implementación de esta tecnología.

El objetivo principal de este proyecto de investigación es la correcta identificación del paciente con ayuda de tecnología RFID y la consulta de esta información desde un portal web. En el inicio de este proyecto se contaba con un lector de RFID con USB, lo cual lo limitaba a estar conectado a una computadora todo el tiempo para su funcionamiento. Debido a la necesidad de contar con un lector más versátil, se compararon diversos lectores de diferentes capacidades y se optó tomar diferentes componentes para armar un lector RFID junto con un dispositivo Arduino dotado con una interfaz Ethernet para su comunicación en un infraestructura de red TCP/IP.

Para el desarrollo de este proyecto se utilizaron los siguientes elementos tecnológicos

(a) Lenguaje de programación Processing, (b) lenguaje de servidor PHP, (c) base de datos

relacional MySQL, (d) dispositivo Arduino Ethernet, (e) lector RFID, (f) diferentes tipos de etiquetas RFID y (g) infraestructura de red basada en TCP/IP.

Planificación

El objetivo de este proyecto es que a través de la tecnología RFID se logre una correcta identificación del paciente adquiriendo la identidad de una etiqueta, depositándola en un servidor, a través de un portal web accediento a esa información para identificar al paciente desde cualquier dispositivo, ya sea computadora o dispositivo móvil.

La primer limitante que se presentó, fue un lector de RFID con interfaz USB que brindaba un puerto serial virtual accesible a través de una CLI (método que permite dar instrucciones a algún dispositivo o programa por medio de una línea de texto simple), lo que limitaba al lector a estar conectado a una computadora para su funcionamiento. Se trabajó con un dispositivo Arduino UNO y se realizaron pruebas para abrir el puerto serial virtual desde una clase en PHP, pero de igual manera se necesitaba una computadora para el funcionamiento. Se buscó otro dispositivo que brindara mejores capacidades y se eligió el Arduino Ethernet, que cuenta con todas las bondades de la placa Arduino tradicional más una interfaz Ethernet. Se adquirió un pequeño lector RFID que trabaja a 125 kHz y se procedió a programar el Arduino Ethernet para que se comunicara con el lector RFID y adquiriera la información desde el lector. Se implentó un pequeño cliente web que enviaba la información a un servidor remoto y éste recibía la información desde el Arduino y la depositaba en la tabla correspondiente de la base de datos, que podía ser consultada mediante una página web. Cabe mencionar que el portal sólo admite al personal autorizado

para acceder a la información y tiene la capacidad de ajustarse a la pantalla de cualquier dispositivo móvil gracias a Responsive web Design (Diseño web adaptativo) que es una técnica de diseño y desarrollo web que consigue adaptar el sitio web al entorno del usuario mediante el uso de estructuras e imágenes fluidas, así como de media-queries en la hoja de estilo CSS.

Modelo de tres capas para el sistema RFID

Para la implementación del proyecto se propone un modelo de tres capas, como muestra la Figura 20 que describe el funcionamiento del sistema RFID para la (a) identificación de pacientes , (b) la convivencia e integración con los requerimientos de red mencionados anteriormente y (c) la interacción de los diferentes protocolos en diferentes capas. El modelo de tres capas para el sistema RFID estaría situado en la capa de acceso del modelo de red jerárquico, como se muestra en la Figura 21, este último modulariza la red de datos para poder trabajar en conjuntos, aislando los problemas que se puedan suscitar y así promoviendo el mantenimiento de la misma.



Figura 20. Modelo de tres capas del sistema RFID.

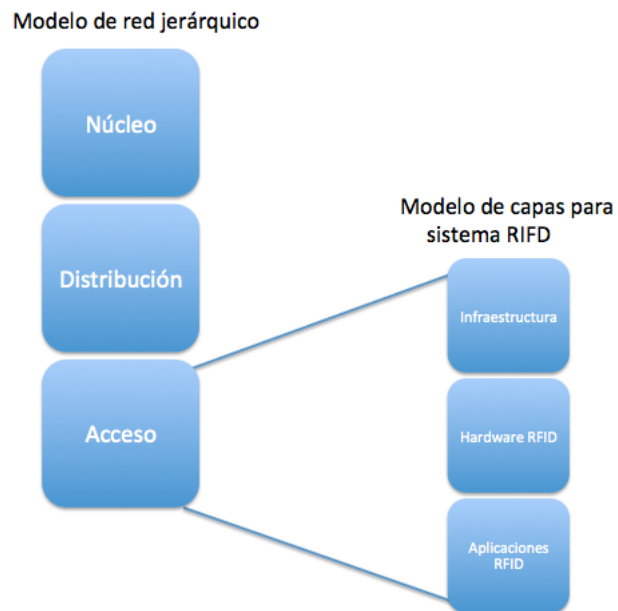


Figura 21. Modelo jerárquico y modelo de tres capas del sistema RFID.

Capa de infraestructura

Es la encargada de la comunicación en la red de datos y de poner la información en el medio de transmisión. Típicamente los medios de transmisión para redes son cable de cobre UTP, fibra óptica y tecnología wireless. Estos medios de transmisión utilizan el protocolo Ethernet 802.3 para poner la información en el medio de transmisión.

Las capas del modelo TCP/IP que trabajan dentro de la capa de red del modelo de tres capas son: (a) capa de Internet y (b) capa de acceso a la red.

La capa de Internet puede utilizar diferentes protocolos, para este proyecto el protocolo que se usará será el IPv4, que proporciona (a) direccionamiento de dispositivos, (b) encapsulación de la información proveniente de la capa de transporte, (c) enrutamiento para llegar a otras redes y (d) desencapsulamiento en el lado de receptor para extraer la información y pasarla a la capa de transporte.

La capa de acceso a la red se divide en subcapas: conexión y física. Para entender el funcionamiento de la subcapa de conexión, esta se divide en dos subcapas (a) control de enlace lógico y (b) control de acceso al medio. Donde la subcapa de control de enlace lógico (LLC) identifica el protocolo y entrama la información que llega de la capa de Internet. Los protocolos que trabajan en estas capas son 802.2 en LLC, 802.3 (Ethernet) en MAC y subcapa física, como se muestra en la Figura 22.

Las librerías SPI.h y Ethernet.h permiten que la placa electrónica Arduino, interactúe y se comunique exitosamente en las redes TCP/IP.

El dispositivo lector RFID cuenta con una interfaz Ethernet para comunicarse en

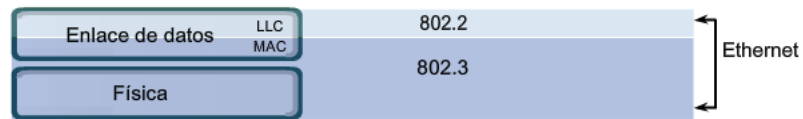


Figura 22. protocolos 802.2 y 802.3.

la red de datos. Gracias a esta interfaz es que el dispositivo lector puede depositar la información en un servidor, que de igual manera se accede desde la red de datos.

El lector RFID es un nodo en la red y cuenta con los parámetros adecuados para poder comunicarse: (a) dirección IP, (b) máscara de subred, (c) gateway y (d) dirección MAC como se muestra en la Figura 23. Además se especifica de forma manual el servidor en el cual depositará la información de las etiquetas RFID para identificar a un paciente.

