



Centro de Investigación y de Estudios Avanzados  
del Instituto Politécnico Nacional

Departamento de Ingeniería Eléctrica  
Sección de Computación

## **Sistema de Control de Acceso con RFID**

Tesis que presenta

**Jorge Alberto Alvarado Sánchez**

Para obtener el Grado de

Maestro en Ciencias

En la Especialidad de

Ingeniería Eléctrica

Opción Computación

Director: Dr. Xiaou Li Zhang

Co-Director: Dr. Aldo Gustavo Orozco Lugo

México, D.F., Enero 2008







Los abajo firmantes, integrantes del jurado para el examen de grado que sustentará el Sr. Jorge Alberto Alvarado Sánchez, declaramos que hemos revisado la tesis titulada:

**SISTEMA DE CONTROL DE ACCESO CON RFID**

Y consideramos que cumple con los requisitos para obtener el Grado de Maestría en Ciencias en la especialidad de Ingeniería Eléctrica opción Computación.

Atentamente,

Dr. Xiaou Li Zhang

Dr. Aldo Gustavo Orozco Lugo

Dr. Adriano de Luca Pennacchia

Dr. José Oscar Olmedo Aguirre



# Agradecimientos

Todo mi agradecimiento a mi familia, en especial a mis padres y hermana por todo su apoyo, motivación y cariño.

Agradezco a todos mis amigos del CINVESTAV por haber compartido conmigo esta aventura, por su solidaridad y apoyo, en especial a Francisco y a Gil.

Un agradecimiento muy especial a mis asesores Dra. Xiaou Li y Dr. Aldo Orozco, por su apoyo, paciencia y confianza.

Agradezco a todos mis maestros del Departamento de Computación por compartir sus conocimientos y experiencia.

Agradezco al CINVESTAV, por ser una gran institución, de la cual me llevo un gran aprendizaje y una muy buena experiencia.

Agradezco al CONACYT por los recursos aportados, sin los cuales hubiera sido muy difícil este trayecto.





## Resumen

La tecnología de RFID es un sistema de autoidentificación inalámbrico, el cual consiste de etiquetas que almacenan información y lectores que pueden leer a estas etiquetas a distancia. La tecnología RFID está siendo adoptada cada vez por más industrias debido a que su costo es cada vez menor y sus capacidades son mayores. Esto permite genera grandes beneficios como incrementos en la productividad y administración principalmente en los sectores de cadenas de suministro, transporte, seguridad y control de inventarios.

En esta tesis, se hace un estudio de la tecnología de RFID, se exploran sus capacidades, se plantean sus ventajas sobre otras tecnologías de autoidentificación y los elementos que intervienen en un proyecto de este tipo. En esta tesis se propone un caso de estudio orientado hacia el control de acceso con tecnología RFID, el cual consiste de un bus de RS-485 que tiene conectados lectores de RFID y tarjetas que permiten controlar dispositivos actuadores. Una PC se encuentra conectada a este bus y tiene aplicaciones de software para realizar la configuración del sistema, coordinar la interacción con los elementos en la red, hacer la administración de usuarios y generar reportes.

## Abstract

RFID technology is a wireless identification technology that consists of tags that can store data and readers that can get this data . RFID technology is being adopted by more industries due to its capabilities and its low cost. It brings many benefits, like productivity increases and ease of management mainly in the supply chain, transportation, security and inventory management sectors.

In this tesis are proposed the advantages of RFID over other autoidentification technologies, its capabilities and all the elements that are related with RFID systems, base in a Access Control study case. This Access Control System has a RS-485 bus that has RFID readers and cards that are able to control actuators. A PC is plug to this bus and runs software applications related with the Access Control such as system configuration, devices management ,users management and reports generation.

# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Antecedentes	2
1.2. Descripción del Problema	2
1.3. Trabajo Relacionado	3
1.4. Motivación	4
1.5. Objetivos	5
1.6. Organización	6
<b>2. Tecnologías de Autoidentificación y RFID</b>	<b>7</b>
2.1. Tecnologías de Autoidentificación	7
2.1.1. Comparación de Tecnologías de Autoidentificación	7
2.1.2. Comparación entre Tecnologías de RFID y Código de Barras	12
2.1.3. Ventajas de la Identificación por Radiofrecuencia	12
2.2. Identificación por Radiofrecuencia	14
2.2.1. Tecnología RFID	14
2.2.2. Lectores de RFID	16
2.2.3. Frecuencias	18
2.2.4. Estándares	19
2.2.5. Conectividad	21
2.2.6. Middleware	22
2.2.7. Seguridad	24
2.2.8. Tendencias	25
<b>3. Descripción del Sistema</b>	<b>27</b>
3.1. Infraestructura	27
3.2. Componentes del Sistema	29
3.3. Arquitectura del Sistema desarrollado	31
<b>4. Diseño de Hardware</b>	<b>35</b>
4.1. Dispositivos RFID	36
4.1.1. Transponder (Tarjetas RFID)	36
4.1.2. Lector de RFID	36
4.2. Tarjeta de Conversión RS232 - RS485	43

4.3. Tarjeta Controladora . . . . .	45
4.3.1 Módulo de Comunicaciones . . . . .	46
4.3.2 Microcontrolador . . . . .	46
4.3.3 Módulo de Potencia . . . . .	48
4.4. Tarjetas impresas . . . . .	50
<b>5. Diseño de Software</b>	<b>53</b>
5.1. Esquema de Base de Datos . . . . .	55
5.2. Módulo de Configuración del Sistema desarrollado . . . . .	57
5.3. Módulo de Control de Acceso . . . . .	62
5.4. Módulo de control de acceso para Visitantes . . . . .	68
5.5. Módulo de Administración de Usuarios . . . . .	70
5.6. Módulo Generador de Reportes en PDF y en EXCEL . . . . .	74
5.7. Aplicación WEB . . . . .	77
<b>6. Conclusiones y Trabajo Futuro</b>	<b>83</b>
6.1. Conclusiones . . . . .	83
6.2. Trabajo Futuro . . . . .	85
<b>A. Arquitectura SOA</b>	<b>87</b>
A.1. Arquitectura de integración del Sistema Desarrollado . . . . .	87
A.2. Conceptos SOA . . . . .	88
A.3. Descripción de la integración de la aplicación de control de acceso con Otros sistemas . . . . .	89
A.4. Ejecución del Proceso BPEL . . . . .	98
<b>Bibliografía</b>	<b>103</b>

# Índice de tablas

2.1. Tecnologías de Control de Acceso . . . . .	11
4.1. Configuración del Lector[4] . . . . .	38
4.2. Comandos del Lector[4] . . . . .	40

# Índice de figuras

2.1. Tipos de etiquetas de Códigos de Barras . . . . .	9
2.2. Memoria de contacto . . . . .	10
2.3. Componentes de un sistema RFID . . . . .	15
2.4. Componentes de un lector RFID . . . . .	18
2.5. Capas del Middleware RFID . . . . .	23
3.1. Diagrama del sistema . . . . .	32
3.2. Configuraciones del sistema . . . . .	33
4.1. Configuraciones del sistema . . . . .	35
4.2. Tag RFID . . . . .	36
4.3. Lector RFID . . . . .	37
4.4. Paquete Request[4] . . . . .	39
4.5. Paquete Response[4] . . . . .	40
4.6. Trama de petición de lectura . . . . .	41
4.7. Trama de respuesta de lectura . . . . .	42
4.8. Configuración MAX232 . . . . .	44
4.9. Configuración de red RS485[6] . . . . .	44
4.10. Microcontrolador . . . . .	46
4.11. IAR Workbench . . . . .	47
4.12. Diagrama de flujo del microcontrolador . . . . .	48
4.13. Diagrama de TRIAC . . . . .	49
4.14. Diagrama de Transistor . . . . .	49
4.15. Diagrama del Módulo de Potencia . . . . .	49
4.16. Diagrama esquemático . . . . .	50
4.17. Diagrama PCB . . . . .	50
4.18. Esquemático tarjeta de conversión RS232-RS485 . . . . .	51
4.19. PCB de la tarjeta de conversión RS232-RS485 . . . . .	51
4.20. Modelo 3D de la tarjeta de conversión RS232-RS485 . . . . .	52
4.21. Esquemático tarjeta de controladora . . . . .	52
5.1. Diagrama general del sistema de control de acceso . . . . .	54
5.2. Esquema de Base de Datos . . . . .	56
5.3. Módulo de configuración . . . . .	58
5.4. Agregar un nodo . . . . .	59
5.5. Configurar nodo . . . . .	59
5.6. Diagrama de flujo agregar nodo . . . . .	60
5.7. Ventana abrir nodo . . . . .	61
5.8. Diagrama de la capa de Middleware . . . . .	63
5.9. Diagrama de Lista Ligada . . . . .	64
5.10. Transmisión de datos RS-232 . . . . .	65
5.11. Recepción de repuestas de lectura . . . . .	66
5.12. Tabla Registros . . . . .	68

5.13. Diagrama de bloques del módulo registro de visitantes . . . . .	69
5.14. Módulo de administración de usuarios . . . . .	70
5.15. Búsqueda de usuarios . . . . .	71
5.16. Agregar usuario . . . . .	72
5.17. Diagrama de actividades administración de usuarios . . . . .	74
5.18. Reporte de fatlas, retados en pdf y asistencia en excel . . . . .	76
5.19. Arquitectura de la aplicación web . . . . .	79
5.20. Controlador de la aplicación web . . . . .	80
5.21. Ambiente de desarrollo . . . . .	80
5.22. Reporte de retardos del día y búsqueda de usuarios . . . . .	82
5.23. Reporte de retardos del mes agrupado por departamento . . . . .	82
A.1 Arquitectura de integración del sistema desarrollado . . . . .	87
A.2. Representación del archivo WSDL . . . . .	90
A.3. Página de prueba del Servicio Web . . . . .	91
A.4. Respuesta del Servicio Web . . . . .	94
A.5. XML Schema . . . . .	93
A.6. Tipo de dato complejo XML Schema . . . . .	93
A.7. Proceso BPEL que invoca un servicio web . . . . .	96
A.8. Componente BPEL de interacción humana . . . . .	97
A.9. Adaptador de archivo . . . . .	97
A.10. Consola de prueba de BPEL . . . . .	98
A.11. Visualización de la información en el proceso BPEL . . . . .	99
A.12. Proceso BPEL en espera de interacción humana . . . . .	99
A.13. Aplicación Web de interacción humana . . . . .	100
A.14. Interacción humana, detalle del proceso . . . . .	100
A.15. Escritura de archivo . . . . .	101

# Capítulo 1. Introducción

La tecnología RFID, ha tenido mucho auge en los últimos años debido a la relativa reducción de precios en el mercado, al incremento en sus capacidades y a las ventajas que presenta frente a otras tecnologías de autoidentificación[1].

RFID promete revolucionar la vida de las personas por sus diversas aplicaciones. Al estar involucrada en la cadena de producción y distribución de las fábricas, generará grandes beneficios tales como: especificación de la línea de producción, verificación de la calidad de los productos, elaboración de inventarios automáticos, a partir de que éstos ingresan al almacén de las tiendas; conocimiento sobre el momento de abastecimiento y cobro automático del carrito al pasarlo frente a la caja.

Los beneficios también se incluyen dentro de las actividades cotidianas. El desarrollo de refrigeradores con tecnología RFID permitirá no sólo identificar el momento en que un producto está a punto de caducar sino también en caso de requerir su abastecimiento, se informará al usuario de esta condición. Las lavadoras, por ejemplo, identificarán el ciclo de lavado que le corresponde a determinado tipo de ropa.

Finalmente otra aplicación se encuentra en plantas recicladoras de basura ya que las máquinas emitirán informes acerca del material en que están hechos ciertos productos logrando así facilidad en su separación y agrupamiento.

En fin las aplicaciones de esta tecnología son prometedoras, no obstante, aún se encuentra en proceso de evolución.

En esta tesis, se analizan las ventajas que ofrece la tecnología RFID frente a otras semejantes. Se presentan los elementos que participan en un proyecto de este tipo y pese a que en esta ocasión se orientó al control de acceso, la mayoría de las cuestiones estudiadas aplican para múltiples casos. Se analizó el ciclo completo de este tipo de sistemas, desde que se genera información a partir de los lectores de RFID; hasta el procesamiento de los datos, a partir de aplicaciones cliente servidor y web. Finalmente, se muestra la interacción con una arquitectura SOA en un esquema de este tipo. Es así como esta tesis pretende ser una guía de los elementos a considerar en un proyecto de RFID.



## **1.1. Antecedentes**

A lo largo de los años han surgido distintas tecnologías de autoidentificación[14]. Entre sus múltiples aplicaciones, podemos mencionar la administración del acceso del personal.

La seguridad en el acceso a instalaciones privadas, gestionada en la actualidad por sus respectivas entidades administrativas, genera problemas en el adecuado control de registro de los usuarios que acceden a sus instalaciones y en los horarios de trabajo que estos tienen establecidos.

En algunos lugares se utiliza infraestructura costosa y/o de manejo delicado; material informático de alto valor, tanto en equipos como en información; que en caso de pérdida o daño ocasiona un grave problema para la institución. Debido a esto, surge la necesidad de desarrollar alternativas que permitan resolver el problema de control de acceso, haciendo uso de la tecnología actual, para así explotar al máximo sus capacidades.

## **1.2. Descripción del Problema**

El sistema de control de acceso propuesto en esta investigación surge a partir de la idea de realizar un sistema barato; con tecnología de punta; que su tiempo de vida sea, al menos a mediano plazo; escalable; confiable; seguro y con interfaces amigables.

La tecnología utilizada fue RFID, debido a la seguridad que presenta actualmente, la adaptabilidad que posee para este proyecto, además de ser una innovación tecnológica que poco a poco va tomando fuerza en el sector industrial.

Una ventaja de estas tarjetas es que no necesitan contacto físico (como introducir la tarjeta en una ranura o esperar a que sea reconocido ópticamente); sólo con aproximarla a cierta distancia del lector, la tarjeta será validada. Esto proporciona una “firma virtual”, esto es, el sistema asigna identificadores únicos a cada tarjeta para validar en el sistema dos tipos posibles de ingreso al poseedor de la credencial, según las necesidades de seguridad: aceptado, para usuarios autorizados y, denegado, para usuarios que en determinado acceso no están autorizados para ingresar.

Requerimientos de las aplicaciones de software propuestas para esta tesis:

- Un sistema fácil de configurar y capaz de adaptarse a distintas situaciones que tenga la posibilidad de aumentar el número de dispositivos o accesos que controla.
- Un sistema que le permita a una PC tener comunicación bidireccional con múltiples dispositivos lectores y actuadores (tarjeta controladora de cerradura) a distancias de hasta 1km.

Comunicación:

PC -> Lector : petición de lectura.  
Lector -> PC: respuesta de lectura.  
PC -> Tarjeta controladora: comando para liberar actuador  
Tarjeta controladora -> PC: envío de alarmas (puerta abierta o falla).

- Registro de la hora y fecha en cada entrada y salida de la persona, al momento de ingresar con tarjetas de RFID. Esto proporciona un registro de las personas que acceden al inmueble.
- Los reportes de asistencia y retardos, generados por el sistema, son enviados por correo electrónico en formato pdf o excel, para facilitar el control de asistencia del personal.
- El software permite el registro de visitantes con fotografía, así como modificaciones, altas y bajas de usuarios en el sistema.
- Aplicación web que permite hacer búsquedas de usuarios y verificar sus accesos.
- Se definió una arquitectura orientada a servicios cuyo objetivo es tener la posibilidad de crear procesos que integren diversas aplicaciones o tecnologías heterogéneas, y así poder llevar la información generada a otro nivel.

### 1.3. Trabajo Relacionado

Existen diversos trabajos relacionados con RFID. Algunos de ellos se enfocan en cuestiones de comunicaciones, en los que se realizan pruebas o estudios de las antenas de las etiquetas así como el estudio de las colisiones de las señales[17]. Otros trabajos se enfocan en métodos para optimizar la lectura de estos componentes a través de metales y de líquidos, con los cuales, actualmente, tienen muchas dificultades[18].

Un tema de gran importancia cuando se habla de RFID es el tema de la privacidad, ya que mientras más dispositivos con RFID se utilicen, se está más expuesto a poder ser leído por otra persona la cual podría obtener información de preferencias de productos, redes sociales e inclusive localización. Por este motivo existen muchas investigaciones que intentan resolver esta problemática[2].