Requerimientos de la red

Dada la necesidad de comunicar los dispositivos lectores con el servidor de base de datos, se eligió una red de datos basada en el modelo TCP/IP, pues gracias a su robustez, estandarización, interoperabilidad de protocolos y soporte para muchos medios de transmisión, es el modelo óptimo para el éxito de las comunicaciones hoy en día. El gran soporte de medios de transmisión y estándares como 802.3 y 802.11, que aporta la capa de acceso a la red, hace adecuado este modelo para la implementación del proyecto.

```
byte mac[] = { 0x90, 0xA2, 0xDA, 0x0D, 0x1F, 0x80 }; // MAC
byte ip[] = { 192, 168, 2, 20 }; // ip
byte subnet[] = { 255, 255, 255, 0 };
byte gateway[] = { 192, 168, 2, 1 };
byte server[] = { 10, 4, 254, 1 }; // ip server
```

Figura 23. Parámetros para comunicación en la red en el lector RFID.

El dispositivo lector RFID se ubicará dentro de la capa de acceso y contará con una VLAN exclusivamente para conectar los dispositivos lectores, de esta manera todo el tráfico en la red generado por los lectores estará aislado y se comunicarán únicamente con la VLAN donde se encuentre el servidor de base de datos, de igual manera esta VLAN contará con seguridad en los puertos del Switch. Como se propone en la Figura 24.

Capa de hardware RFID

Los lectores RFID con los que se trabajó son RFID ID-20 y operan a una frecuencia de 125 kHz, usan el protocolo EM4001 de 64-bits, cuentan con una interfaz serial (interfaz de comunicaciones donde la información es transmitida bit a bit enviando un solo bit a la vez) a 9600 bps y una interfaz RS232 y un rango de lectura de 20 cm.

En esta capa, el componente de hardware Arduino recibe la información desde el

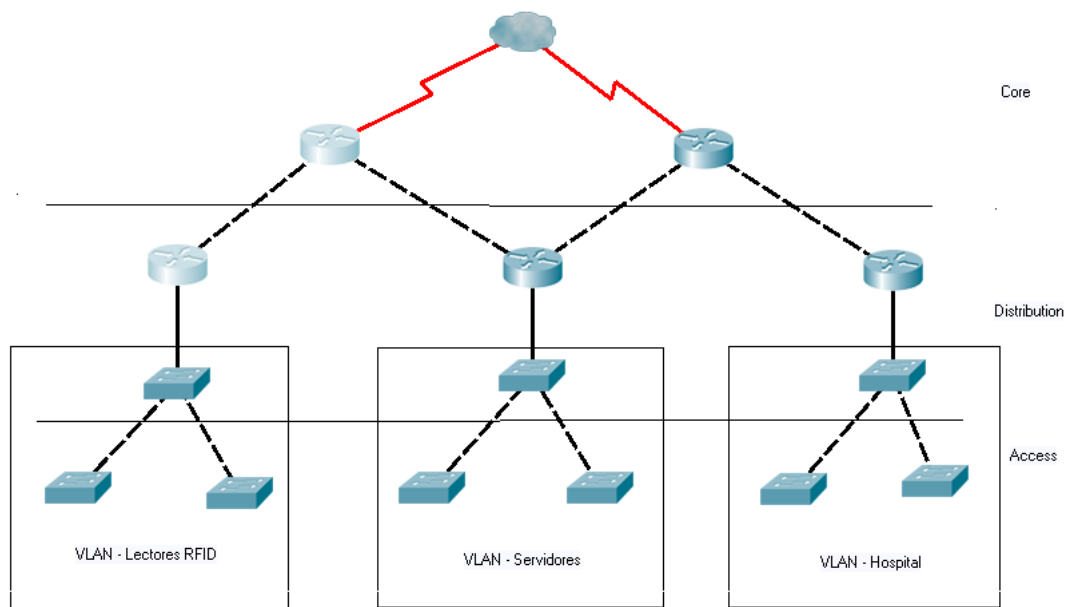


Figura 24. Red jerárquica con VLAN.

lector RFID a través del protocolo RS 232, es decir, por medio de este canal de comunicación llega el contenido de la etiqueta que identifica al paciente. La placa electrónica Arduino fue programada para recibir esta información y enviarla a un servidor. Para enviar esta información, se manda a la capa de red utilizando el protocolo 802.2, el cual marca un antes y un después en el software y prepara la información para ser enviada al protocolo 802.3 que es el encargado de entramar la información y ponerla en el medio de red, viajar por los medios de transmisión y llegar a servidor.

Arduino Ethernet + RFID

La programación del dispositivo Arduino se realiza en lenguaje de programación Processing. Un problema que surgió es que no todos los interrogadores RFID se comunican de la misma manera. Primero se realizaron pruebas de comunicación con el lector RFID empleando ejercicios sencillos. Al dominar la adquisición de datos desde el interrogador y mostrarla en una interfaz serial, se procedió a configurar la comunicación en la red TCP/IP, la cual se explica más adelante.

Adquiriendo los datos de las etiquetas RFID

Como se muestra en la Figura 25, la primera parte del sistema consiste en adquirir la identidad de una etiqueta mediante la interrogación del lector RFID a la etiqueta utilizando el protocolo EM4001. El interrogador RFID cuenta con una antena integrada con un rango de lectura de 20 cm, trabaja a una frecuencia de 125 kHz para la lectura de etiquetas y el envío de datos hacia la placa electrónica Arduino, esto se hace por medio de una interfaz serial que funciona a 9600 bps.



Figura 25. Interrogatorio RFID.

Las etiquetas, brazaletes, tarjetas y cápsulas de vidrio que se emplean en este proyecto trabajan a la misma frecuencia que el lector y se identifican por un número de serie único de 12 dígitos hexadecimales. La comunicación entre la placa electrónica Arduino y el interrogador se recibe por medio de la entrada digital D0 o RX, que funciona como pin para la recepción de comunicaciones seriales (véase Figura 26).

La Figura 27 muestra un fragmento de código de la programación en Processing

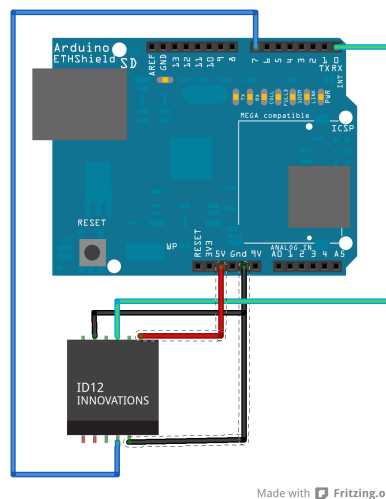


Figura 26. Diagrama de pines para la comunicación entre el interrogador RFID y la placa electrónica Arduino.

para la comunicación entre el interrogador y la placa Arduino. Al inicio se declara una variable tipo char con nombre tagString que contiene un arreglo de 12 caracteres, definido como tagString[11]. En este arreglo se guarda el código hexadecimal leído desde la etiqueta. En la siguiente línea se declara una variable index inicializada en 0 para saber en qué posición va el arreglo tagString, al final una variable tipo boolean con el nombre Reading establecido en false.

El ciclo while ejecuta la lectura del interrogador invocando una clase Serial.available, si existe comunicación serial disponible, en este caso la que llega desde el interrogador, entra dentro del ciclo while e indica el inicio de la lectura del tag bit por bit, estableciendo un 2 como inicio de la cadena de caracteres que va a leer, y al mismo tiempo cambiando el valor booleano de la variable reading a 2, indicando el inicio de la lectura del bit proveniente del interrogador. Después de leer el bit, la variable readByte se establece en 3 para indicar el final de la lectura y al mismo tiempo cambia el valor booleano de la variable reading a false, así mismo se valida y se guarda cada bit leído en la variable tagString y se igualan con readByte y al final se aumenta un valor en la posición de la variable index. Al final del ciclo while, cuando se completó la lectura de una etiqueta, se pasa el contenido de la variable tagString al método enviartag que desemboca en en el proceso de enviar el tag al servidor por medio de un cliente web. Después de haber sido enviado el valor del tag al servidor, la variable tagString detiene la ejecución del método clearTag, se inicializa y al final se ejecuta un método para reiniciar la actividad en el interrogador. La Figura 27 muestra el código para la lectura de etiquetas.

```

char tagString[11];
int index = 0;
boolean reading = false;

while(Serial.available()){
  int readByte = Serial.read(); // lee el siguiente byte disponible
  if(readByte == 2) reading = true; //inicio del tag
  if(readByte == 3) reading = false; //final del tag
  if(reading && readByte != 2 && readByte != 10 && readByte != 13){
    tagString[index] = readByte;
    index ++;
  }
}
enviartag(tagString);
clearTag(tagString); //limpia cualquier valor contenido en el char
resetReader(); //resetear el lector RFID

```

Figura 27. Programación para la comunicación del interrogador y la placa Arduino.

Configuración IP Arduino

La Figura 28 muestra el código de la configuración que debe tener todo dispositivo. En la primera línea se declara una variable tipo byte que contiene un arreglo llamado mac, donde se especifica cual será la dirección MAC de la NIC (tarjeta de interfaz de red); en la segunda línea se declara otra variable byte con un arreglo llamado ip, donde se configura la dirección de red del dispositivo; en la tercera línea se declara una variable byte con un arreglo llamado subnet que indica la máscara de subred a la que pertenece ese host. En la cuarta línea se declara una variable byte con un arreglo llamado gateway que indica la puerta de enlace predeterminada que le permitirá salir de la red en la que se encuentra; la siguiente variable es de tipo char, con un arreglo llamado serverName que contiene el nombre de dominio del servidor en el se guardará la información; en la quinta línea se encuentra una variable tipo EthernetClient que contiene la instancia del cliente configurado y al final una variable tipo int con una variable llamada RFIDResetPin con un valor establecido en 7, el cual se usa para reiniciar el interrogador después de una lectura.

```

byte mac[] = { 0x90, 0xA2, 0xDA, 0x0D, 0x1F, 0x80 }; // MAC
byte ip[] = { 192, 168, 2, 20 }; // ip
byte subnet[] = { 255,255,255,0 };
byte gateway[] = { 192, 168, 2, 1}; //201.134.41.98
//byte server[] = { 201, 134, 41, 98 }; // ip server
char serverName[] = "curriculas.cisco.um.edu.mx";
EthernetClient client;
int RFIDResetPin = 7;

```

Figura 28. Configuración de IP de la placa Arduino.

Cliente web incrustado

La Figura 29 muestra un fragmento de código del método `enviartag`, donde recibe una variable tipo `char` y la almacena en un arreglo con nombre `tag[]`, se verifica que la longitud del string no sea igual a cero, en caso de ser cierto se sale del ciclo; el siguiente paso es un ciclo `if` donde se usa la clase `client.connect` y se le pasan la variable `serverName` (nombre de dominio del servidor) y el puerto al cual se va a dirigir la petición, de cumplirse esta condición se procede a imprimir cierta información en el monitor serial de la placa Arduino, se usa la clase `Serial.println` para imprimir información en el monitor serial, aquí mismo se imprime la variable `tag` que se recibe al inicio del método y así podemos ver el contenido de caracteres hexadecimales que identifican a una etiqueta. El paso que sigue es utilizar la clase `client.print` para crear un URI (identificador uniforme de recursos, identifica inequívocamente un recurso o servicio, página, documento, dirección de correo electrónico) y depositar el contenido del tag en el archivo que recibirá la petición en el servidor web, al completarse este ciclo se imprimirá un mensaje en el monitor serial que indica que se ha hecho un depósito de información de manera correcta. De no cumplirse la condición inicial de que el servidor web este disponible en el monitor serial se imprimirá un mensaje de “falla en la conexión” y desconectará el cliente web. Las últimas tres líneas indican


```

void enviartag(char tag[]){
  if(strlen(tag) == 0) return;
  if (client.connect(serverName, 80)) { // puerto al que envia la peticion
    Serial.println("El Contenido del tag es ");
    Serial.println(tag);
    client.print("GET /hospital/arduino2.php?tag="); // Envia los datos utilizando GET
    client.print(tag);
    client.println(" HTTP/1.0");
    client.println("User-Agent: Arduino 1.0");
    client.println();
    Serial.println("Conexion exitosa");
    //tag [11]= 0;
  }
  else
  {
    Serial.println("Falla en la conexion");
  }
  if (client.connected()) {}
  else {
    Serial.println("Desconectado");
  }
  client.stop();
  client.flush();
  delay(1000);
}

```

Figura 29. Configuración de un cliente web para el envío de información al servidor web.

que se detiene el cliente con la sentencia `client.stop()` y se hace un reinicio del mismo con `client.flush()` y se espera 1000 milisegundos para la siguiente lectura con `delay(1000)`.

En la Figura 30 se muestra el monitor serial y el mensaje de monitoreo, así como el valor de la etiqueta que acaba de leer.

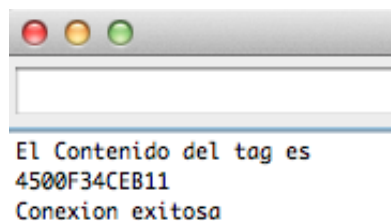


Figura 30. Monitor serial de la placa electrónica Arduino mostrando la identidad del tag.

Enviando la información al servidor web

La Figura 31 muestra una topología de red, donde el dispositivo Arduino utiliza la interfaz Ethernet para poder comunicarse en una red TCP/IP, obedeciendo los protocolos y estándares para mandar la información al servidor y depositar la información en la base de datos.

La Figura 32 muestra intercambio de tramas entre el cliente web y el servidor web para enviar la identidad de una etiqueta por medio de una petición tipo GET.

La Figura 33 muestra el contenido del intercambio de tráfico entre el cliente y el servidor. En letras rojas se puede apreciar la petición GET que hace el cliente web a la URI `/hospital/Arduino2.php?tag=` y se muestra el valor del tag `4500F34CEB11` (como se visualiza en el monitor se la placa electrónica Arduino) y demás datos como versión del protocolo y agente del cliente.