Uno de los sectores más beneficiados por la tecnología RFID, o uno de los que más impactos tendrán es la cadena de suministro de bienes de consumo. Existen muchos estudios que analizan los puntos en donde se deberían implementar soluciones RFID en

este tipo de industria y el impacto que estas tendrían en inventarios, almacenes, puntos de venta y usuario final[20].

Toda la información generada por los dispositivos de RFID, debe ser controlada, filtrada y administrada de algún modo. Actualmente existe una capa de software dedicada a estas funciones llamada Middleware de RFID. Existen múltiples investigaciones relacionadas con este tema, en donde se plantean diversas arquitecturas y funcionalidades para esta capa[16]. Como esta debería conectarse con el hardware y como debería entregar la información recolectada a las capas superiores, que pueden ser otras aplicaciones, otras redes e inclusive dispositivos móviles[26].

Debido a los nuevos requerimientos que plantean estas nuevas tecnologías, surgen arquitecturas como EDA (arquitectura orientada a eventos) y SOA (arquitectura orientada a eventos) que intentan satisfacer las necesidades de integración y manejo de altos volúmenes de información. Existen diversos trabajos en donde se plantea como utilizar estas arquitecturas para explotar de mejor forma los datos generados[37].

## **1.4. Motivación y planteamiento del problema**

La tecnología de RFID promete revolucionar la vida de las personas. Día a día salen al mercado dispositivos con mayores capacidades y menores precios. Los estándares se van robusteciendo, lo que da como resultado que, en un futuro muy próximo, estos dispositivos estén por todas partes. Por ello debemos estar preparados para poder explotar la información que generan estos dispositivos

En este proyecto se propone una aplicación específica del uso de esta tecnología, además se exploraron todos los elementos involucrados en este tipo de soluciones, por lo cual se desarrollaron componentes de software intentando utilizar lo último en cuanto a estándares frameworks y tecnologías de software disponibles, como es el caso de JSF[32], AJAX y Business Components[35].

En la actualidad, el paradigma de desarrollo de sistemas está evolucionando hacia arquitecturas en las que se exponen servicios con cierta funcionalidad y se programan aplicaciones que consumen local o remotamente estos servicios. Esto último trae ventajas en cuanto a la facilidad de integración de las aplicaciones, la reutilización de componentes y la flexibilidad del mantenimiento. En esta tesis se empleó una arquitectura del estilo mencionado anteriormente para compartir la información generada por el sistema de control de acceso con otros sistemas.

## **1.5. Objetivos**

### **Generales**

- Establecer todos los puntos a considerar cuando se desarrolla un sistema que involucra RFID, así como definir las ventajas y desventajas que esta tecnología tiene contra otras soluciones de autoidentificación.
- Establecer los puntos que deben considerarse al momento que se desarrolla un sistema RFID.
- Definir las ventajas y desventajas de la tecnología RFID frente a otras soluciones de autoidentificación.
- Desarrollar un sistema que permita realizar el control de acceso de personal, en múltiples puntos, utilizando la tecnología de identificación por radiofrecuencia.
- Definir una arquitectura que permita la integración de esta aplicación –control de acceso- con otros sistemas.

### **Particulares**

- Proporcionar un protocolo de comunicación basado en RS-485 que permita a un ordenador controlar diversos dispositivos lectores y actuadores a distancias de hasta 1 km.
- Desarrollar una aplicación configurable y escalable para poder adaptarse a distintas topologías.
- Involucrar en el sistema tarjetas de radiofrecuencia para identificar al personal.
- Implementar una serie de aplicaciones cliente-servidor que permitan controlar y administrar la operación del sistema de accesos.
- Obtener un sistema funcional y de bajo costo.
- Desarrollar una aplicación web que permita explotar los datos generados, de forma remota.

## 1.6. Organización

El contenido de esta tesis está organizado de la siguiente manera:

- Capítulo 2

Se presenta un estudio comparativo de las tecnologías de autoidentificación como son: código de barras, dispositivos biométricos, memorias de contacto, tarjetas magnéticas y RFID. Se definen los conceptos, estándares y tecnologías relacionadas con RFID.

- Capítulo 3

Se explica de manera general, la funcionalidad e infraestructura del sistema y todos los módulos que lo conforman.

- Capítulo 4

Se exponen detalladamente los componentes de hardware utilizados, y las tarjetas que fueron desarrolladas.

- Capítulo 5

Se describen los distintos módulos de software, cliente-servidor y web, así como el diseño de la base de datos.

- Capítulo 6

Se dan las conclusiones de este trabajo y se propone el trabajo futuro.

# Capítulo 2.

## Tecnologías de Autoidentificación y RFID

A lo largo de los años han surgido distintas tecnologías orientadas a la autoidentificación. En este capítulo se hace un análisis de las tecnologías de identificación existentes y se presenta un comparativo de sus principales características, a ventajas y desventajas. Se muestra un estudio más profundo de la tecnología de RFID y los componentes necesarios para desarrollar un proyecto con esta tecnología.

### 2.1. Tecnologías de Autoidentificación

Desde hace años están disponibles en el mercado distintas tecnologías para la identificación de productos, personas e incluso animales. En ese sentido, uno de los principales exponentes ha sido el código de barras, el cual, ha logrado penetrar prácticamente en todas las cadenas de distribución, almacenes y sistemas de control de acceso, por citar algunos ejemplos. Sin embargo, en los últimos 10 años, se ha dado un *boom* de nuevas tecnologías, o más bien de aquellas que ya existían, pero que hasta ahora pudieron entrar al mercado masivo. La razón principal son todas las ventajas tecnológicas que ofrecen frente a los esquemas tradicionales, aunado a la baja en los precios.

#### 2.1.1. Comparación de Tecnologías de Autoidentificación

Dentro del ámbito de la tecnología de identificación, aplicado al control de acceso, se pueden encontrar diversas tecnologías como: sistemas biométricos, tarjetas magnéticas, código de barras, RFID y memorias de contacto que se describen en los siguientes párrafos.

##### 1. Acceso con Sistemas Biométricos

Este tipo de identificación se realiza a través del análisis y/o medición de características físicas. Algunas de las técnicas biométricas que existen son[25]:

- Reconocimiento de iris
- Reflexión retinal
- Geometría de la mano
- Geometría facial
- Termografía mano, facial
- Huellas dactilares
- Patrón de la voz

La identificación biométrica ofrece una ventaja significativa, dado que bajo este sistema, se identifica explícitamente a la persona, no así a alguna credencial u otro objeto.

La razón por la cual no es aplicable para ciertos problemas una tecnología de este tipo es porque no existen sistemas que ofrezcan una confiabilidad cercana al 100 por ciento.

La mayoría de los sistemas de este tipo tienen una eficiencia menor a lo deseable. Otra desventaja de este tipo de sistemas es que son más costosos.

## **2. Acceso con Tarjetas magnéticas**

Estos sistemas se basan en la lectura de una banda magnética. Utilizan señales electromagnéticas para registrar y codificar información en una banda que puede ser leída por una máquina para identificación instantánea. La aplicación más difundida es la de las tarjetas de crédito[7].

Sus ventajas son proporcionar agilidad en el acceso, dar identificación única al poseedor, bajo costo, además de que no son fácilmente falsificables. Sin embargo, su uso continuo las deteriora físicamente como consecuencia de la fricción al momento de la lectura. Además si alguna tarjeta es acercada a alguna fuente electromagnética, relativamente fuerte, puede modificar la información que contiene, perdiendo con ello su utilidad.

## **3. Acceso con Tarjetas de Código de Barras**

El código de barras se inventó hace más de 25 años[10] y durante este tiempo, ha sido la tecnología más utilizada por los comercios para identificar los productos en venta. Este tipo de identificación se realiza codificando datos en una imagen formada por combinaciones de barras y espacios. Las imágenes son leídas por equipos especiales de lectura óptica a través de los cuales se pueden comunicar datos a la computadora.

Proporciona las mismas ventajas que las tarjetas magnéticas y no es necesario el contacto físico entre la tarjeta y el lector, no obstante debe de existir una línea de vista entre ellos. Este tipo de sistema es barato, sin embargo, estas tarjetas son fácilmente falsificables o alterables siendo esto una gran debilidad para un sistema estricto de control de acceso, por lo que esta desventaja es significativa para descartar el uso de tarjetas por código de barras para esta aplicación.

Se han inventado alrededor de 270 diferentes simbologías para soportar requerimientos específicos y aproximadamente 50 de éstos se utilizan ampliamente en la actualidad. Cada una de estas simbologías cae dentro de alguna de las siguientes tres categorías[10]:

- **Lineal.** Consiste en líneas verticales, de diferentes anchos, con espacios blancos que separan dos líneas adyacentes. El máximo número de caracteres que pueden ser codificados, mediante esta metodología, son 50.
- **Dos dimensiones.** Esta simbología tiene la mayor capacidad de almacenamiento, el máximo número de caracteres que pueden ser codificados es de 3,750.
- **Tres dimensiones (Bumpy).** Este tipo de código de barras es leído, utilizando el relieve de las barras, es decir, no depende del contraste entre barras oscuras y espacios, por lo tanto puede ser embebidos directamente en los productos como por ejemplo en llantas o en partes plásticas directamente desde el molde. La ventaja de estos códigos es que pueden ser utilizados en ambientes de uso rudo.



Figura 2.1.: Tipos de etiquetas de Códigos de Barras.

#### 4. Acceso con Tarjetas de RFID (Identificación por Radio Frecuencia)

La tecnología de radiofrecuencia se desarrolló en 1940, como medio para la identificación de los aviones aliados y enemigos durante la Segunda Guerra Mundial. Años más tarde evolucionó, logrando así ser utilizada en la industria ferroviaria para el seguimiento de los coches del ferrocarril y para los años 60's y 70's, su uso se enfocó en la seguridad de materiales nucleares[1].

En la actualidad RFID se utiliza principalmente en el rubro de seguridad, como es el caso de los cruces fronterizos, credenciales de identidad, en el control vehicular, identificación de ganado, envío de paquetes, control de equipaje en los aeropuertos y de artículos para renta o préstamo (películas y libros) en videoclubes y bibliotecas, en la industria automotriz, para los procesos de automatización y seguimiento, en el sector agrícola y en el de administración de flora y fauna, para rastrear al ganado y a los animales, así como en el mercado minorista como dispositivo antirrobo[2].

La Tecnología de Identificación por Radiofrecuencia es un método electrónico que consiste en asignar un código de información a un producto, proceso o persona y usar esta información para identificar o acceder a información adicional al respecto. Los sistemas de identificación por radiofrecuencia consisten generalmente de dos componentes:



- El "transponder", pequeña etiqueta electrónica (tag) que contiene un minúsculo microprocesador y una antena de radio. Esta etiqueta contiene un identificador único que puede ser asociado a una persona o producto.
- El "lector", que obtiene el identificador del "transponder".

La tecnología del transponder se basa en la aplicación de un transmisor/receptor encapsulado.

El receptor se puede activar por medio de una batería incorporada (transponder activo) o puede ser alimentado por la señal enviada por el lector (transponder pasivo). El lector genera un campo magnético cuya señal de RF es captada por el receptor del chip. Éste, a su vez activará al transmisor, el cual enviará un mensaje codificado único. Este mensaje es decodificado por el lector y procesado por la computadora.

## 5. Acceso con Memorias de Contacto

Los botones de memoria de contacto son un tipo específico de tecnología de auto identificación que requiere un contacto físico con el botón para leer los datos de la etiqueta. La adopción ha sido muy limitada, comparada con la pequeña inversión a realizar y las innovaciones que ha habido en esta área.

La memoria de contacto no ha tenido una amplia adopción como solución de auto identificación. Una de las principales preocupaciones al respecto es que los tres mayores sistemas conocidos de esta tecnología en la actualidad son propietarios. Y si cualquiera de estos es descontinuado, será complicado encontrar un sustituto.

Pero entre sus ventajas están la de ser dispositivos de múltiples lecturas y escrituras, además de ser muy resistentes, ya que pueden ser empleados en entornos hostiles y con vibraciones propias de aplicaciones de manufactura[9].



Figura 2.2.:Memoria de contacto

Habiendo detallado las características de cada sistema por separado, se puede resumir lo expuesto en el Cuadro1.

	<b>Código de Barras</b>	<b>Banda Magnética</b>	<b>Memoria de Contacto</b>	<b>Sistemas Biométricos</b>	<b>RFID Pasivo</b>	<b>RFID activo</b>
<b>Modificación de la información</b>	No Modificable	Modificable	Modificable	No Modificable	Modificable	Modificable
<b>Seguridad de los Datos</b>	Mínima	Media	Alta	Alta	Variable (baja a alta)	Alta
<b>Capacidad de Almacenamiento de datos</b>	-Lineales(8-30 caracteres) - 2D hasta 7.200 caracteres	Hasta 128 bytes	Hasta 8MB	No aplica	Hasta 64 KB	Hasta 8MB
<b>Precio</b>	Bajo	Medio-Bajo	Alto (cerca de US\$1 por memoria)	Alto	Medio (menos de US\$0.50 por tag)	Muy Alto (US\$10 a US\$100 por tag)
<b>Estándares</b>	Estables	Estables	Proprietarios, no estándar	No estándar	Evolucionando hacia estándar	Propietario y en evolución hacia estándar
<b>Ciclo de Vida</b>	Corto	Mediano	Largo	Indefinido	Indefinido	Depende de la batería (3 a 5 años)
<b>Distancia de Lectura</b>	Línea de vista y (hasta 1.5m)	Requiere contacto	Requiere contacto	Depende del biométrico	No requiere línea de vista ni contacto Hasta 10m.	No requiere línea de vista ni contacto Hasta 100 m. y mayores
<b>Interferencia Potencial</b>	Cualquier modificación en las barras y objetos entre el código y el lector	Bloqueo del contacto	Bloqueo del contacto	Puede ser bloqueo del contacto, o bloqueo de línea de vista e inclusive el ruido.	Ambientes o campos que afecten la transmisión de radio frecuencia	La interferencia es muy limitada, debido a la potencia de transmisión.

Tabla 2.1.: Tecnologías de Control de Acceso

## **2.1.2. Comparación entre tecnologías de Radiofrecuencia y Código de Barras**

RFID es una tecnología que ha tenido gran crecimiento en los últimos años, de hecho se piensa que puede reemplazar al código de barras, empero, por el momento no reemplazará a ninguna de las otras tecnologías de auto identificación existentes, ya que cada una tiene sus propias ventajas y desventajas.

La tecnología de RFID se ha visto como el sucesor del código de barras, porque ofrece diferentes ventajas sobre esta tecnología. Por ejemplo: una etiqueta de RFID no necesita línea de vista directa con el lector para poder ser identificada y, dependiendo de la tecnología que se utilice, la distancia entre el transponder y el lector puede ser desde un par de centímetros hasta cientos de metros.

Otra ventaja es que con RFID se identifica un producto como único, es decir, productos iguales pueden ser diferenciados por una clave contenida en su etiqueta de RFID, a diferencia del código de barras que para productos iguales es el mismo. Una etiqueta de RFID es mucho más complicada de clonar que un código de barras que puede ser igualado por medio de una fotocopia.

Un código de barras no puede ser modificado, una vez que se ha impreso, por lo tanto, es un tecnología de solo lectura. En contraste, los tags de RFID pueden tener la capacidad de lectura/escritura, ya que cuentan con una memoria direccionable que puede ser modificada miles de veces durante su periodo de vida. Esta capacidad hace de RFID una tecnología muy poderosa.

Otro problema del código de barras es la capacidad simultánea de lectura, que en cualquier sistema de código de barras es uno. Esto significa que sólo se puede identificar un solo producto al mismo tiempo, a diferencia de la tecnología RFID que puede realizar múltiples lecturas simultáneas.

Y finalmente una etiqueta de RFID tiene una mayor durabilidad y un menor desgaste, debido a que, si un código de barras sufre de desgaste o tachaduras, ya no podrá ser leído.

El único punto a favor del código de barras es que su precio puede llegar a ser insignificante. Por ello existe la creencia acerca de que RFID no reemplazará, por completo, al código de barras, sino más bien convivirán.

## **2.1.3. Ventajas de la Identificación por radiofrecuencia**

A continuación se describen las principales ventajas de la tecnología de RFID en cuanto a seguridad, línea de vista, velocidad de lectura, mantenimiento, reescritura, entre otras.

- **Seguridad.** Es una tarjeta que por su diseño tecnológico, no puede duplicarse fácilmente. Cada una posee un código distinto y no permite que varios usuarios puedan tener una tarjeta duplicada. Es una diferencia fundamental cuando se le compara con los sistemas de banda magnética o código de barras, donde la duplicación de tarjetas es bastante frecuente. Son ideales para situaciones de máxima seguridad y alta tecnología.
- **Sin necesidad de alineación o línea vista.** [1] De todos es el sistema más ágil y práctico, por varias razones. Una de ellas es que no necesita que la tarjeta sea pasada por una ranura o en el sentido correcto, lo que le da una mayor agilidad y practicidad de uso. Esto garantiza el éxito de la implementación de un sistema nuevo, donde, en general, los usuarios se resisten a ser controlados, pero al ser tan cómodo su uso, brinda una aceptación muy grande por parte de los usuarios.
- **Inventarios de alta velocidad.** Múltiples dispositivos pueden ser leídos simultáneamente, esto puede ahorrar tiempo si se compara con otras tecnologías, en las que es necesario alinear los dispositivos para leerlos uno por uno.
- **Lectores sin mantenimiento.** Los lectores son unidades sin partes móviles, lo que garantiza un correcto funcionamiento sin límite de uso y sin que haya que hacerles algún tipo de mantenimiento. También se pueden instalar a la intemperie sin que las inclemencias del tiempo, como altas y bajas temperaturas ambientales, los dañen. La distancia de lectura, dependerá del tipo de lector. Los hay con distintos alcances dependiendo de su aplicación. Pueden ir desde 7 cm. a 2 m., siempre hablando de proximidad pasiva.
- **Tarjetas sin desgaste.** La tarjeta no tiene fricción alguna con el lector, por lo cual no se desgasta y su vida útil es prolongada. Esto permite su reutilización tras asignarlas, al personal de nuevo ingreso. El resultado es la optimización de recursos. Las tarjetas de proximidad vienen de varias formas. La más difundida y estándar es una de plástico bastante rígido, que está preparado para que se le pueda personalizar por medio de una impresión.
- **Reescribible.** Algunos tipos de etiquetas RFID, pueden ser leídas y escritas en múltiples ocasiones. En caso de que se aplique a componentes reutilizables, puede ser una gran ventaja.
- **Factibilidad.** El área de aplicación de la tecnología de RFID es muy amplia.
- **Otras Tareas.** Además de almacenar y transmitir datos, una etiqueta de RFID, puede ser diseñada para desempeñar otras funciones como medir condiciones de humedad o temperatura en el ambiente.

## **2.2. Identificación por Radio Frecuencia (RFID)**

Como se mencionó en la sección 3.1, la tecnología RFID no es nueva, más bien tardó varios años en popularizarse debido a los altos costos y a sus limitantes.

A finales de los 90s, la tecnología RFID, con la dirección de EAN Internacional y la UCC, adquirió un nuevo desafío: la reducción de tamaño de los dispositivos además de los costos[8].

En aquella época, EAN internacional y la UCC fueron punta de lanza en el desarrollo del los estándares globales para que RFID facilitará el comercio global, proporcionando trazabilidad a toda la cadena de suministro alrededor del mundo. Paralelamente a esto, en el MIT se desarrollaban otras investigaciones como el desarrollo de Auto ID (predecesor del estándar EPC).

Su gran empuje surgió cuando Wal-Mart anuncio que requeriría que sus 100 principales proveedores integraran etiquetas de RFID en sus contenedores de productos para Enero del 2005 [3].

El principio fundamental de RFID consiste en un transponder y un lector de RFID. El lector interroga al transponder utilizando cierta frecuencia y el transponder contesta a distancia con la información que contiene, que puede ser un número identificador de producto. El lector recoge esta información y la envía a una unidad de cómputo para su procesamiento.

### **2.2.1. Tecnología RFID**

Existen 3 componentes básicos en un sistema de RFID[22] :

1. El tag, etiqueta o transponder de RFID consiste en un pequeño circuito, integrado con una pequeña antena, capaz de transmitir un número de serie único hacia un dispositivo de lectura, como respuesta a una petición. Algunas veces puede incluir una batería.
2. El lector, (el cual puede ser de lectura o lectura/escritura) está compuesto por una antena, un módulo electrónico de radiofrecuencia y un módulo electrónico de control.
3. Un controlador o un equipo anfitrión, comúnmente una PC o Workstation, en la cual corre una base de datos y algún software de control.

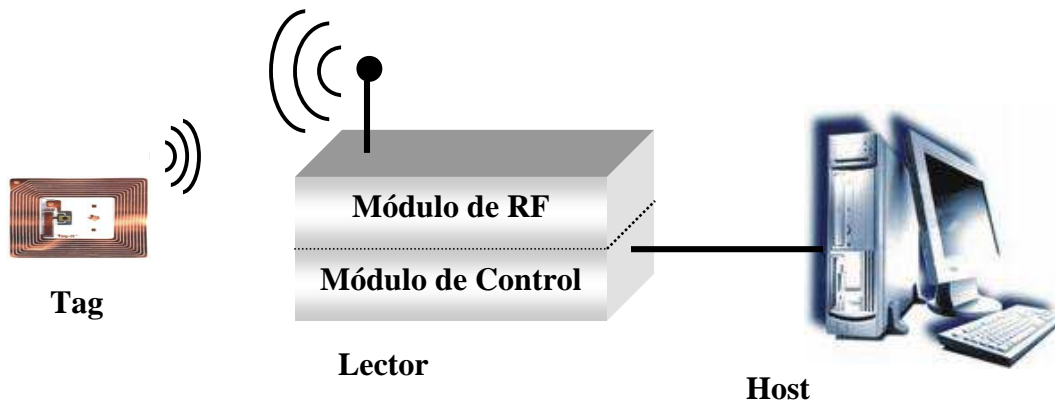


Figura 2.3.: Componentes de un sistema RFID.

La tecnología de identificación por radiofrecuencia[19] puede ser dividida principalmente en 3 categorías:

1. *Sistemas pasivos*, en los cuales las etiquetas de RFID no cuentan con una fuente de poder. Su antena recibe la señal de radiofrecuencia enviada por el lector y almacena esta energía en un capacitor. La etiqueta utiliza esta energía para habilitar su circuito lógico y para regresar una señal al lector. Estas etiquetas pueden llegar a ser muy económicas y pequeñas, pero su rango de lectura es muy limitado.
2. *Sistemas activos*. Utilizan etiquetas con fuentes de poder integradas, como baterías. Este tipo de etiquetas integra una electrónica más sofisticada, lo que incrementa su capacidad de almacenamiento de datos, interfaces con sensores, funciones especializadas, además de que permiten que exista una mayor distancia entre lector y etiqueta (20m a 100m). Este tipo de etiquetas [7] son más costosas y tienen un mayor tamaño. Pueden permanecer dormidas hasta que se encuentran dentro del rango de algún lector, o pueden estar haciendo broadcast constantemente.
3. *Sistemas Semi-Activos*. Emplean etiquetas que tienen una fuente de poder integrada, la cual energiza al tag para su operación [17], sin embargo, para transmitir datos, una etiqueta semi-activa utiliza la potencia emitida por el lector. En este tipo de sistemas, el lector siempre inicia la comunicación. La ventaja de estas etiquetas es que al no necesitar la señal del lector para energizarse (a diferencia de las etiquetas pasivas), pueden ser leídas a mayores distancias, y como no necesita tiempo para energizarse, estas etiquetas pueden estar en el rango de lectura del lector por un tiempo substancialmente menor para una apropiada lectura. Esto permite obtener lecturas positivas de objetos moviéndose a altas velocidades.

Tanto los tags activos como los pasivos pueden adicionalmente ser clasificados de la siguiente forma:

- Solo Lectura (RO)

En estos dispositivos, los datos son grabados en el tag durante su fabricación, para esto, los fusibles en el microchip del *tag* son quemados permanentemente utilizando un haz láser muy fino. Después de esto, los datos no podrán ser reescritos. Este tipo de tecnología se utiliza en pequeñas aplicaciones, pero resulta poco práctico para la mayoría de aplicaciones más grandes, que intentan explotar todas las bondades de RFID.

- Una Escritura, Muchas Lecturas (WORM)

Un tag WORM, puede ser programado sólo una vez, pero esta escritura generalmente no es realizada por el fabricante sino por el usuario justo en el momento que el tag es creado. Este tipo de etiquetas puede utilizarse en conjunto con las impresoras de RFID, las cuales escriben la información requerida en el tag.

- Lectura y Escritura (RW)

Estas etiquetas, pueden ser reprogramadas muchas veces, típicamente este número varía entre 10,000 y 100,000 veces, incluso mayores. Esta opción de reescritura ofrece muchas ventajas, ya que el tag puede ser escrito por el lector, e inclusive por sí mismo en el caso de los tags activos. Estas etiquetas regularmente contienen una memoria Flash o FRAM para almacenar los datos.

## 2.2.2. Lectores de RFID

El lector de RFID es un dispositivo que puede leer y escribir datos hacia tags RFID compatibles.

El lector es el componente central del hardware en un sistema de RFID y tiene los siguientes componentes:

- **Transmisor**

El transmisor emite potencia y envía el ciclo de reloj a través de su antena hacia los tags que se encuentran dentro de su rango de lectura.

- **Receptor**

Este componente recibe las señales analógicas provenientes del tag a través de la antena y envía estos datos al microprocesador, donde esta información es convertida en su equivalente digital.

- **Antena**

Esta antena va conectada directamente al transmisor y al receptor. Existen lectores con múltiples puertos para antenas, lo que les permite tener múltiples antenas y extender su cobertura.

- **Microprocesador**

Este componente es responsable de implementar el protocolo de lectura empleado para comunicarse con tags compatibles. Decodifica y realiza verificación de errores a las señales recibidas. Adicionalmente, puede contener cierta lógica para realizar filtrado y procesamiento de bajo nivel de los datos leídos, esto es, eliminar lecturas duplicadas o erróneas.

- **Memoria**

La memoria es utilizada para almacenar información como los parámetros de configuración del lector, además de una lista de las últimas lecturas realizadas, de modo tal que si se pierde la comunicación con la PC, no se pierdan todos los datos.

- **Canales de Entrada/Salida**

Estos canales permiten al lector interactuar con sensores y actuadores externos. Estrictamente hablando, es un componente opcional, pero incluido en la mayoría de los lectores comerciales de la actualidad.

- **Controlador**

El controlador es el componente que permite a una entidad externa, sea un humano o un software de computadora, comunicarse y controlar las funciones del lector. Comúnmente los fabricantes integran este componente como un firmware.

- **Interfaz de Comunicación**

Esta interfaz provee las instrucciones de comunicación, que permiten la interacción con entidades externas, mediante el controlador, para transferir datos y recibir comandos. Un lector puede tener distintos tipos de interfaz como se discute más adelante, por ejemplo: RS-232, RS-485, interfaz de red, entre otras.

- **Fuente de Alimentación**

Este componente provee de alimentación eléctrica a los componentes del lector y regularmente consiste en un cable con un adaptador de voltaje, conectado hacia la toma de corriente. Pero en los últimos años se han incrementado el número de lectores de tipo pistola, los cuales son móviles y su fuente de alimentación es una batería recargable.



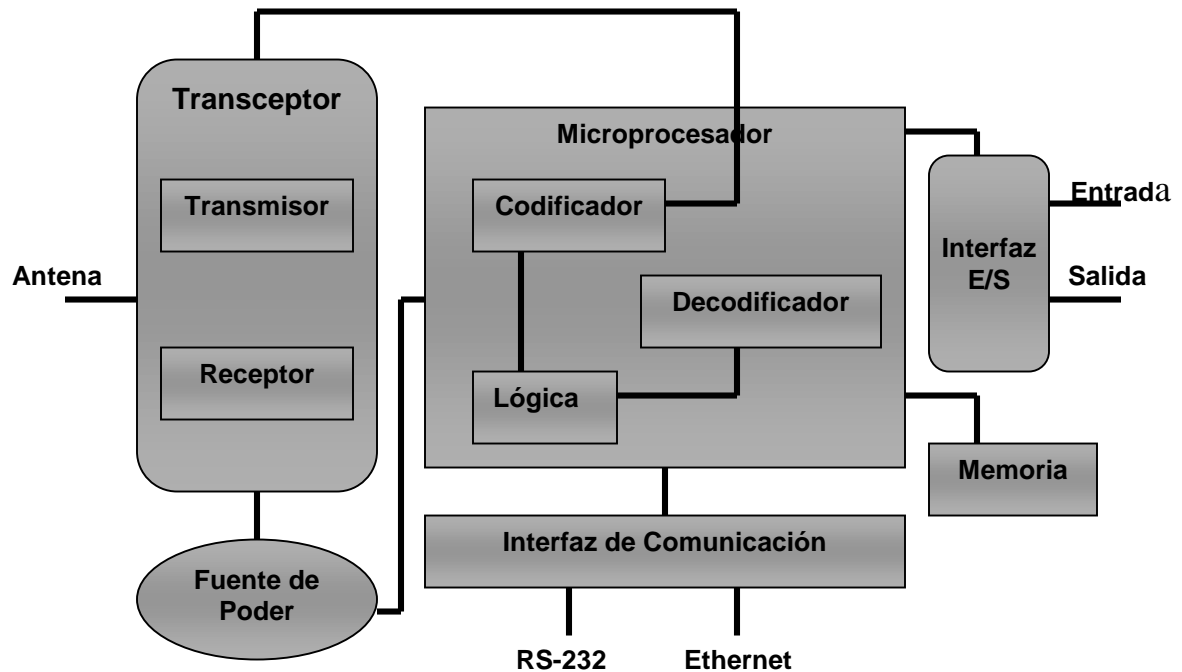


Figura 2.4.: Componentes de un lector RFID.

### 2.2.3. Frecuencias

Las frecuencias de RFID [1] pueden ser divididas en 4 rangos:

- 1) Baja Frecuencia (9-135 KHz). Los sistemas que utilizan este rango de frecuencia tienen la desventaja de una distancia de lectura de sólo unos cuantos centímetros. Sólo pueden leer un elemento a la vez.
- 2) Alta Frecuencia (13.56 MHz). Esta frecuencia es muy popular y cubre distancias de 1cm a 1.5 m. Típicamente las etiquetas que trabajan en esta frecuencia son de tipo pasivo.
- 3) Ultra High Frequency (0.3-1.2GHz). Este rango se utiliza para tener una mayor distancia entre la etiqueta y el lector (de hasta 4 metros, dependiendo del fabricante y del ambiente). Estas frecuencias no pueden penetrar el metal ni los líquidos a diferencia de las bajas frecuencias pero pueden transmitir a mayor velocidad y por lo tanto son buenos para leer más de una etiqueta a la vez.
- 4) Microondas (2.45-5.8GHz). La ventaja de utilizar un intervalo tan amplio de frecuencias es su resistencia a los fuertes campos electromagnéticos, producidos por motores eléctricos, por lo tanto, estos sistemas son utilizados en líneas de producción de automóviles. Sin embargo, estas etiquetas requieren de mayor potencia y son más costosas, pero es posible lograr lecturas a distancias de hasta 6 metros [8]. Una posible aplicación es el cargo automático en autopistas, en donde se coloca un tag en los automóviles que funciona como tarjeta de prepago. En las casetas de cobro existen

lectores, antenas y sistemas que permiten realizar el cargo correspondiente, sin la necesidad de que el auto se detenga.

## Anticolisión y Múltiples Lecturas

Para que un lector de RFID tenga la capacidad de comunicarse con múltiples tags simultáneamente, es necesario implementar algoritmos anticolisión. Un lector antes de emitir una señal de lectura no sabe cuántos tags se encuentran a su alrededor, entonces debe existir un plan de cómo realizar estas lecturas, de lo contrario en el caso en que hubiera cientos de tags en el rango de lectura intentando contestar al mismo tiempo, podrían existir colisiones.

Existen tres técnicas anticolisión. Espacial, por frecuencia y en dominio de tiempo. Las tres son utilizadas para establecer un orden jerárquico, o algún método aleatorio en el sistema.

## Transferencia de Datos

Los sistemas de RFID que operan en la banda de baja frecuencia tienen una transferencia de datos de baja velocidad, en el orden de Kbits/s. Estas velocidades aumentan de acuerdo con la frecuencia de operación, alcanzando tasas de Mbit/s en las frecuencias de microondas.

### 2.2.4. Estándares

La tecnología RFID debe cumplir con estándares creados por organizaciones como ISO y EPC.