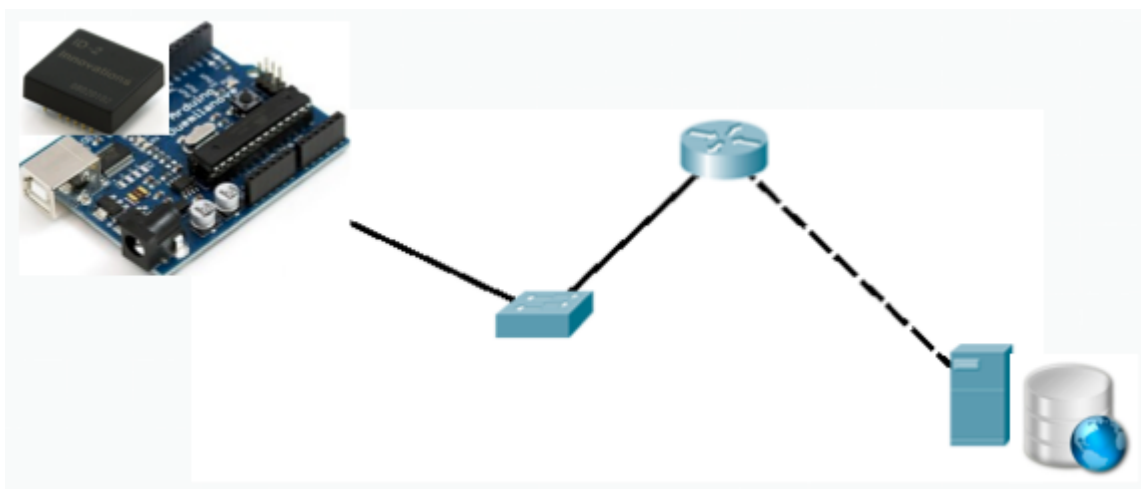


Figura 31. Red TCP/IP para la comunicación entre el cliente web en Arduino y el servidor web.

Number	Source	Destination	Protocol	Details
0	0.000000000	GheoSa Od:1f:80	Broadcast	60 who has 192.168.2.1? Tell 192.168.2.20
2	0.000035000	Private_ef:ef:c9	ARP	42 192.168.2.1 is at ac:de:48:ef:ef:c9
3	0.000213000	192.168.2.20	TCP	60 neod2 > http [SYN] Seq=0 Win=2048 Len=0 MSS=1460
4	0.001147000	10.4.254.1	TCP	58 http > neod2 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1460
5	0.001380000	192.168.2.20	TCP	60 neod2 > http [ACK] Seq=1 Ack=1 Win=2048 Len=0
6	0.003047000	192.168.2.20	TCP	60 [TCP segment of a reassembled PDU]
7	0.003412000	192.168.2.20	TCP	67 [TCP segment of a reassembled PDU]
8	0.003587000	192.168.2.20	TCP	63 [TCP segment of a reassembled PDU]
9	0.003805000	192.168.2.20	TCP	60 [TCP segment of a reassembled PDU]
10	0.003971000	192.168.2.20	TCP	60 [TCP segment of a reassembled PDU]
11	0.004163000	10.4.254.1	TCP	54 http > neod2 [ACK] Seq=1 Ack=32 Win=14600 Len=0
12	0.004436000	10.4.254.1	TCP	54 http > neod2 [ACK] Seq=1 Ack=45 Win=14600 Len=0
13	0.004474000	192.168.2.20	TCP	77 [TCP segment of a reassembled PDU]
14	0.004610000	192.168.2.20	TCP	60 [TCP segment of a reassembled PDU]
15	0.004770000	192.168.2.20	TCP	60 [TCP segment of a reassembled PDU]
16	0.005024000	192.168.2.20	TCP	60 [TCP segment of a reassembled PDU]
17	0.005107000	10.4.254.1	TCP	54 http > neod2 [ACK] Seq=1 Ack=54 Win=14600 Len=0
18	0.005235000	192.168.2.20	HTTP	60 GET /hospital/arduino2.php?tag=4500F34CEB11020 HTTP/1.0
19	0.005441000	192.168.2.20	TCP	60 neod2 > http [FIN, ACK] Seq=83 Ack=1 Win=2048 Len=0
20	0.005740000	10.4.254.1	TCP	54 http > neod2 [ACK] Seq=1 Ack=55 Win=14600 Len=0
21	0.006073000	10.4.254.1	TCP	54 http > neod2 [ACK] Seq=1 Ack=56 Win=14600 Len=0
22	0.006673000	10.4.254.1	TCP	54 http > neod2 [ACK] Seq=1 Ack=79 Win=14600 Len=0
23	0.007824000	10.4.254.1	TCP	54 http > neod2 [ACK] Seq=1 Ack=80 Win=14600 Len=0
24	0.007956000	10.4.254.1	TCP	54 http > neod2 [ACK] Seq=1 Ack=81 Win=14600 Len=0
25	0.008343000	10.4.254.1	TCP	54 http > neod2 [ACK] Seq=1 Ack=82 Win=14600 Len=0
26	0.008784000	10.4.254.1	TCP	54 http > neod2 [ACK] Seq=1 Ack=83 Win=14600 Len=0
27	0.046254000	10.4.254.1	TCP	54 http > neod2 [ACK] Seq=1 Ack=84 Win=14600 Len=0
28	0.080451000	10.4.254.1	HTTP	266 HTTP/1.1 200 OK (text/html)
29	0.082009000	10.4.254.1	TCP	54 http > neod2 [FIN, ACK] Seq=213 Ack=84 Win=14600 Len=0

Figura 32. Captura de tramas entre en cliente web y el servidor web.

```

Stream Content
GET /hospital/arduino2.php?tag=4500F34CEB11. HTTP/1.0
User-Agent: Arduino 1.0

HTTP/1.1 200 OK
Date: Tue, 07 May 2013 14:09:07 GMT
Server: Apache/2.2.22 (Ubuntu)
X-Powered-By: PHP/5.3.10-1ubuntu3.6
Vary: Accept-Encoding
Content-Length: 1
Connection: close
Content-Type: text/html

```

Figura 33. Contenido del stream de datos.

Recibiendo información desde Arduino

La información se recibe correctamente en la interfaz del servidor, se verifica que las tramas no estén corruptas o incompletas y que sean dirigidas a esa dirección MAC destino, en las capas superiores se determina a qué servicio va dirigido ese segmento, en este caso apunta al puerto 80 pues es un servidor web. La Figura 34 muestra el resultado del monitoreo del servidor web, se pueden ver las peticiones de entrada que se realizan.

Podemos apreciar (a) la petición entrante de cierta dirección IP, (b) la fecha y hora, (c) tipo de petición, en este caso un GET, la URI y el valor de tag en 4500F34CEB11.

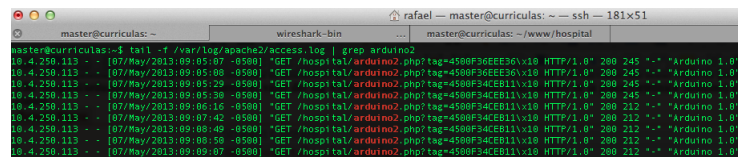
Archivos PHP y conexión con la base de datos

La Figura 35 muestra el contenido del archivo Arduino2.php, que es a donde llegan las peticiones del cliente web, este archivo recibe las variables que se le envían del cliente, la variable que recibe por medio de la petición GET es ?tag=4500F34CEB11, este mismo hace una conexión con la base de datos y especifica en que tabla y que campos se van a llenar con esta petición.

Hasta este momento la información ha sido enviada con éxito desde el cliente al servidor web, pasó por el log (monitor de servicio), llegó a la URI y al archivo arduino2.php correctamente, ahora el archivo Arduino2.php realiza una consulta para depositar la información en la base de datos, esto se visualiza en la Figura 36.

Capa de aplicaciones RFID

En esta capa residen todas las aplicaciones y servicios a los que accederán los profesionales de la salud y usuarios finales.



```
master@curriculas:~$ tail -f /var/log/apache2/access_log | grep arduino2
192.4.250.113 - - [07/May/2013:09:05:07 -0500] "GET /hospital/arduino2.php?tag=4500F34CEB11 HTTP/1.0" 200 245 "-" "Arduino 1.0"
192.4.250.113 - - [07/May/2013:09:05:08 -0500] "GET /hospital/arduino2.php?tag=4500F34CEB11 HTTP/1.0" 200 245 "-" "Arduino 1.0"
192.4.250.113 - - [07/May/2013:09:05:29 -0500] "GET /hospital/arduino2.php?tag=4500F34CEB11 HTTP/1.0" 200 245 "-" "Arduino 1.0"
192.4.250.113 - - [07/May/2013:09:05:30 -0500] "GET /hospital/arduino2.php?tag=4500F34CEB11 HTTP/1.0" 200 245 "-" "Arduino 1.0"
192.4.250.113 - - [07/May/2013:09:06:16 -0500] "GET /hospital/arduino2.php?tag=4500F34CEB11 HTTP/1.0" 200 212 "-" "Arduino 1.0"
192.4.250.113 - - [07/May/2013:09:07:42 -0500] "GET /hospital/arduino2.php?tag=4500F34CEB11 HTTP/1.0" 200 212 "-" "Arduino 1.0"
192.4.250.113 - - [07/May/2013:09:08:49 -0500] "GET /hospital/arduino2.php?tag=4500F34CEB11 HTTP/1.0" 200 212 "-" "Arduino 1.0"
192.4.250.113 - - [07/May/2013:09:08:58 -0500] "GET /hospital/arduino2.php?tag=4500F34CEB11 HTTP/1.0" 200 212 "-" "Arduino 1.0"
192.4.250.113 - - [07/May/2013:09:09:07 -0500] "GET /hospital/arduino2.php?tag=4500F34CEB11 HTTP/1.0" 200 212 "-" "Arduino 1.0"
```

Figura 34. Log de servicio en el software de servidor web.

```

<?
// Parametros de base de datos
$mysql_servidor = "localhost";
$mysql_base = "";
$mysql_usuario = "";
$mysql_clave = "";

//$id = htmlspecialchars($_GET["id"], ENT_QUOTES);
$tag = htmlspecialchars($_GET["tag"], ENT_QUOTES);
$tag = preg_replace('[^A-Za-z0-9]', "", $tag);
$hoy = date('Y-m-d H:i:s');
//echo "hoy es " . $hoy;
// Valida que esten presente todos los parametros
if (isset($tag)) {
    mysql_connect($mysql_servidor,$mysql_usuario,$mysql_clave) or die ("Imposible conectarse al servidor.");
    mysql_select_db($mysql_base) or die ("Imposible abrir Base de datos");
    $sql = "insert into reader5 (date, idtag) values ('$hoy','$tag')";
    mysql_query($sql);
    echo mysql_error();
}
else{
    echo "imposible conectar a la base de datos";
}
?>

```

Figura 35. Archivo Arduino2.php.

Consulta de la identidad del paciente por medio del portal web

El portal web cuenta con diferentes usuarios: (a) Doctores, (b) enfermeras y (c) recepcionistas. Cada usuario tiene acceso a ciertos privilegios específicos en el portal. La Figura 37 muestra el diagrama del portal web.

Dependiendo de los privilegios del usuario, al iniciar sesión en el portal web, se accederá a diferentes áreas del portal; en el caso de ser una recepcionista que inicia sesión, accederá al formulario de registro para los pacientes, de esta manera se estará ingresando la identidad del paciente (datos inequívocos) a la base de datos.

Los dispositivos lectores RFID están identificados por el lugar donde se encuentran,

```

~/www/hospital$ tail -f /var/log/mysql/mysql.log | grep reader4
722 Query    insert into reader4 (date, idtag) values ('2013-05-07 09:06:16','4500F34CEB11')

```

Figura 36. Consulta para depositar el valor del tag en la base de datos.

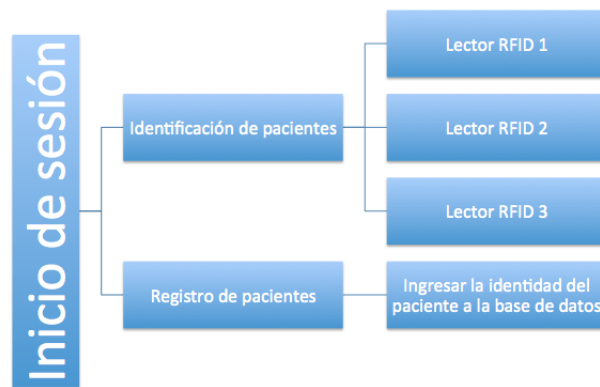


Figura 37. Diagrama de usuario y privilegios.

estarán situados estratégicamente en el área de camas, consultorios, etc. Cuando un doctor o una enfermera inicia sesión en el portal, tendrá la opción de elegir la ubicación del lector y consultar la identidad del paciente. Si una enfermera está en la cama 1, deberá iniciar sesión en el portal y elegir el lector RFID 1 para identificar al paciente de forma correcta. Si la enfermera desea consultar la identidad del paciente en la cama 2, deberá elegir el lector RFID 2 para consultar su identidad.

Como se explicó anteriormente, el dispositivo lector de RFID está situado en una ubicación específica e interrogará la identidad de la etiqueta que tiene el paciente que se encuentra dentro del rango de lectura y depositará la identidad de la etiqueta en la tabla que le corresponde, como se muestra en la Figura 38. De esta manera los doctores y enfermeras podrán consultar la identidad de cualquier paciente que se encuentre registrado en el sistema, identificado con una pulsera RFID y que esté dentro del rango de lectura del lector.