#### A) ISO

ISO tiene 3 estándares para [12] RFID: ISO 14443 (para sistemas sin contacto), ISO15693 (para sistema de proximidad) e ISO 18000 (para especificar la interfaz aérea para una variedad de aplicaciones).

#### B) EPC

EPC global es una organización sin fines de lucro que ha desarrollado una amplia gama de estándares para la identificación de productos. Los estándares EPC están enfocados a la cadena de suministro y particularmente definen la metodología para la interfaz aérea; el formato de los datos almacenados en una etiqueta RFID, para la identificación de un producto, captura, transferencia, almacenamiento y acceso de estos datos; así como el middleware y la base de datos que almacena esta información.

Las funciones de EPC o Código Electrónico de Producto son similares a las de UPC o Código de Producto Universal encontrado en la tecnología de código de barras. EPC es un esquema de identificación para identificar objetos físicos de manera universal por medio de etiquetas RFID. El código EPC en una etiqueta RFID puede identificar al fabricante, producto, versión y número de serie, y adicionalmente provee un grupo de dígitos extra para identificar objetos únicos.

La red de EPCglobal es un grupo de tecnologías que habilita la identificación automática e inmediata de elementos en la cadena de suministro y la compartición de dicha información.

La tecnología RFID involucra colocar las etiquetas RFID en los objetos, la lectura de etiquetas (idealmente sin intervención humana) y el paso de la información a un sistema dedicado de infraestructura de Tecnologías de la Información. Con dicha infraestructura se pueden identificar objetos automáticamente, rastrear, monitorear y activar eventos relevantes.

### C) ONS

EPCglobal ha desarrollado un sistema llamado ONS (Object Naming Service) que es similar al DNS (Domain Name Service) utilizado en Internet. ONS actúa como un directorio para las organizaciones que desean buscar números de productos en Internet.

### D) Gen 2

EPCglobal ha trabajado con un estándar internacional para el uso de RFID y EPC, en la identificación de cualquier artículo, en la cadena de suministro para las compañías de cualquier tipo de industria, esto, en cualquier lugar del mundo. El consejo superior de la organización incluye representantes de EAN International, Uniform Code Council, The Gillette Company, Procter & Gamble, Wal-Mart, Hewlett-Packard, Johnson & Johnson, Checkpoint Systems y Auto-ID Labs.

El estándar gen 2 de EPCglobal fue aprobado en diciembre de 2004, y es probable que llegue a formar la espina dorsal de los estándares en etiquetas RFID de ahora en adelante. EPC Gen2 es la abreviatura de “EPCglobal UHF Generation 2”.

### E) Otros

Existen, así mismo, muchos más estándares, pero enfocados a industrias específicas, por ejemplo: el AIAG B-11 (Automotive Industry Action Group) para identificación de llantas y ANSI MH10.8.4, para aplicaciones estándar de RFID con contenedores reutilizables. Las siguientes son algunas organizaciones que han producido algún estándar relacionado con RFID, o han desarrollado alguna función regulatoria al respecto:

- ANSI ( American National Standards Institute )
- AIAG ( Automotive Industry Action Group )

- EAN.UCC ( European Article Numbering Association International, Uniform Code council )
- EPCglobal
- ISO ( International Organization for Standarization )
- CEN ( Comité Européen Normalisation )
- ETSI ( European Telecommunications Standards Institute )
- ERO ( European Radocommunications Office )
- UPU ( Universal Postal Union )
- ASTM (American Society for Testing Materials)

## 2.2.5. Conectividad

Cuando se desarrolla un sistema de RFID [10] la elección de la conectividad de red para los lectores de RFID, es una consideración importante.

Históricamente los lectores de RFID han tendido a usar comunicaciones seriales, ya sea RS-232 o RS-485. Actualmente la mayoría de los fabricantes intenta habilitar Ethernet en sus lectores e inclusive conectividad wireless 802.11.

Siendo las opciones las siguientes:

- *RS-232*. Este protocolo provee sistemas de comunicación confiables de corto alcance. Tiene ciertas limitantes como una baja velocidad de comunicación, que va de 9600 bps a 115.2 kbps. El largo del cable está limitado a 30 metros, no cuenta con un control de errores y su comunicación es punto a punto.
- *RS-485*. El protocolo RS-485 es una mejora sobre RS-232, ya que permite longitudes de cables de hasta 1,200 metros. Alcanza velocidades de hasta 2.5 Mbps y es un protocolo de tipo bus lo cual permite a múltiples dispositivos estar conectados al mismo cable.
- *Ethernet*. Se considera como una buena opción, ya que su velocidad es más que suficiente para los lectores de RFID. La confiabilidad del protocolo TCP/IP sobre Ethernet asegura la integridad de los datos enviados y finalmente al ser la infraestructura común para las redes, la mayoría de las instituciones ya cuentan con una red de este tipo, lo que permite una instalación más sencilla y menos costos de integración.
- *Wireless 802.11*: Se utiliza en la actualidad en los lectores de RFID móviles. Además de que esta solución reduce los requerimientos de cables y por lo tanto de costos.
- *USB*: Pensando desde la tendiente desaparición del puerto serial en las computadoras, algunos proveedores de lectores RFID han habilitado sus equipos para poder comunicarse mediante el puerto USB.

Con los avances tecnológicos actuales, se habla también que los datos generados por los dispositivos de RFID, puedan ser [10] movilizados a través de la red de telefonía celular.

## **2.2.6. Middleware**

Las nuevas políticas propuestas por organismos como Wal-Mart, el Departamento de Defensa de EUA, Tesco, Target y Metro AG han forzado a los proveedores a poner sus planes de RFID en práctica, lo más rápido posible. Esto ha provocado que no se exploten al máximo los beneficios operacionales de RFID al utilizar los datos RFID para mejorar sus procesos.

Esto significa que las empresas deben incorporar de una manera inteligente los datos RFID en los procesos de negocio que apliquen, de modo que estos impacten en la toma de decisiones de la empresa. Esta tarea no es nada sencilla pero se resuelve a través de una capa de software llamada middleware.

El middleware es el software que permite la conexión entre el hardware de RFID y los sistemas de Tecnologías de la Información de la empresa como pueden ser sistemas legado, ERP (Enterprise Resource Planning), CRM (Client Relationship Management), sistemas de inteligencia de negocio, entre otros.

El middleware es una plataforma para filtrar, administrar y rutear datos de las redes de RFID hacia los sistemas empresariales.

El middleware de RFID[21] debe incluir una combinación balanceada de cinco capas:

- 1) Administración del Lector. Debe permitir al usuario configurar, monitorear y aplicar comandos directamente a los lectores, a través de una interfaz común.
- 2) Administración de los datos. Una vez que el middleware de RFID captura los datos enviados por los lectores, debe ser capaz de filtrar lecturas duplicadas o erróneas y rutear los datos a su correcto destino[23].
- 3) Integración de Aplicaciones. Debe proveer características de conectividad, ruteo y mensajes, requeridas para integrar los datos RFID con sistemas existentes como SCM (Supply Chain Management), WMS (Warehouse Management System), CRM (Client Relationship Management) o ERP (Enterprise Resource Planining), idealmente a través de una arquitectura orientada a servicios (SOA).
- 4) Integración con socios de negocio. Algunos de los beneficios más prometedores de RFID vendrán al compartir los datos RFID con los socios de negocio para mejorar los procesos colaborativos, para lo cual es necesaria la compatibilidad con protocolos de transporte B2B (Business to Business).

5) Administración y escalabilidad en la arquitectura. La adopción de RFID producirá mucha información, y el middleware de RFID es la primera línea de defensa para un procesamiento de los datos confiable. Esto significa que las plataformas de middleware de RFID deben estar habilitadas para funcionar en ambientes de alta disponibilidad o en cluster, con la capacidad de hacer un balanceo de carga dinámico y re-enrutamiento de los datos en caso de que un servidor falle.

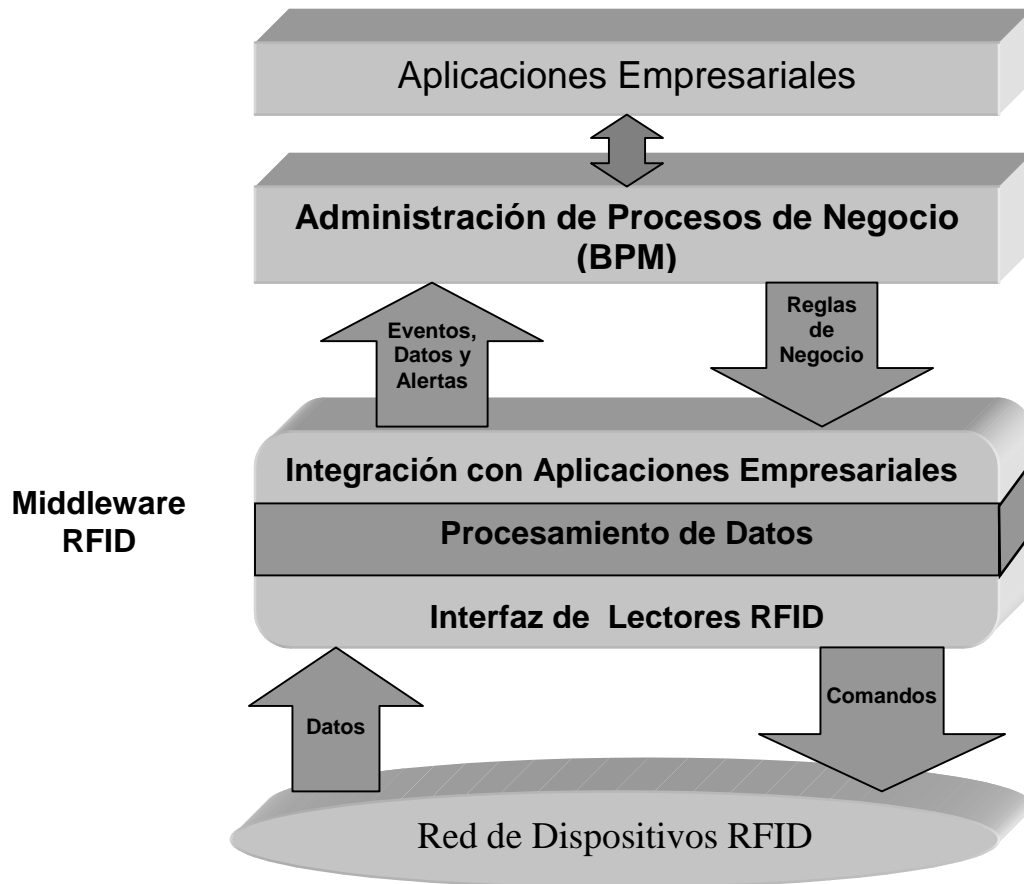


Figura 2.5.: Capas del Middleware RFID

Existen en la actualidad ofertas de los mayores fabricantes de software que intentan resolver el problema del middleware de RFID. Cisco estableció que para el 2009, la mayor parte del tráfico en sus redes estará relacionada con EPC, y que para el 2014 el número de lectores EPC a escala mundial alcanzara los 300 millones.

## ALE

Eventos del Nivel de Aplicaciones (ALE) [12]. Es una especificación de EPCglobal. El rol de la interfaz ALE, dentro de la arquitectura de una red EPCglobal, es proveer independencia entre los componentes de la infraestructura que adquieren los datos EPC crudos, los componentes de la arquitectura que filtran y cuentan los datos y las aplicaciones que utilizan esos datos. Esto permite que los componentes cambien sin

necesidad de modificar las aplicaciones. ALE es una especificación de servicios web que contiene un archivo WSDL para definir, configurar y pedir reportes sobre los datos EPC en tiempo real. Existe un esquema XML para hacer la petición de reportes.

## Plataforma Middleware

Algunas de las empresas líderes de software en el ámbito mundial como IBM, Oracle, Microsoft, SUN y Progress han desarrollado plataformas middleware, aprovechando todo el portafolio de productos con que cuentan, para tratar de dar soluciones más integrales, en las cuales, inclusive ya existía la integración con algunas aplicaciones.

### 2.2.7. Seguridad

#### A) Privacidad

La inminente ubicuidad de las etiquetas de RFID, también representa una potencial amenaza a la privacidad del consumidor. La más simple etiqueta de RFID enviará su identificador único de 64-128 bits a cualquier lector cercano. Esto significa que cualquier persona con un lector, podría escanear estas etiquetas y obtener información personal a través de documentos como la licencia de conducir o el pasaporte; hábitos de consumo, mediante los accesorios que porta e inclusive la cantidad de dinero que alguien trae en la cartera mediante la lectura de su cartera.

#### B) Esquemas de Seguridad para RFID

1. *Desactivar Etiquetas* [2]. El enfoque más simple para proteger la privacidad del cliente consiste en desactivar a las etiquetas de RFID antes de que sean puestas en manos del consumidor. Una vez desactivadas, estas etiquetas no pueden volver a ser reactivadas. Una etiqueta puede ser desactivada al enviarle un comando especial. En realidad este enfoque no es el más adecuado, ya que una tecnología tan poderosa y de bajo costo como RFID, será inevitablemente utilizada en muchas aplicaciones, las cuales requerirán que las etiquetas permanezcan activas cuando estén en posesión del cliente.

2. *La Jaula de Faraday*. Una etiqueta de RFID puede ser protegida por medio de una jaula de Faraday que consiste en un contenedor hecho de una malla de metal que es impenetrable a las señales de radio (de ciertas frecuencias). Si se adicionara una etiqueta RFID a los billetes, una jaula de Faraday en las carteras, sería una buena solución[19].

3. *Interferencia Activa*. Es otra forma de proteger a las etiquetas de la vista de otros. El consumidor podría cargar un dispositivo que transmita señales de radio para bloquear o alterar el funcionamiento de cualquier lector de RFID cercano. Este esquema podría causar severas alteraciones a todos los sistemas de RFID cercanos, incluso de aquellos cuya aplicación sea legítima y no representen un riesgo a la privacidad[11].

## 2.2.8. Tendencias

RFID se muestra actualmente como una tecnología con mucho potencial, por lo que aún queda mucho por desarrollar e implementar en los diferentes campos que la integran. A continuación se mencionan algunas de las principales tendencias.

### A. Industria

Surgirán nuevos estándares industriales y legislaciones gubernamentales.

La disminución en el costo de los componentes, especialmente el de las etiquetas, jugará un rol muy importante para determinar su ubicuidad.

Un nivel de etiquetado, por artículo, es la última frontera del desarrollo de RFID. Este concepto permitiría todo tipo de aplicaciones en la cadena de suministro, empero, quedan por resolverse los problemas de seguridad y privacidad por parte de los consumidores.

### B. Aplicaciones

Aplicaciones como inventarios [25] en tiempo real y una visibilidad total durante toda la cadena de distribución de los productos. Permitirá que la industria sea más eficiente y ahorrará costos ya que se podrían eliminar los centros de distribución y recibir los productos directamente de los proveedores.

Innovaciones en las aplicaciones para beneficio de los consumidores como control de acceso, pagos electrónicos, cuidado de pacientes, cliente frecuente, marcas deportivas y muchas más. Los proveedores de Software [15] de manejo de almacenes y cadenas de suministro, ofrecerán nuevos niveles de funcionalidad en sus aplicaciones, tomando ventaja de los datos RFID

### C. Diseños de etiquetas alternativos

Muchos factores afectan el rango de lectura y precisión de las etiquetas, incluyendo aquellos que son físicos y del ambiente. Algunos ejemplos son: la detección cerca de metales o líquidos y condiciones de clima extremas como baja temperatura o alta humedad. Además de simplemente mejorar estos aspectos en la tecnología existente, se ha empleado física alternativa para cubrir estas limitantes.

La mayor parte del trabajo en esta área incluye desarrollos de etiquetas chipless (etiquetas sin chip). Un ejemplo de estas etiquetas es el de superficie de onda acústica [16](SAW), la cual envuelve la propagación de las ondas de radio frecuencia acústica. Otras prometedoras tecnologías de chipless, que tienen el potencial de revolucionar las aplicaciones de RFID, utilizan nanotecnología, genómica e incluso química para generar etiquetas sin chip para la identificación de objetos únicos.



## D. Etiquetas sensoras

Etiquetas cuyo empaquetamiento integra sensores que pueden monitorear, grabar e inclusive reaccionar ante todo tipo de condiciones ambientales. Estas etiquetas promueven toda una nueva gama de aplicaciones.