Cada lector RFID está asociado a una tabla en la base de datos, existe otra tabla

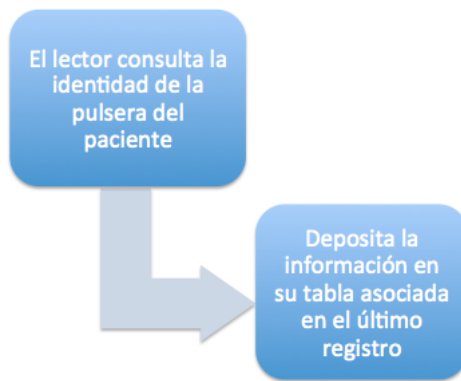


Figura 38. Proceso de depósito de la identidad en la tabla relacionada.

que contiene la identidad (datos inequívocos) de todos los pacientes registrados, el campo idtag se encuentra presente en ambas tablas. La Figura 39 muestra el procedimiento de inicio de sesión de un doctor o enfermera: (a) se selecciona la ubicación del lector para identificar al paciente, (b) se hace una consulta dependiendo el lector al que se este accediendo, (c) se verifica el último registro del campo idtag de la tabla asociada al lector y (d) se busca ese idtag en la tabla de los pacientes (tabla donde se encuentran registrados todos los pacientes). De esta manera se realiza la identificación y se contruye la página web a la que está accediendo el usuario.

La Figura 40 muestra la pantalla a la que acceden los doctores o enfermeras, los lectores RFID disponibles para indentificar a los pacientes en determinada ubicación. Cada circulo rojo representa un lector posicionado estratégicamente en un plano del hospital.

La Figura 41 muestra la selección del lector RFID que accede a la página donde se muestra la información inequívoca del paciente (a) fotografía, (b) nombre, (c) apellidos, (d) edad, (e) sexo, (f) estatura, etc.

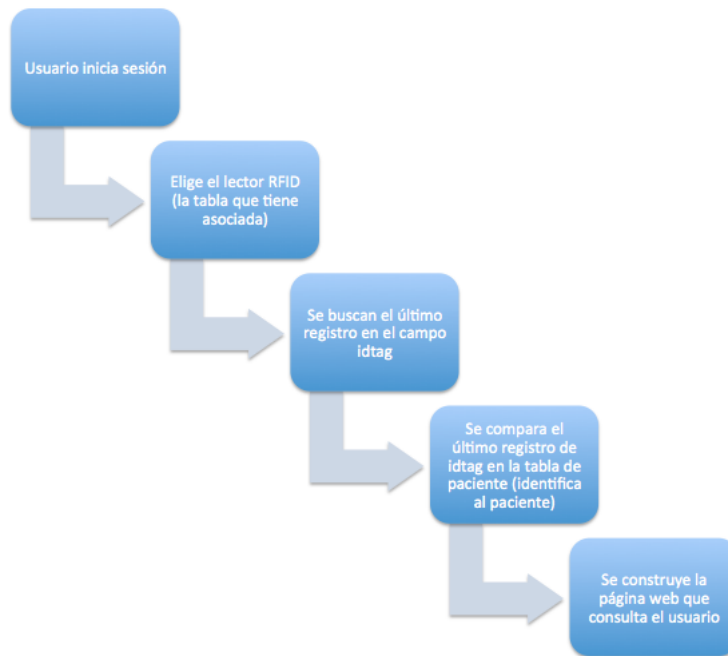
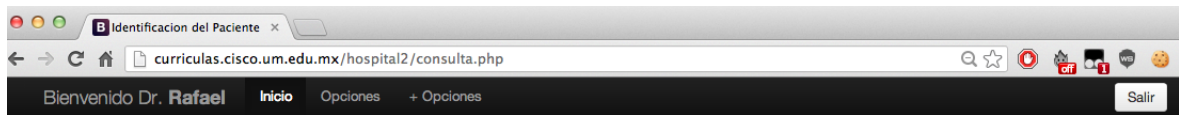


Figura 39. Consulta de la identidad iniciando sesión.



Seleccione un lector RFID

Para identificar a un paciente, pulse un círculo rojo en el mapa.



Figura 40. Selección de lector RFID.

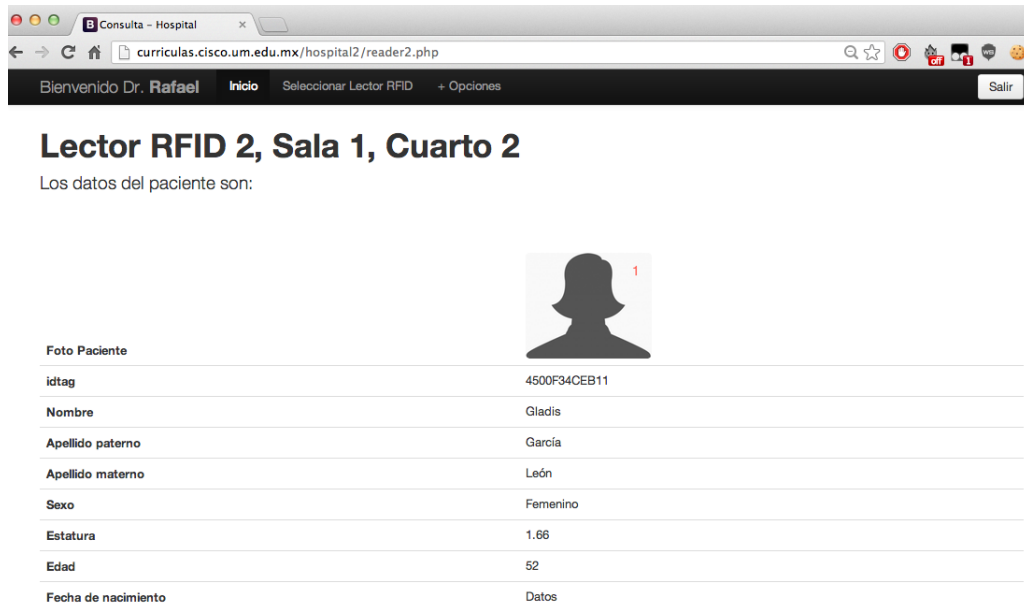


Figura 41. Portal web para la identificación del paciente mostrando la última lectura de interrogador RFID accediendo desde una computadora.

Para la creación del portal web se utilizó la herramienta Twitter Bootstrap, que es una colección de herramientas de software libre para la creación de sitios y aplicaciones web. Contiene plantillas de diseño basadas en HTML y CSS con (a) tipografías, (b) formularios, (c) botones, (d) gráficos, (e) barras de navegación y (f) demás componentes de interfaz, así como (g) extensiones opcionales de JavaScript. Gracias a esta herramienta, el diseño de las interfaces para el portal se realizó de manera óptima y podrá ser visualizada en la pantalla de cualquier dispositivo. La Figura 42 muestra el acceso al portal desde un dispositivo móvil. Se probó el acceso al portal web desde diferentes dispositivos como iPad, iPhone, móvil Android y PC.



Figura 42. Acceso al portal web desde un dispositivo móvil.

CAPÍTULO IV

CONCLUSIONES

Resumen

El capítulo I describe la importancia de la identificación de un paciente. Se analizaron protocolos a seguir para la identificación, dentro de la metodología se describe la forma como se registra al paciente y cuáles son los pasos para identificarlo. Se analizan las ventajas de implementación de aplicaciones hospitalaria, con tecnología RFID. Se proponen objetivos: (a) diseñar y desarrollar un lector de RFID con interfaz Ethernet para comunicarse en una red de datos TCP/IP, (b) desarrollo de un portal web para la consulta de la identidad de los pacientes, con la capacidad de poder acceder desde cualquier dispositivo móvil y exploradores web.

El capítulo II profundiza en la teoría de identificación de pacientes, las áreas que dependen de la correcta identificación, y se muestran los datos inequívocos para identificar a un paciente tales como son (a) Nombre, (b) apellidos, (c) sexo, (d) estatura, (e) edad y (f) fecha de nacimiento. Se aborda la teoría sobre RFID, componentes básicos de un sistema RFID, tipos de etiquetas, estándares y protocolos que rigen la comunicación de esta tecnología. Se describe el componente de hardware, placa electrónica Arduino así como sus bondades y capacidades. Por último se describen los componentes de software utilizados para la creación de páginas web con PHP y MySQL.

El capítulo III plantea la metodología para el desarrollo del proyecto, se analizan las desventajas e inconvenientes de identificar a un paciente hospitalizado de forma verbal, por medio de una entrevista o la identificación mediante una pulsera con datos impresos. Se propone el desarrollo de una herramienta para la correcta identificación del paciente usando tecnología RFID y un portal web que sea accesible desde la pantalla de cualquier dispositivo y de esta manera evitar las complicaciones que se pueden presentar al identificar incorrectamente a un paciente.

Se describe claramente todo el proceso de identificación de un paciente hospitalizado, desde (a) la adquisición de la identidad de una etiqueta, (b) el proceso de la información adquirida y enviada desde la placa electrónica Arduino hacia el servidor web, (c) la recepción de la información en el servidor web, (d) el depósito de la misma en el software de base de datos y finalmente (e) la consulta de esta información para identificarlo.

Gracias a la bondades de la tecnología RFID, es posible que no sea necesaria una identificación de forma verbal, una entrevista con el paciente hospitalizado o la identificación del paciente verificando los datos impresos en una pulsera.

El capítulo IV presenta las conclusiones y recomendaciones sugeridas de la realización del proyecto.

Conclusiones

El documento presenta un proyecto interdisciplinario donde software y hardware interactúan para cumplir el objetivo de identificar correctamente a pacientes hospitalizados. El resultado ha sido el desarrollo de un modelo para el sistema RFID, donde cada

capa describe las funciones, protocolos y estándares utilizados; además se comprende desde la adquisición de la identidad de una etiqueta para identificar a una persona y consultar desde cualquier dispositivo móvil o estación de trabajo. Contar con un modelo de capas brinda modularidad al sistema, ya que se pueden quitar, añadir o reemplazar dispositivos y el funcionamiento no se verá afectado.

Un componente fundamental para el desarrollo del proyecto es la infraestructura de red que se describe en el modelo jerárquico, ya que brinda comunicación continua y redundante, la cual es sumamente importante para el envío y recepción de información desde el dispositivo final como el lector RIFD hacia el servidor web y base de datos.

Al finalizar el desarrollo de los dispositivos que conforman el proyecto, la funcionalidad descrita en los objetivos fue alcanzada.

La frecuencia a la que trabaja el lector RFID es de 125 kHz, lo cual permite rangos de lectura cortos y obliga a tener un lector por cada cama, consultorio o cualquier otro lugar donde se necesite identificar a un paciente.

El protocolo que se utiliza para la lectura de etiquetas RFID es EM4001, el cual permite la lectura de una etiqueta a la vez, en parte esto es benéfico pues así se evita leer dos etiquetas e identificar erróneamente a un paciente.

Resultados

Como resultado de este proyecto se tiene el marco general para la creación de dispositivos funcionales para la lectura de etiquetas y envío de información, como se menciono anteriormente, puede ser empleado en otros proyectos de identificación . Se desarrolló,

además un portal web que despliega información inequívoca del paciente, además de tener la capacidad de manejar usuarios y privilegios. Se obtuvo un amplio conocimiento y dominio de diferentes plataformas tanto de software como de hardware y la capacidad de hacerlos interactuar para suplir una necesidad.

CAPÍTULO V

TRABAJOS FUTUROS

Como trabajos futuros se proponen los siguientes:

1. Desarrollar una antena y circuitería necesaria para aumentar el rango de alcance de los lectores RFID.
2. Implementar un protocolo anti-colisión para la lectura de etiquetas diferentes de forma simultanea, el cual se pueda adecuar a aplicaciones sanitarias como logística de bienes, inventarios en almacenes y farmacias, entre otras.
3. Implementar toda la circuitería en un solo PCB para el fácil uso, mantenimiento y reducción de costos por unidad.
4. Implementación de un sistema de localización en tiempo real (RTLS) para monitorear la ubicación de pacientes, personal y equipo médico.
5. Añadir al sistema la capacidad de manejar el expediente clínico electrónico del paciente. De esta manera no solo se mostrarían los datos inequívocos del paciente, si no su historial completo.

APÉNDICE A

LECTOR RFID & CLIENTE WEB


```

#include <SPI.h>
#include <Ethernet.h>
byte mac[] = { 0x90, 0xA2, 0xDA, 0x0D, 0x1F, 0x80 }; // MAC
byte ip[] = { 192, 168, 2, 20 }; // ip
byte subnet[] = { 255,255,255,0 };
byte gateway[] = { 192, 168, 2, 1};
byte server[] = { 10, 4, 254, 1 }; // ip server
//char serverName[] = "curriculas.cisco.um.edu.mx";
EthernetClient client;
int RFIDResetPin = 7;
void setup(){ Serial.begin(9600);
Serial.println("Comunicacion_Serial_Lista");
Serial.println("");
pinMode(RFIDResetPin, OUTPUT);
digitalWrite(RFIDResetPin, HIGH);
Ethernet.begin(mac, ip, subnet, gateway);
Serial.print("Direccion_IP_del_Lector_RFID_");
Serial.println(Ethernet.localIP());
//Serial.println(mac[]); delay(1000); }
void loop(){ char tagString[11];
int index = 0;
boolean reading = false;
while(Serial.available()){
int readByte = Serial.read(); // lee el siguiente byte disponible
if(readByte == 2) reading = true; //inicio del tag
if(readByte == 3) reading = false; //final del tag
if(reading && readByte != 2 && readByte != 10 && readByte
!= 13){ tagString[index] = readByte; index ++; } }
enviartag(tagString);
clearTag(tagString); //limpia cualquier valor conteniso en el char
resetReader(); //resetear el lector RFID
}
void enviartag(char tag[]){ if(strlen(tag) == 0) return;
if (client.connect(server, 80)) { // puerto al que envira la peticion
Serial.println("El_Contenido_del_tag_es_");
Serial.println(tag);
client.print("GET_/hospital/Arduino.php?tag="); // Envia los datos utilizando GET
client.print(tag);
client.println("_HTTP/1.0");
client.println("User-Agent:_Arduino_1.0");
client.println(); Serial.println("Conexion_exitosa"); //
tag [11]= 0; }
else {
Serial.println("Falla_en_la_conexion");
}
if (client.connected()) {}
else { Serial.println("Desconectado"); }
client.stop();
client.flush();
delay(1000); }
void resetReader(){

```

```
digitalWrite(RFIDResetPin, LOW);
digitalWrite(RFIDResetPin, HIGH);
//Serial.println("Reset and wait");
delay(150);
}
void clearTag(char one[]){
tag has been read otherwise
for(int i = 0; i < strlen(one); i++){
one[i] = 0;
}
}
```