## E. Arquitectura

Los sistemas de RFID generan montañas de información que necesita ser sincronizada, filtrada, analizada, administrada y todo esto en tiempo real. Cada etiqueta es esencialmente un dispositivo de cómputo, que actúa como un nodo en una red de eventualmente millones o billones de dispositivos.

Esta nueva red es diferente y aún más compleja que Internet, debido al número de nodos que pueden existir (un número mucho mayor de nodos). Esto significa que las arquitecturas e [17] infraestructuras de cómputo tradicionales no serán las adecuadas para manejar estos altos volúmenes de información. Considerando el escenario de una cadena de suministro como Wal-mart, en donde se etiquetan todos los productos de todas las tiendas, el número de elementos etiquetados puede ser de 1000 millones o más.

Esto significa que, la información generada por esos 1000 millones de artículos, representa 12 gigabytes. Si estos artículos son leídos una vez cada 5 minutos, en algún punto de la cadena de suministro, generarán cerca de 1.5 terabytes por día. Para alcanzar estas capacidades, actualmente se investiga y desarrolla un nuevo concepto en el desarrollo de una nueva arquitectura de microprocesadores llamada Chip Multi-Threading (CMT). Esta arquitectura permite la ejecución eficiente de múltiples tareas simultáneamente, esto es, cómputo paralelo llevado a la capa del procesador. Adicionalmente, los lectores de RFID cada vez tendrán mayor poder de procesamiento local, lo cual disminuirá dramáticamente la carga de los recursos de cómputo centralizados.

## F. Inteligencia de Negocios

Como se ha mencionado RFID genera una gran cantidad de información, pero el valor real de esta información es utilizarla para realizar mejores decisiones de negocios. La capacidad de responder nuevas preguntas o descubrir patrones en los datos que proveen de mayor inteligencia al negocio.

## G. RFID Implantado en Humanos

La empresa FDA tiene planes de comercializar un chip de RFID implantado debajo de la piel [24], con el objetivo de almacenar el expediente médico de la persona que permita a los doctores escanear a los pacientes para identificarlos y proporcionarles el mejor tratamiento y los medicamentos más adecuados. Se espera que estos dispositivos salven vidas y reduzcan lesiones ocasionadas por tratamientos no adecuados.

# Capítulo 3. Descripción del Sistema

El sistema desarrollado intenta cubrir la mayor parte de los puntos o tecnologías que podrían estar involucrados en cualquier tipo de proyectos o implementaciones de esta naturaleza.

La idea es sentar las bases de todos los problemas que se deben resolver cuando se desea implementar un sistema con tecnología RFID.

Por lo tanto en esta tesis, aunque se trata un problema en particular, como es el Control de Acceso, la solución propuesta con ciertas variantes, podría resolver otros casos de estudio.

## 3.1. Infraestructura

A continuación se describe la infraestructura utilizada para la realización de esta tesis.

- Transponders (Badges o Tarjetas RFID)

*Rol del componente.*- Son los componentes que guardan en su interior el número de identificación del usuario, el cual intercambian con el lector al ser aproximados a él.

*Componente empleado.*- Los tags utilizados para este proyecto fueron los de tipo tarjeta, de la serie Tiris de Texas Instruments.

- Lector de RFID

*Rol del componente.*- Este componente permite realizar las lecturas de los transponders, y enviar la información obtenida a la PC.

*Componente empleado.*- El lector utilizado es el WallPlate de Texas Instruments Tiris de 13.56 MHz de frecuencia y basado en el estándar ISO 15693.

- PC

*Rol del componente.*- Este componente es el orquestador del sistema, ya que en él se ejecutan la mayoría de los programas de software, que permiten la operación del sistema y que les dicen a los lectores de RFID y a los actuadores qué operaciones realizar. Así mismo, permite recolectar información y explotarla, de modo que sea valiosa para la

institución o empresa. El tipo de información con la que se podría contar es la siguiente: identificar retardo, ausencias, accesos no autorizados, entre otros.

*Componente empleado.-* La PC utilizada fue una laptop con 1GB en RAM, procesador Intel Pentium M a 1.86 GHz y 60 GB en disco duro. Para una implementación pequeña (de 1 a 8 puntos de acceso y hasta 300 accesos por día), un equipo con características similares debería ser suficiente, pero para una implementación mayor, se tendría que evaluar otro tipo de arquitectura, separar la base de datos en otro servidor, e inclusive para implementación empresarial de gran tamaño, se podría pensar en una arquitectura de alta disponibilidad con redundancia en caso de posibles fallos.

- **Manejador de Base de Datos**

*Rol del componente.-* La base de datos, es un componente que no puede faltar en cualquier implementación de sistemas de tecnologías de la información. Y este caso no es una excepción. Se requiere de una Base de Datos que almacene toda la información generada por el sistema.

*Componente empleado.-* A lo largo de esta tesis se trabajo con tres distintos motores de Bases de Datos: MySQL, Microsoft SQLServer 2005 y Oracle Express Edition. La razón de utilizar 3 DBMS distintos fue con el objetivo de desarrollar un proyecto que pudiera ser compatible con los DBMS más populares, de modo que pudiera ser implementado en distintas organizaciones, sin que el DBMS fuera un problema.

- **Cámara Web**

*Rol del componente.-* La cámara permite tomar las fotografías de los visitantes.

*Componente empleado.-* En esta tesis se empleo una webcam logitech muy sencilla y de poca resolución, pero este es un componente que se puede mejorar, con base en la calidad de fotografías que se requiera y desde luego, al presupuesto.

- **Actuadores**

*Rol del componente.-* Una vez que la PC procesa los datos recibidos de los lectores de RFID, debe decidir qué operación se debe ejecutar sobre los actuadores, es decir, si estos se liberarán o no. Algunos ejemplos de actuadores que se podrían controlar serían plumas, torniquetes, motores, chapas eléctricas o magnéticas, entre otras.

*Componente empleado.-* El actuador con el que se trabajo, fue una chapa o cerradura eléctrica.

- Tarjeta controladora y microcontrolador

*Rol del componente.*- Este componente permite establecer comunicación con la PC y controla la activación de los actuadores mediante su módulo de potencia.

*Componente empleado.*- Fue necesario construir esta tarjeta usando un microcontrolador de montaje superficial. El microcontrolador que se utilizó fue el MSP430F149 de Texas Instruments con 60 KB de memoria de programa, 2048 de SRAM y 2 puertos USART.

- Tarjeta de conversión RS-232 a RS-485

*Rol del componente.*- El protocolo empleado para mantener la comunicación con los distintos componentes del sistema es el RS-485. La problemática que tuvo que resolverse fue: la dificultad para encontrar computadoras que cuenten con esta tarjeta, por lo tanto, se desarrolló una tarjeta que permite traducir del protocolo RS-232 a RS-485, ya que es más común que una computadora cuente con una tarjeta con puerto RS-232.

*Componente empleado.*- Esta fue construída para esta tesis. Cabe mencionar, que en el mundo actual de las computadoras y de las tecnologías de la información, los adelantos tecnológicos se dan a pasos agigantados. Como consecuencia de ello el número de computadoras con tarjetas RS-232, es cada vez menor, por lo cual será una buena opción trabajar con otro tipo de tecnología de comunicación que podría ser USB, ethernet, wifi e inclusive bluetooth.

- Otros

Adicionalmente se utilizaron otros componentes de menor jerarquía, pero igual de importantes que los anteriores, entre los que destacan: cable UTP, componentes electrónicos diversos, reguladores de voltaje, conectores tipo DB9 y RJ45.

## **3.2. Componentes del Sistema**

### **1. Módulos de Hardware**

- Tarjeta de Conversión RS-232 a RS-485
- Tarjeta Controladora de Actuadores
  1. Módulo de Comunicaciones
  2. Microcontrolador
  3. Módulo de Potencia

## 2. Módulos de Software

Los componentes de software presentados, son cliente-servidor, web y componentes de la arquitectura orientada a servicios:

### a) Componentes Cliente Servidor

- Configuración del Sistema

Permite definir los identificadores de todos los lectores de RFID y de las tarjetas controladoras en la red, así como definir como están conformados los distintos puntos de acceso. Una vez que se ingresa esta información, el programa genera un archivo, que permite que el módulo de control de acceso se autoconfigure, cada vez que es iniciado.

- Control de Acceso

Este módulo permite la interacción e intercambio de datos entre la PC y los distintos lectores de RFID y actuadores en la red.

Así mismo, almacena toda la información de los accesos realizados en la base de datos.

- Registro de Visitantes

Este módulo permite el registro de visitantes con fotografía, y la asignación de una tarjeta con accesos restringidos.

- Administración de Usuarios

Esta aplicación permite realizar altas, bajas y cambios de los usuarios del sistema. Permite definir una fecha de expiración para los accesos válidos de los usuarios, y su hora de entrada y salida (turno de trabajo).

- Generación de Reportes

Este módulo permite obtener el resumen de ausencias y retardos de los usuarios al final del día. Envía los resultados por correo electrónico, como archivos adjuntos, en formato excel y pdf.

### b) Componentes Web

- Administración de Usuarios

Este es un componente que permite la visualización de información de usuarios, permite revisar los registros de acceso en el sistema por usuario, por departamento y por día, mes o año.

### c) Componentes SOA (Arquitectura Orientada a Servicios)

- Web Services

Expone la funcionalidad de obtener las faltas del día como servicio web.

- Integración del sistema con otras aplicaciones por medio de BPEL

Se desarrolló un proceso BPEL capaz de interactuar con distintos servicios y tecnologías.

## **3.3. Arquitectura del Sistema Desarrollado**

Uno de los objetivos de este trabajo es resolver el problema del control de acceso con ayuda de la tecnología de Identificación por Radiofrecuencia, proponiendo un sistema altamente efectivo y con componentes de bajo costo.

- Conectividad

La computadora, los lectores de RFID, las tarjetas controladoras y actuadores se encuentran conectados a través de un cable UTP, utilizando el protocolo RS-485.

- Funcionamiento Básico

El funcionamiento básico se encuentra en uno de los módulos de software, que constantemente se encuentra haciendo un pooling, tipo round robin, entre todos los lectores de la red RS-485, enviando peticiones de lectura.

Cuando un lector de RFID recibe la instrucción de realizar una lectura, éste envía una señal al ambiente en búsqueda de etiquetas de RFID, si encuentra alguna, obtendrá su número de identificación y lo enviará de vuelta a la computadora.

El módulo de control de acceso procesará este identificador y decidirá si el usuario tiene permitido el acceso en ese punto, y de ser así, enviará una señal a la tarjeta controladora, para que libere el actuador correspondiente y se permita el acceso.

Simultáneamente, el módulo de control de acceso, almacenará toda la información generada en la Base de Datos.

- **Funcionalidad Adicional**

El sistema, también permite el registro de visitantes con fotografía, la administración de los usuarios del sistema, la generación de reportes de retardos y ausencias, consulta web de la información de los usuarios en el sistema e integración del sistema con otras aplicaciones, mediante el uso de una arquitectura orientada a servicios.

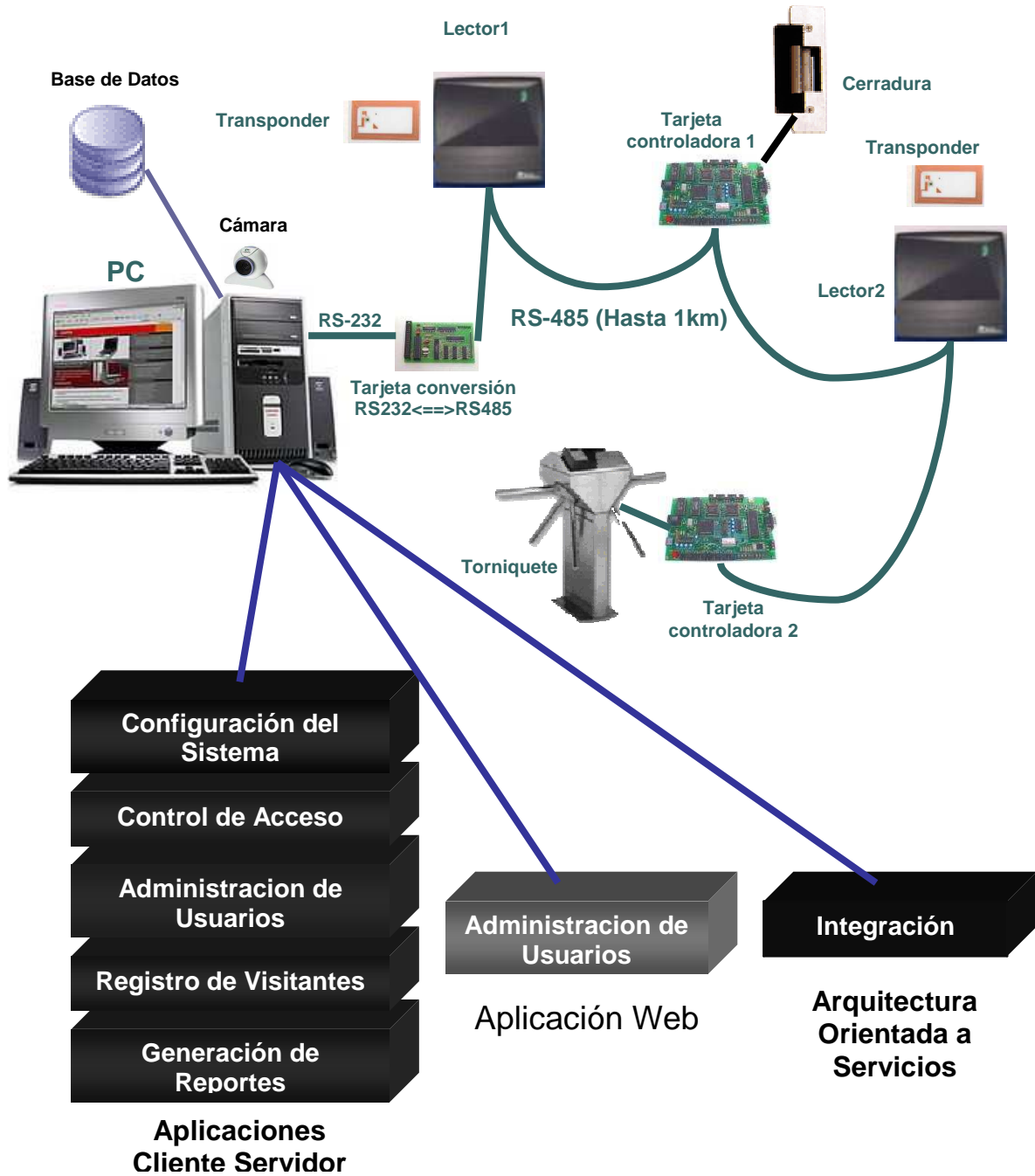


Figura 3.1.: Diagrama del sistema

El sistema es lo suficientemente flexible para permitir dos tipos de configuración en un punto de acceso: con dos lectores y un actuador, o bien, con un sólo lector y un actuador. Esta configuración (un lector y un actuador) hace sentido en puntos en los que, la logística o configuración del acceso, facilitan la utilización de un solo lector para controlar entradas y salidas (Ej. Situaciones, donde solo se requiera controlar la entrada y no así la salida).

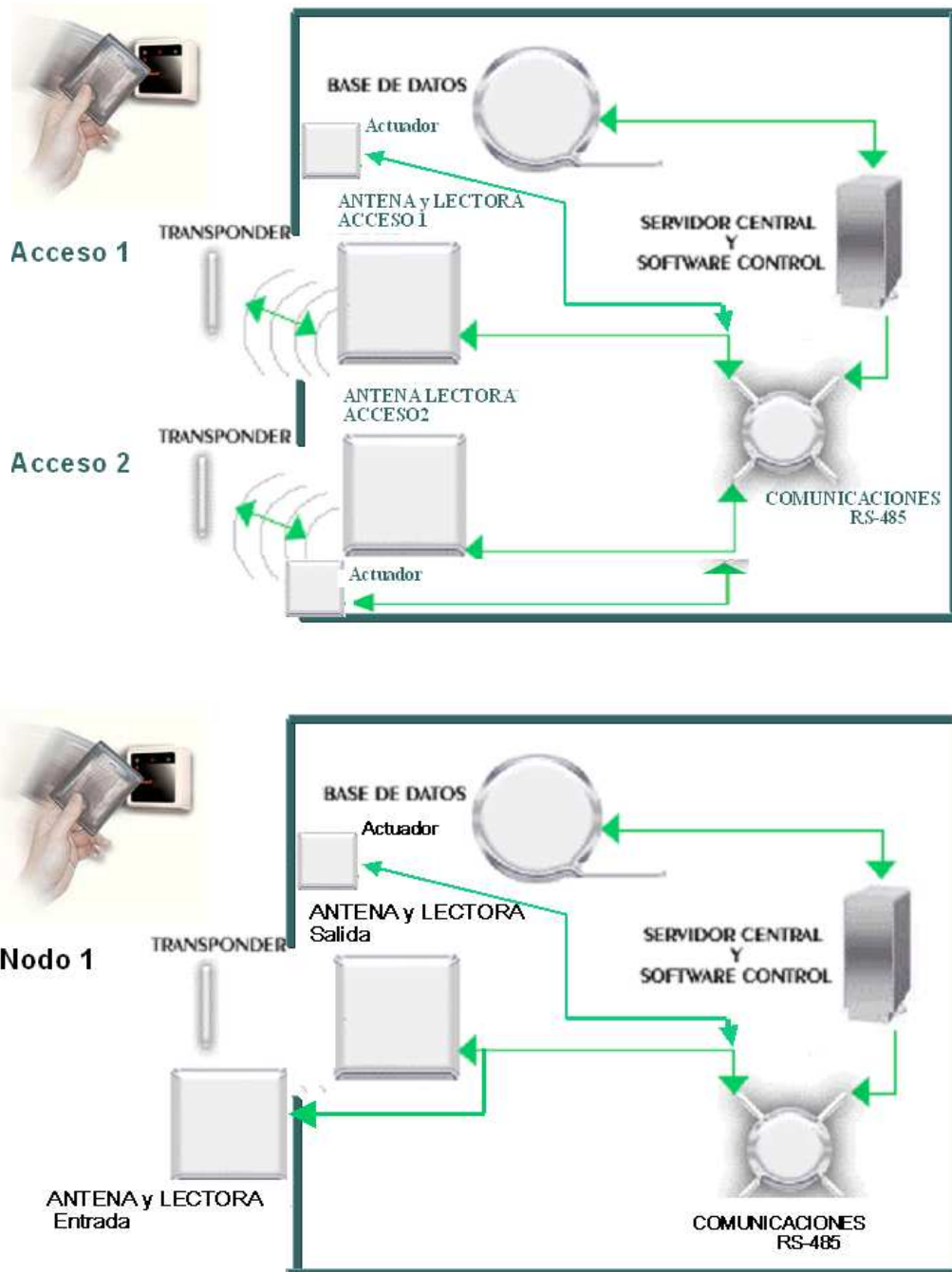


Figura 3.2.: Configuraciones del sistema.





# Capítulo 4. Diseño de Hardware

En este capítulo se describen todos los componentes utilizados y desarrollados en el proyecto, las capacidades de los lectores y tags de RFID utilizados y su modo de operación.

Tanto los lectores de RFID como los actuadores están conectados a una red RS-485. Se eligió este protocolo, debido a sus capacidades, en cuanto a distancia, que permiten tener dispositivos interconectados a más de 1km. La tarjeta RS-485, es poco común en las computadoras, por lo que fue necesario desarrollar una tarjeta que funcionara como interfaz entre la red RS-485 y el puerto RS-232 de la computadora.

En la red también existen dispositivos actuadores. Para tener control sobre ellos, se creo una tarjeta que permite la conectividad con la red RS-485 y a la vez tiene la capacidad de controlar la activación de los actuadores.

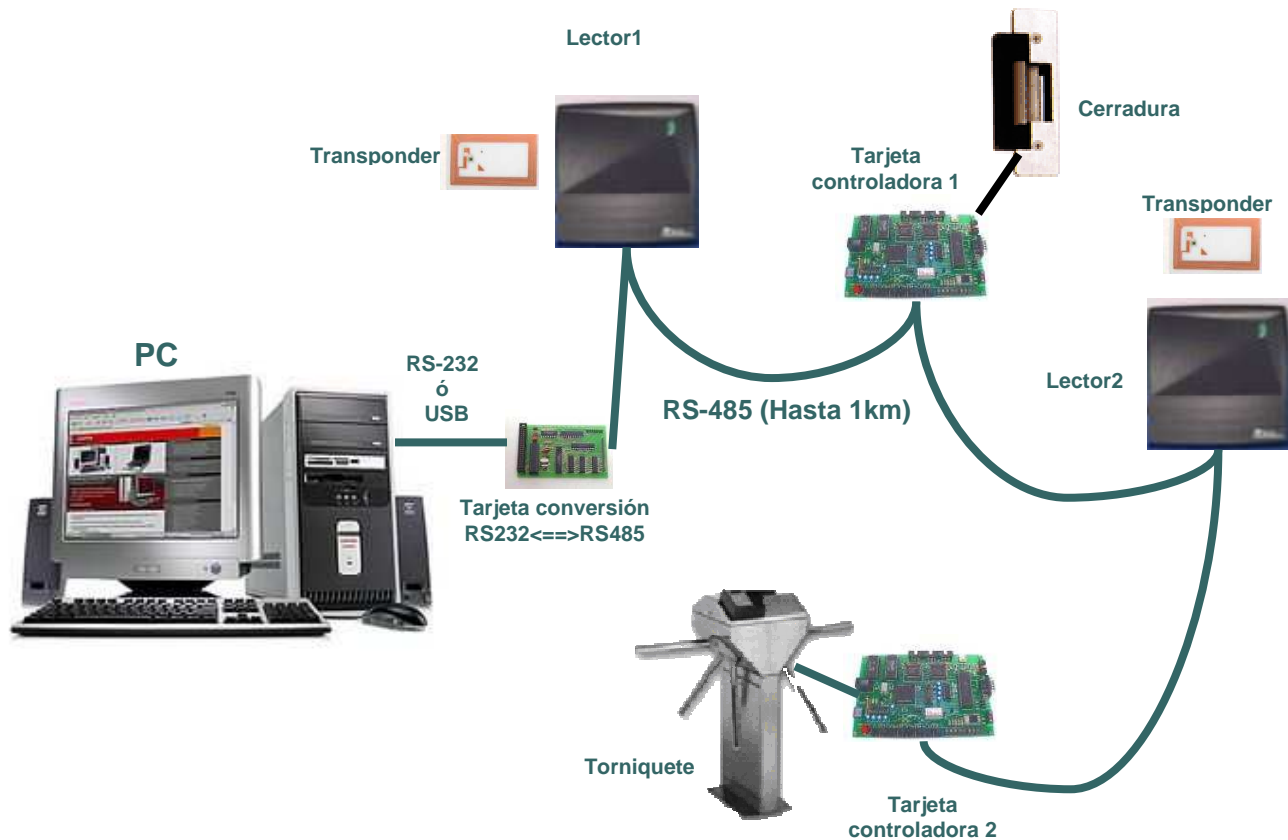


Figura 4.1.: Configuraciones del sistema.

## 4.1. Dispositivos RFID

En esta sección se describen los componentes de un sistema de RFID, el transponder y el lector de RFID.

### 4.1.1 Transponder (Tarjetas RFID)

Las tarjetas de RFID que se utilizaron son de Texas Instruments de alta frecuencia, operan a 13.56 MHz., siguen el estándar ISO 15693 de tarjetas de proximidad, son de lectura y escritura, tienen 2k bits de memoria organizados en 64 bloques de 32 bits cada uno y permiten la identificación simultánea[8].

Estas tarjetas son los elementos que proporcionarán la identificación, por radiofrecuencia, del usuario que accede al interior del inmueble. Por su construcción, proporcionan alta seguridad y no son fácilmente duplicables.

Cada tarjeta está basada en el estándar ISO 15693. Desde su fabricación, es grabada con identificador único (UID) de 8 bytes que no es posible modificar. Este identificador es el que se utilizó en este proyecto para identificar a los usuarios, ya que en la base de datos, esta relacionando cada uno con un UID.

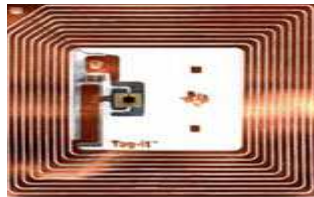


Figura 4.2.: *Tag* RFID.

### 4.1.2 Lector de RFID

El lector utilizado fue el Wall Plate Reader S6410[29] con tecnología TIRIS de Texas Instruments. Es un lector de alta frecuencia, lo que le permite leer hasta 10 tags de manera simultánea[5].



Figura 4.3.: Lector RFID.

Principales características del Lector de RFID:

- Alta Frecuencia: 13.56 MHz
- Estándar: ISO 15693
- Voltaje: entre +9 y +14 VDC
- Corriente promedio: Normal 50mA, durante una lectura 130 mA
- Tasa de Baudios RS-485: 9600, 19200 y 38400
- Protocolos soportados: Wiegand 26-64 bits y RS-485
- Integridad de Datos: Wiegand 150m, RS485 1219 m
- Rango de Lectura: Hasta 20 cm.
- Temperatura de Operación: -20°C a +70°C

Este lector no puede ser montado sobre estructuras metálicas, porque generan interferencia, lo cual provoca disminución dramática en el rango de lectura.

El wallplate, viene encapsulado en una carcasa de plástico, y contiene los cables listos para conectarse y ponerse en funcionamiento de inmediato, pero como el dispositivo soporta distintos protocolos, estos cables se deben de conectar basándose en la Tabla 4.1.

Cabe mencionar que en esta Tabla 4.1 falta la funcionalidad de otros dos cables de colores verde (Wiegand Data 0) y blanco (Wiegand Data 1), los cuales permiten la transmisión y recepción de datos para el protocolo Wiegand.

<b>Color</b>	<b>Wiegand</b>	<b>RS-485</b>	<b>Función</b>
Negro	Esencial	Esencial	Tierra(Gnd)
Rojo	Esencial	Esencial	F.Poder(9-14 v.)
Gris	NA	Esencial	Rs-485 (A o -)
Violeta	NA	Esencial	RS-485(B o +)
Café	Opcional	Opcional	Led Rojo
Naranja	Opcional	Opcional	Led Verde
Amarillo	Opcional	Opcional	Audio
Azul	Opcional	Opcional	Hold

Tabla 4.1.: Configuración del Lector[4].

## Modo de Operación del lector de RFID

El lector utilizado tiene la capacidad de operar en dos modos:

- Automático

En el modo automático, el lector se encuentra realizando lecturas de manera periódica indefinidamente. En el momento en que un tag entre en su rango de lectura, éste será leído. El problema de este modo de operación es que en la arquitectura que se planteó, se tuvieron varios lectores conectados en una red RS-485. Si todos operan en modo automático y todos realizan lecturas al mismo tiempo, saturarán la red y generará colisiones, ocasionando pérdida de información.

- Por comando

En el modo comando, el lector se encuentra en un estado pasivo, y solo realiza lecturas cuando recibe un comando del host. Este esquema, permite evitar las colisiones en una red, ya que el host puede comunicarse con los lectores, uno por uno, evitando que estos transmitan al mismo tiempo. Este es el modo que se utilizó en este proyecto.

Un lector puede recibir comandos de configuración, que le indiquen la tasa de baudios a la que debe trabajar, prender o apagar el led o el buzzer, y puede recibir comandos para realizar lecturas de tarjetas. Existen muchas variantes, en cuanto al comando, para realizar una lectura, ya que es posible solicitar la lectura de algún tag específico, o de

algún bloque de memoria de los tags, e inclusive la escritura de ciertos datos en éstos últimos.

Pero independientemente del comando que se trate, este debe de cumplir con cierto protocolo para que pueda ser entendido por el lector.

### Paquete Request (PC-Lector RFID)

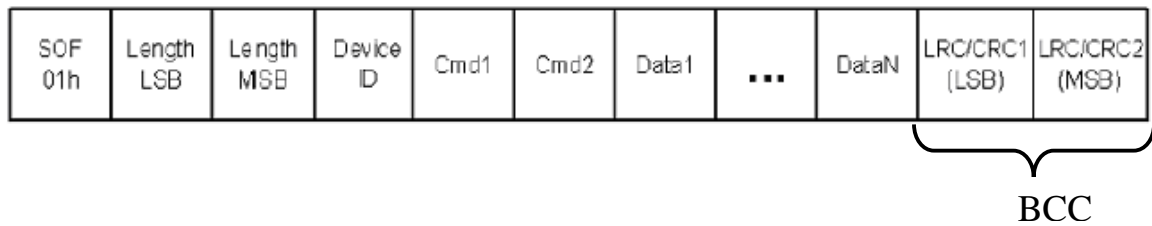


Figura 4.4.: Paquete Request[4].

En la Figura 4.4 se muestra el formato del paquete de request que envia la PC al lector.

Donde

- *SOF* (Inicio de Trama) delimita el inicio del comando ( $01_{16}$ ), tamaño 1 byte
- *Length*, determina el tamaño completo del paquete, incluyendo *SOF*, tamaño 2 bytes
- *Device ID*, identifica el tipo de dispositivo, en este caso el tipo de dispositivo es ( $10_{hex}$ ), tamaño 1 byte.
- *Cmd1* y *Cmd2*, especifican la acción que debe realizar el lector, tamaño 1 byte cada uno.
- *Data*, contiene los parámetros y datos del comando, el tamaño depende del comando. Puede ser de 0 a 1000 bytes.
- *LRC/CRC* (control de redundancia cíclica, longitudinal) le permite al lector validar que recibió correctamente el paquete de request, ya que LCR/CRC son calculados por el componente que transmite el paquete y agregados al final de este, y son recalculados por el componente receptor. Los resultados obtenidos deben ser los mismos para considerar una recepción correcta, tamaño 2 bytes. En esta tesis el CRC, fue calculado con ayuda del software de Texas Instruments para el lector utilizado.

## BCC (Block Check Character)

BCC se utiliza para la detección de errores y consta de 16 bits que se calculan incluyendo a SOF. BCC consiste en 2 partes: el byte LS, que es un Chequeo de Redundancia Longitudinal (LRC) y el byte MS que es el complemento de LRC.

LRC se calcula realizando operaciones OR-Exclusivas sobre todos los bytes del paquete.

## Paquete Response (Lector RFID-PC)

El paquete de datos, con el que responde el lector, tiene exactamente el mismo formato. La diferencia radica en que los campos Cmd, ahora, en vez de contener comandos, se envían banderas que indican si el comando se ejecutó satisfactoriamente.

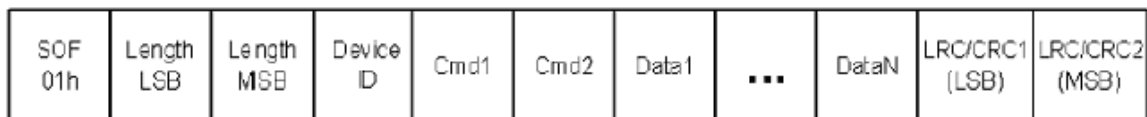


Figura 4.5.: Paquete Response[4].

A continuación se muestran algunos comandos que pueden ser enviados a los lectores, basados en los Códigos de Comandos ISO 15693.

<b>Función del Comando</b>	<b>Código de Comando ISO</b>
<b>Inventario (identifica las tarjetas alrededor)</b>	<b>01hex</b>
<b>Silencio (no hacer nada)</b>	<b>02hex</b>
<b>Leer un solo bloque</b>	<b>20hex</b>
<b>Escribir un solo bloque</b>	<b>21hex</b>
<b>Bloquear un bloque</b>	<b>22hex</b>
<b>Leer múltiples bloques</b>	<b>23hex</b>
<b>Obtener Información del Sistema(configuración)</b>	<b>2Bhex</b>

Tabla 4.2.: Comandos del Lector[4].

El comando que se utilizó en esta ocasión fue el de inventario 01<sub>hex</sub>, ya que permite al lector obtener los identificadores de todos los tags dentro de su área de lectura. Este identificador es grabado en los tags desde su fabricación y no es posible modificarlo. Es conocido como el UID y es un código hexadecimal de 8 bytes, como se explicó anteriormente.

## Recepción y validación de paquetes por lector

El lector está escuchando, constantemente, los datos que se mueven por la red. El procesamiento de las tramas se da de la siguiente forma:

1. BEGIN\_RECEIVE: Descarta todos los caracteres recibidos hasta encontrar un SOF
2. SOF\_RECEIVED: Recibe los prox. 2 caracteres. Obtiene el tamaño del paquete.
3. LEN\_RECEIVED: Recibe (Longitud del paquete – 3 bytes) caracteres más.
4. PACKET\_VALIDATE: Verifica el LRC
5. DeviceID: Valida que el id del dispositivo sea el correcto.
6. VALIDATION\_COMPLETE: Envía el resto del paquete a la rutina de llamada.