APÉNDICE B
ARDUINO.PHP

```

<? // Parametros de base de datos
$mysql_servidor = "localhost";
$mysql_base = "";
$mysql_usuario = "";
$mysql_clave = "";

$tag = htmlspecialchars($_GET["tag"], ENT_QUOTES);
$tag = preg_replace('([^\A-Za-z0-9])', "", $tag);
$hoy = date('Y-m-d_H:i:s');

// Valida que esten presente todos los parametros
if (isset($tag)) {
    mysql_connect($mysql_servidor, $mysql_usuario, $mysql_clave) or die ("
        Imposible_conectarse_al_servidor.");
    mysql_select_db($mysql_base) or die ("Imposible_abrir_Base_de_datos"
    );
    $sql = "insert_into_reader5_(date, _idtag)_values_('$hoy', '$tag')";
    mysql_query($sql);
    echo mysql_error();
} else {
    echo "imposible_conectar_a_la_base_de_datos";
}
?>

```

LISTA DE REFERENCIAS

- Ahson, S. e Ilyas, M. (2008). *RFID handbook: Applications, technology, security, and privacy*. Recuperado de <http://books.google.com.mx/books?id=q4aCyZnq0cwC>
- Atelin, P. y Dordoigne, J. (2007). *TCP/IP y protocolos de internet*. Recuperado de <http://books.google.com.mx/books?id=XCSNAwbuvKkC>
- Bacheldor, B. (2009). RFID prevents valued hospital assets from getting trashed. *RFID Journal*. Recuperado de <http://www.rfidjournal.com/article/view/4808>
- Britton, J. (2007). An investigation into the feasibility of locating portable medical devices using radio frequency identification devices and technology. *Journal of Medical Engineering & Technology*, 31, 450-458. doi:10.1080/03091900701292141
- Cobo, Á. (2005). PHP y MySQL: Tecnología para el desarrollo de aplicaciones web. Recuperado de <http://books.google.com.mx/books?id=zMK3GOMOpQ4C>
- Comunidad. (s.f.). *Arduino ethernet*. Recuperado de <http://arduino.cc/en/Main/ArduinoBoardEthernet>
- Corrales, A., Rivas, R. y Salichs, M. (2007). *Sistema de identificación de objetos mediante RFID para un robot personal*. Recuperado de <http://rua.ua.es/dspace/bitstream/10045/10889/1/ArticuloProceedings.pdf>
- de la Torre, A., Martín, C. y Arribas, C. (1996). *Manual de cuidados intensivos para enfermería*. Recuperado de <http://books.google.com.mx/books?id=dpIsfX430IsC>
- Doherty, J., Anderson, N. y Maggiora, P. (2008). *Cisco networking simplified*. Recuperado de <http://books.google.com.mx/books?id=ho2YuAAACAAJ>
- Fanberg, H. (2004). *The RFID revolution*. Recuperado de <http://goo.gl/FAuxk>
- FDA. (2005, Marzo). *Radiofrequency identification technology: Protecting the drug supply*. Recuperado de <http://www.fda.gov/Drugs/DrugSafety/ucm169918.htm>
- Fuller, J. (2007). *Instrumentación quirúrgica: teoría, técnicas y procedimientos*. Recuperado de <http://books.google.com.mx/books?id=yBwepEJsQZQC>

- Griera, J. y Ordinas, J. (2009). *Estructura de redes de computadores/ structures of computer networks*. Recuperado de <http://books.google.com.mx/books?id=QAxAJEBgUWYC>
- Gutierrez, P. A. (2008). *Seguridad en la identificación de pacientes*. Recuperado de <http://www.saludinnova.com/practices/view/105/>
- Hansen, W. y Gillert, F. (2008). *RFID for the optimization of business processes*. Recuperado de <http://books.google.com.mx/books?id=66IdDCoiu04C>
- Hendrich, A., Chow, M., Skierczynski, B. A. y Lu, Z. (2008). *A 36-hospital time and motion study: How do medical-surgical nurses spend their time?* Recuperado de <http://goo.gl/wK5FE>
- Kumar, S., Livermont, G. y McKewan, G. (2009). Stage implementation of RFID in hospitals. *Technology and Health Care*, 18, 31-36. doi:10.3233/THC-2010-0570
- Lammle, T. (2008). *Ccna: Cisco certified network associate: Fast pass*. Recuperado de <http://books.google.com.mx/books?id=PARCyPSSkF8C>
- Margolis, M. (2011). *Arduino cookbook*. Recuperado de <http://books.google.com.mx/books?id=raHyKejOBF4C>
- Martinez, J. A. H. (2012). *Sistemas y soluciones para el regadio* (Tesis de Maestría no publicada). Universidad Politécnica de Valencia, España.
- Monk, S. (2010). *30 Arduino projects for the evil genius*. Recuperado de <http://books.google.com.mx/books?id=pki5o3lU75gC>
- Montanez, H. G. L. (2008). Sistema de identificación del paciente en el HONADOMANI San Bartolomé. Recuperado de http://www.sanbartolome.gob.pe/calidad/cim/ide_pac.pdf
- Nebrada, C. (1998). *Glosario de términos para la administración y gestión de los servicios sanitarios*. Recuperado de http://books.google.com.mx/books?id=_OIMd9UbOBsC
- Odom, W. (2011). *Ccent / ccna icnd1 640-822 official cert guide*. Recuperado de http://books.google.com.mx/books?id=dB8xxdBqG_EC
- Oxer, J. y Blemings, H. (2009). *Practical arduino: Cool projects for open source hardware*. Recuperado de <http://books.google.com.mx/books?id=HsTxON1L6D4C>

- Portillo, J., Bermejo, A. y Bernardos, A. (2008). *Informe de de vigilancia tecnologica madri+d "tecnología de identificacion por radio frecuencia (RFID): aplicaciones en el ambito de la salud"*. Recuperado de <http://goo.gl/GyG6x>
- Puertas, J. (2010). *Creacion de un portal con php y mysql*. Recuperado de <http://books.google.com.mx/books?id=T4XopwAACAAJ>
- Ramos, B. y Aznar, D. (1994). *Administración de medicamentos: Teoría y práctica*. Recuperado de <http://books.google.com.mx/books?id=NJhzDTwqchkC>
- Solsona, A., Moya, J. y Calero, J. (2006). *Redes de área local: administración de sistemas informáticos*. Recuperado de <http://books.google.com.mx/books?id=V2xogIe99B8C>
- Tamayo, C. (2004). *Admisión y gestión de pacientes en atención especializada*. Recuperado de http://www.seis.es/documentos/informes/secciones/adjunto1/CAPITULO3_1.pdf
- Thibaud, C. (2006). *Mysql 5: Instalacion, implementacion, administracion, programacion*. Recuperado de <http://books.google.com.mx/books?id=wY0bHPmW-NUC>
- Thornton, F., Haines, B., Das, A., Bhargava, H., Campbell, A. y Kleinschmidt, J. (2006). *RFID security*. Recuperado de <http://books.google.com.mx/books?id=UQYjJaSk2EcC>