## Petición de Lectura

Existen múltiples comandos que pueden ser enviados a los lectores, pero a continuación, se explica el comando de petición de lectura que se utilizó en este proyecto. Cada lector RFID, tiene un número serial único que le sirve para poder ser identificado en la red. En el caso que se muestra a continuación, se tiene un lector con un número serial:

Número de Serie del lector 1: 1001958

Con este número de serie se crea la siguiente trama, con datos hexadecimales:

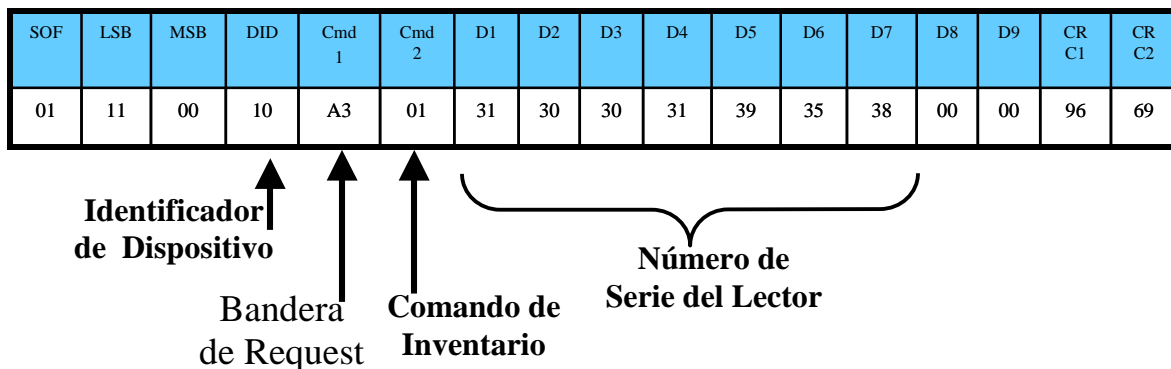


Figura 4.6.: Trama de petición de lectura.

El Comando de Inventario (01), significa que el lector debe obtener el UID de todos los *tags* de RFID dentro de su rango de lectura.

La *Bandera de Request* se crea de la siguiente forma:

Hexadecimal	A	3
Binario	1010	0011
Posición del bit	7654	3210



Donde dependiendo de la posición del bit y si esta en 0 ó 1, se envía cierta señalización al lector RFID.

Posición del Bit	Valor del Bit	Función
7	1	Define que el paquete es de tipo Request
6	0	Es el valor por default
5	1	Se requiere número serial del lector
4	0	UID del tag no requerido
3	0	Modulación FSK
2	0	100% modulación
1	1	Lectura rápida
0	1	Pulso 1/4

Si el bit 5 esta activado, el identificador serial se envia junto con el paquete request, esto permite que solo el lector con dicho UID conteste.

### Respuesta de Lectura

<b>SOF</b>	<b>LSB</b>	<b>MSB</b>	<b>DID</b>	<b>Cmd1 RPS</b>	<b>Cmd2 IR</b>	<b>SB</b>	<b>NT</b>	<b>T1</b>	<b>T1</b>	<b>T1</b>	<b>T1</b>	<b>T1</b>	<b>T1</b>	<b>T1</b>	<b>T1</b>	<b>T2</b>	
01	22	00	10	00	01	00	03	92	AE	81	06	00	00	07	E0	67	
<b>T2</b>	<b>T2</b>	<b>T2</b>	<b>T2</b>	<b>T2</b>	<b>T2</b>	<b>T2</b>	<b>T3</b>	<b>T3</b>	<b>T3</b>	<b>T3</b>	<b>T3</b>	<b>T3</b>	<b>T3</b>	<b>T3</b>	<b>T3</b>	<b>CRC 1</b>	<b>CRC 2</b>
5C	64	01	00	00	07	E0	EB	5E	64	01	00	00	07	E0	E3	1C	

Figura 4.7.: Trama de respuesta de lectura.

Donde los campos más significativos de la trama son los siguientes:

- IR** respuesta de inventario **01**, respuesta generada a un comando de inventario.
- RPS** respuesta de paquete exitosa **00**, la lectura no tuvo errores.
- SB** byte de estatus **00**, se encontraron *tags* dentro del rango de lectura.
- NT** número de tags **03**

En este ejemplo, el lector contestó con una identificación de 3 tags:

```

UID #1  92 AE 81 06 00 00 07 E0
UID #2  67 5C 64 01 00 00 07 E0
UID #3  EB 5E 64 01 00 00 07 E0
    
```

Esta respuesta es procesada en el host y se puede verificar mediante el byte SB, si la lectura fue exitosa. Posteriormente, con el byte NT es posible saber el número de tags que fueron identificados, y dependiendo de este valor, como un UID consta de 8 bytes, sabremos que, como se muestra en el ejemplo, si NT=3, entonces los próximos 24 bytes corresponden a los 3 UID leídos. Es posible procesar estos identificadores en el host y compararlos con valores existentes en la base de datos, para tomar una acción.

## **4.2. Tarjeta de Conversión RS232 - RS485**

Se desarrollo esta tarjeta de conversión para aprovechar las capacidades de RS-485 en el concepto de control de acceso, ya que permite tener dispositivos a distancias de 1200 metros. Permitirá extender la funcionalidad de este sistema, para realizar un control de acceso vehicular en un estacionamiento, controlando una barrera como actuador.

- **RS-232 (RTS, TX, RX)**

El puerto serial de las computadoras cumple con el estándar RS-232. La ventaja de este puerto es que muchas computadoras traen al menos uno, que permite la comunicación entre otros dispositivos, tales como otra computadora, el mouse, la impresora, microcontroladores, y otros dispositivos periféricos. La desventaja de este protocolo es que permite tener dispositivos, a sólo unos pocos metros de distancia.

- **RS-485 (RTS, TX, RX)**

RS-485, es definido como un sistema en bus de transmisión multipunto diferencial. Es ideal para transmitir a altas velocidades sobre largas distancias (35 Mbps hasta 10 metros y 100 Kbps en 1.200 metros) y a través de canales ruidosos, ya que reduce los ruidos que aparecen en los voltajes producidos en la línea de transmisión. El medio físico de transmisión es un par trenzado que admite hasta 32 estaciones en 1 sólo hilo, con una longitud máxima de 1.200 metros. La comunicación es half-duplex y soporta hasta 32 dispositivos. Al tratarse de un estándar, lo suficientemente abierto permite muchas y muy diferentes configuraciones y aplicaciones.

La tarjeta de conversión está formada por 2 componentes principalmente:

- MAX 232

Este circuito integrado contiene dos drivers (convierten de lógica TTL a voltajes RS232) y dos receptores (convierten de RS-232 a niveles de voltaje TTL) para apartar los niveles de voltaje de las señales de RS-232 hacia niveles de lógica TTL.

El MAX 232 necesita solamente una fuente de +5V para su operación. Internamente tiene un elevador de voltaje que convierte el voltaje de +5V al de doble polaridad de +12V y -12V.

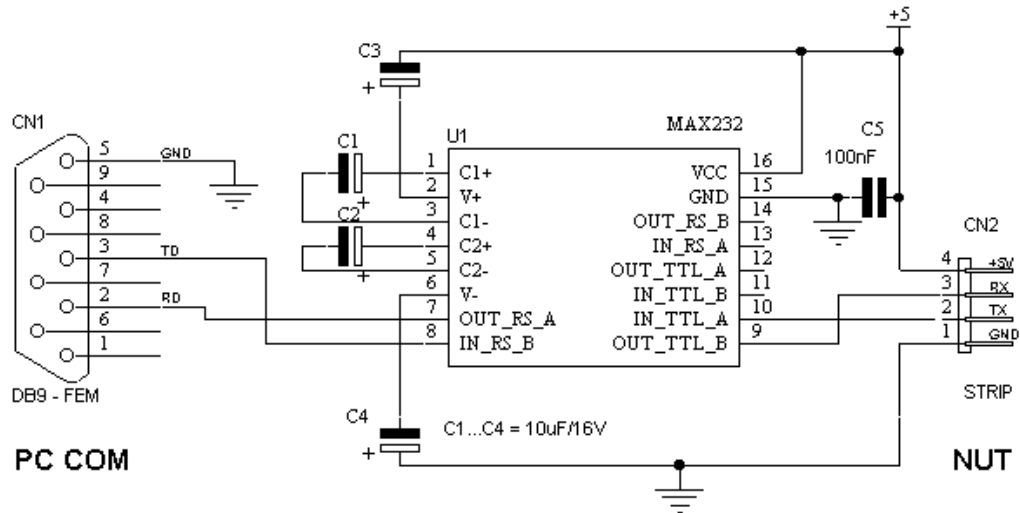


Figura 4.8.: Configuración MAX232[6].

- SN75176BP (Half Duplex)[28]

Este circuito integrado fue diseñado para la comunicación bidireccional de datos sobre líneas de transmisión de bus multipunto. Permite la interacción con una red RS-485 ya que cumple con el estándar TIA/EIA-485-A.

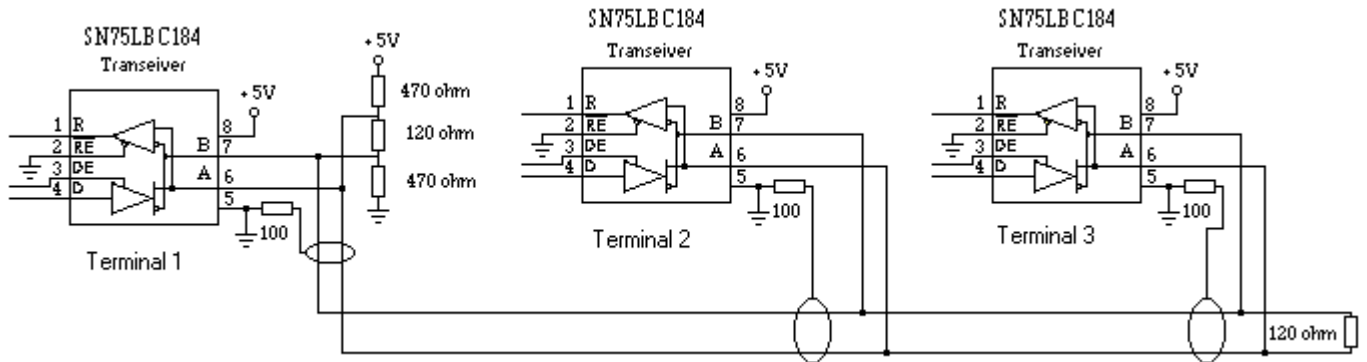


Figura 4.9.: Configuración de red RS485[6].

## Funcionamiento de la Tarjeta

La computadora emite comandos a través del puerto RS-232 en donde se transforman en voltajes RS-232. Este puerto está conectado directamente al puerto RS-232, quien recibe la señal del puerto serial y la transforma a niveles TTL, los cuales son enviados al circuito SN75176BP que manipula los niveles TTL dando como salida una señal RS-485. El proceso de recepción de datos, se da en la forma inversa.

### 4.3. Tarjeta Controladora

Uno de los objetivos de este sistema de control de acceso es manipular actuadores como: cerraduras eléctricas, magnéticas, barreras, torniquetes y motores. Para esto fue necesario desarrollar una tarjeta que tuviera la capacidad de escuchar y transmitir dentro de una red RS-485 y así poder controlar actuadores.

Sus principales características y funciones son las siguientes:

- Su unidad central es un microcontrolador de montaje superficial
- Tiene un módulo de Comunicaciones
- Tiene una módulo de Potencia
- Cuenta con un identificador único en la red
- Capaz de interactuar con sensores, actuadores y alarmas.
- Recibe comandos de una PC
- Envía Ack, como confirmación de que un comando se ejecutó correctamente.

Tiene los siguientes Módulos:

- Módulo de Comunicaciones
- Microcontrolador
- Módulo de Potencia

### 4.3.1 Módulo de Comunicaciones

El módulo de comunicaciones se compone básicamente de un circuito integrado SN75176BP descrito en la sección anterior, el cual permite la interacción con la red RS-485.

### 4.3.2 Microcontrolador

El elemento central de esta tarjeta es un microcontrolador con las siguientes características:

- Modelo MSP430F149 de Texas Instruments
- Componente de montaje superficial
- 60 KB Memoria de Programa
- Programación en lenguaje C
- USART
- Bajo consumo de energía

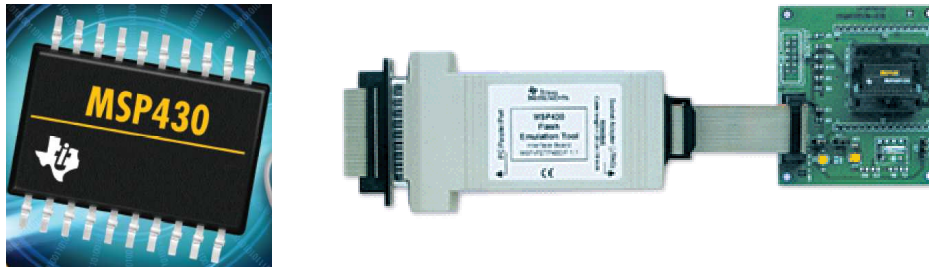


Figura 4.10.: Microcontrolador.

Este microcontrolador fue programado en lenguaje C, con ayuda de la herramienta IAR Workbench, con la cual es posible compilar, hacer debug y cargar el programa en el microcontrolador. El microcontrolador es colocado en una tarjeta de programación que se conecta a la PC por medio del puerto paralelo.

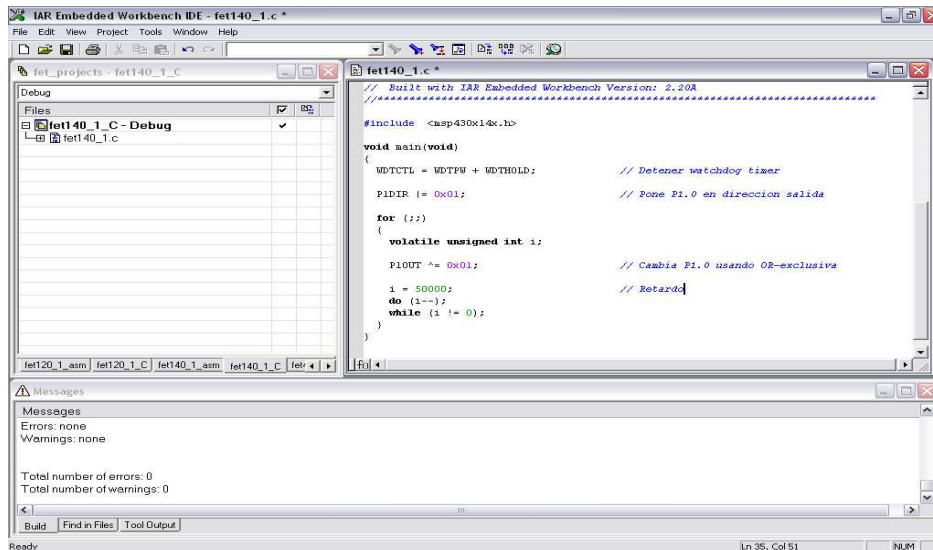


Figura 4.11.: IAR Workbench.

El programa que corre en el microcontrolador usa una UART que con la ayuda del módulo de comunicaciones, se encuentra constantemente escuchando en la red RS-485.

En cada microcontrolador se define un identificador único, a través del cual, es posible establecer la comunicación PC-Tarjeta Controladora. Así, el microcontrolador, al encontrar su identificador en la red, ejecuta un comando para liberar el actuador, que en nuestro caso, es una cerradura eléctrica, conectada al módulo de potencia por unos segundos. (Los segundos que el actuador permanece desactivado deben ser los suficientes para dar tiempo a la persona a que realice su acceso).

Este microcontrolador, monitorea constantemente un sensor que debe ser instalado en la puerta, cuya finalidad es controlar que ésta se encuentre cerrada, de lo contrario, se dispara una alarma auditiva.

Entonces cuando se manda la señal para liberar un actuador, unos segundos después, se valida que la puerta haya sido cerrada, de lo contrario se activará la alarma auditiva.

Una vez que se ejecuta la rutina de liberar el actuador, se envía una confirmación a la PC de que el comando se ejecutó satisfactoriamente.

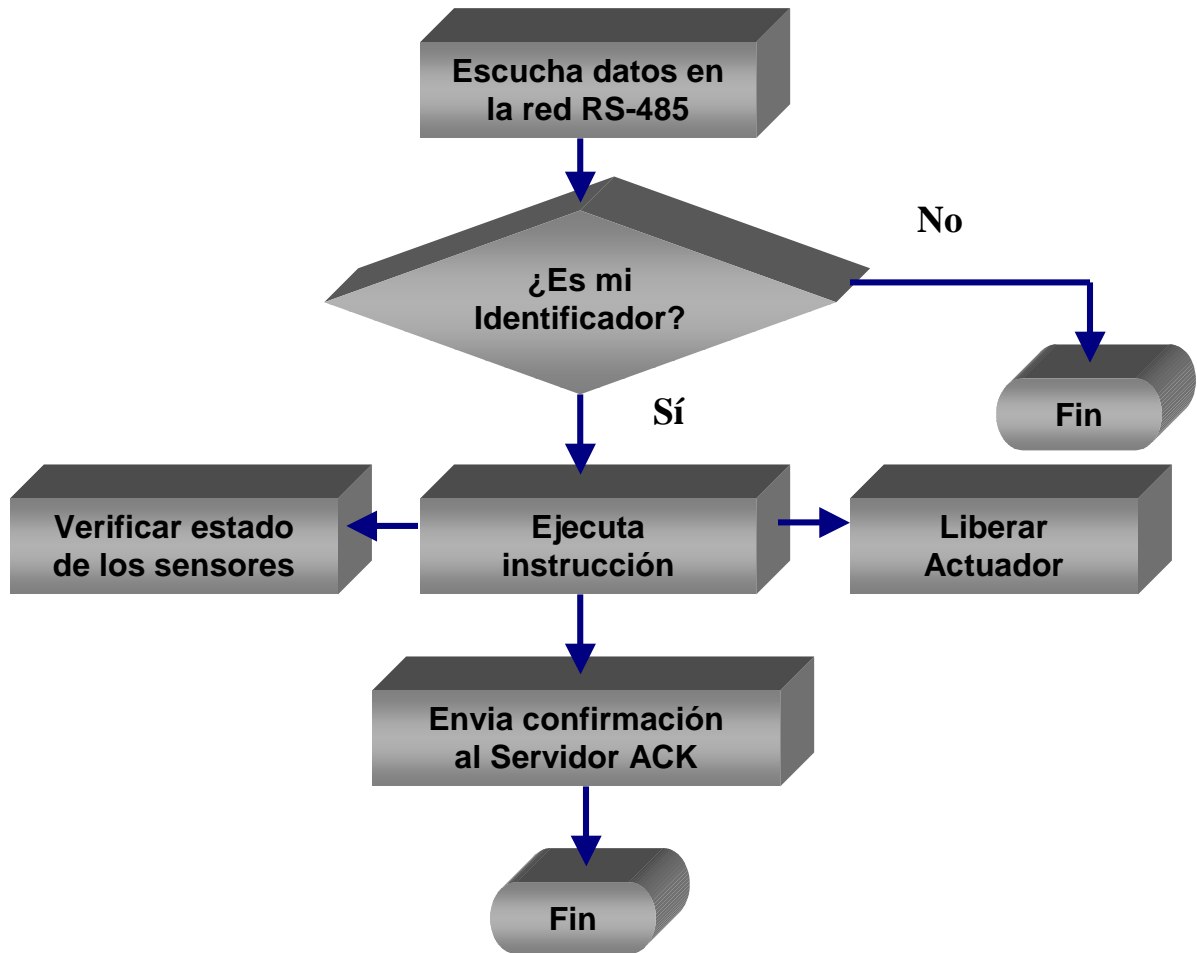


Figura 4.12.: Diagrama de flujo del microcontrolador.

### 4.3.3 Módulo de Potencia

Este módulo consiste en un circuito de corriente alterna, en donde se encuentra conectado el actuador. Este circuito está conectado hacia una toma de corriente y el componente, que impide que el actuador este en todo momento activado, es un TRIAC que sirve como interruptor, que al recibir un pulso del microcontrolador, cierra el circuito, y permite la activación del actuador.

#### TRIAC

Un TRIAC o Triodo para Corriente Alterna es un dispositivo semiconductor, de la familia de los dispositivos de control por tiristores. La diferencia con un tiristor convencional es que éste es unidireccional y el TRIAC es bidireccional. Podría decirse que el TRIAC es un interruptor capaz de conmutar la corriente alterna.

Posee tres electrodos: A1, A2 (en este caso pierden la denominación de ánodo y cátodo) y puerta. El disparo del TRIAC se realiza aplicando una corriente al electrodo puerta.

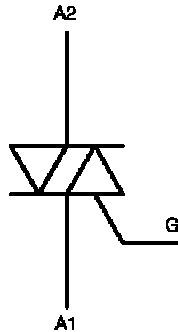


Figura 4.13.: Diagrama de TRIAC.

Cuando el microcontrolador envía una señal para liberar una cerradura eléctrica, esta señal es enviada a un transistor 2n2222A NPN que funciona como un switch y energice al optoacoplador moc3011.

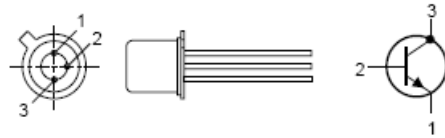


Figura 4.14.: Diagrama de Transistor.

El optoacoplador es un dispositivo que integra a un diodo LED y a un fototransistor, cuando el circuito se energiza, el LED emite luz que ilumina al fototransistor lo que cierra el circuito entre las patas 6 y 4 del moc3011 que es justo en donde se encuentra conectada la puerta del TRIAC, entonces se tiene el disparo necesario para cerrar el circuito de corriente alterna con ayuda del TRIAC.

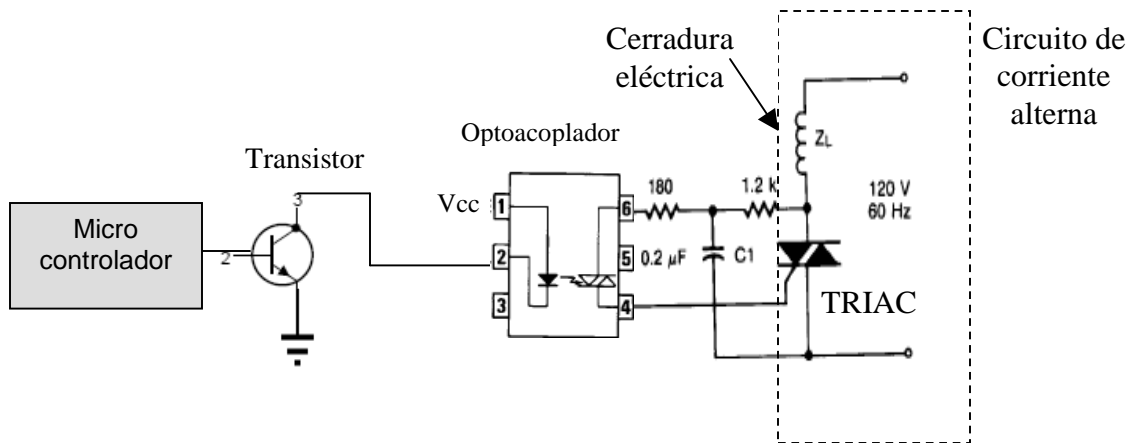


Figura 4.15.: Diagrama del Módulo de Potencia.



## 4.4. Tarjetas impresas

La tarjeta de conversión RS232-RS485 y la tarjeta controladora fueron desarrolladas y probadas en un ambiente de laboratorio, con ayuda de tarjetas protoboard. Lo ideal hubiera sido construir las tarjetas impresas de estos circuitos, lamentablemente no fue posible. Lo que si se realizó fueron los diseños esquemáticos y PCB en protel, que permite que el paso hacia las tarjetas impresas sea mucho más sencillo. Para esto es necesario crear un proyecto en protel y crear un modelo esquemático acerca de cómo están conectados los distintos componentes de circuito, así es posible validar que el modelo esté correcto.

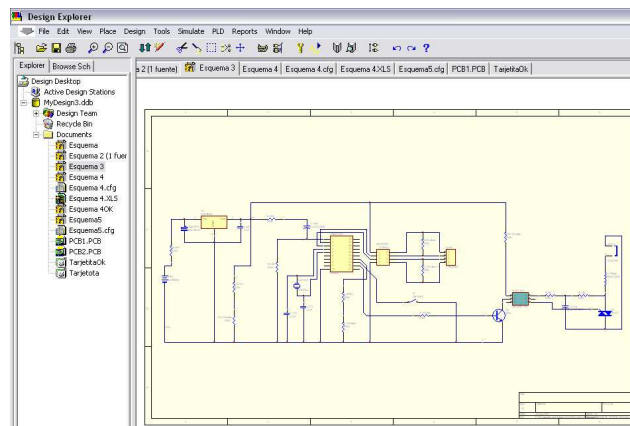


Figura 4.16.: Diagrama esquemático.

Posteriormente, se transforma este esquema hacia un modelo PCB, para lo cual es necesario especificar el footprint de cada elemento (configuración física del elemento, número de patas, tamaño, entre otras). Se definen las dimensiones de la tarjeta y se corre una simulación, para crear las pistas que conectan a todos los elementos.

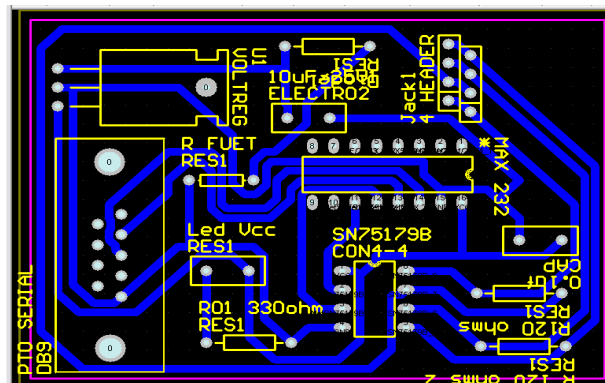


Figura 4.17.: Diagrama PCB.

## Esquemático de la tarjeta de conversión RS232-RS485

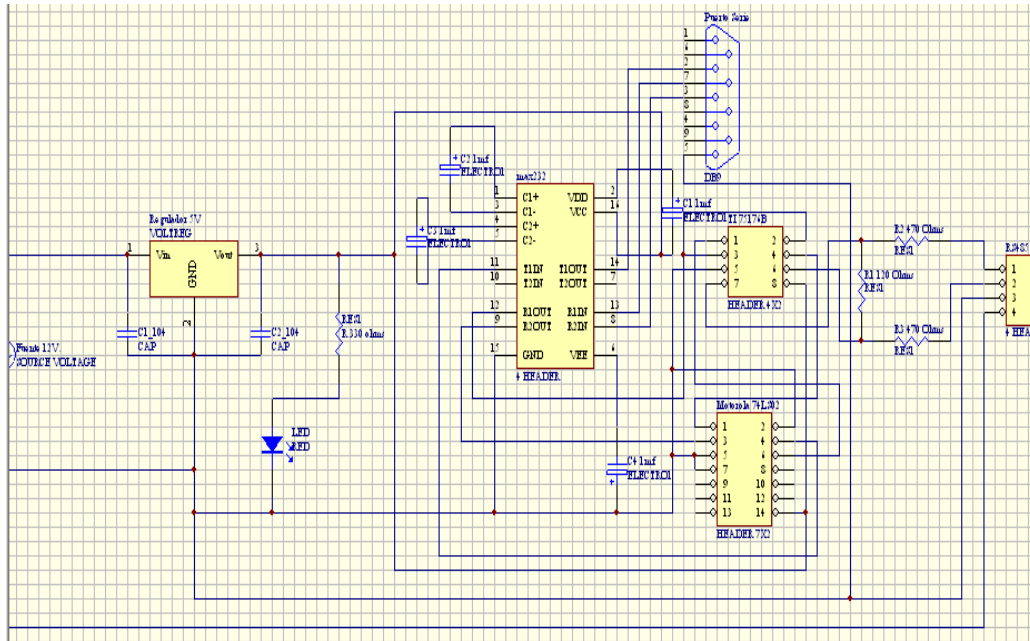


Figura 4.18.: Esquemático tarjeta de conversión RS232-RS485.

## PCB RS232-RS485

El modelo PCB desarrollado utilizó 2 capas (azul cara superior y rojo cara inferior), existen pistas por los dos lados de la placa.

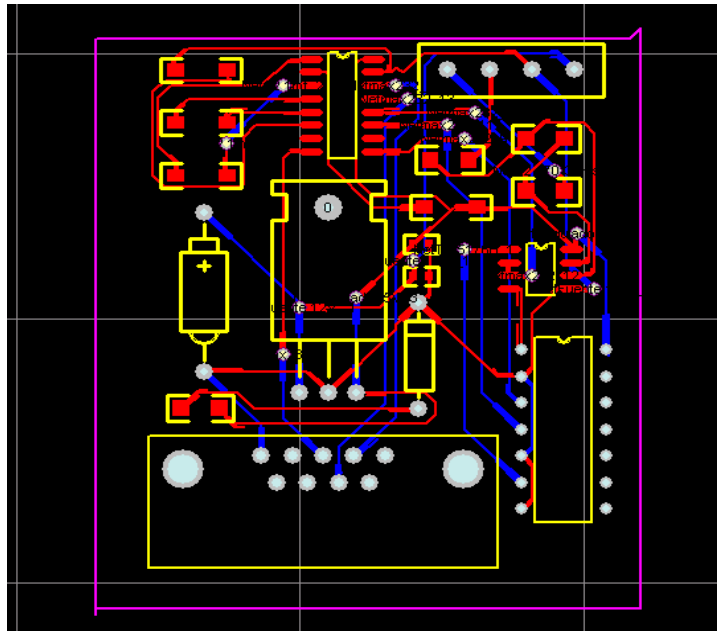


Figura 4.19.: PCB de la tarjeta de conversión RS232-RS485.

## Simulación 3D Tarjeta RS232-RS485

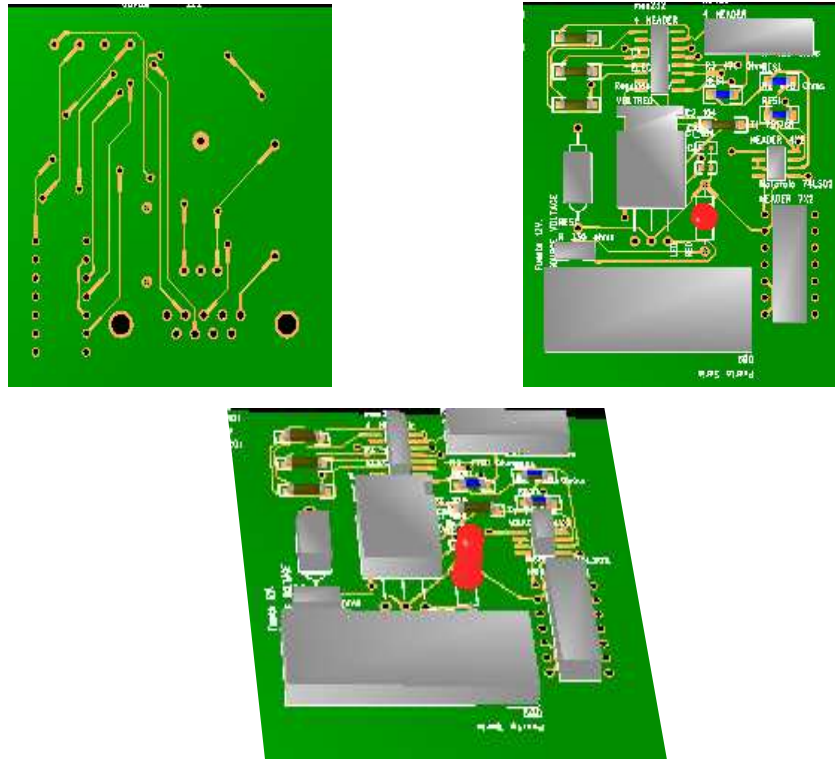


Figura 4.20.: Modelo 3D de la tarjeta de conversión RS232-RS485.

## Esquemático de Tarjeta Controladora

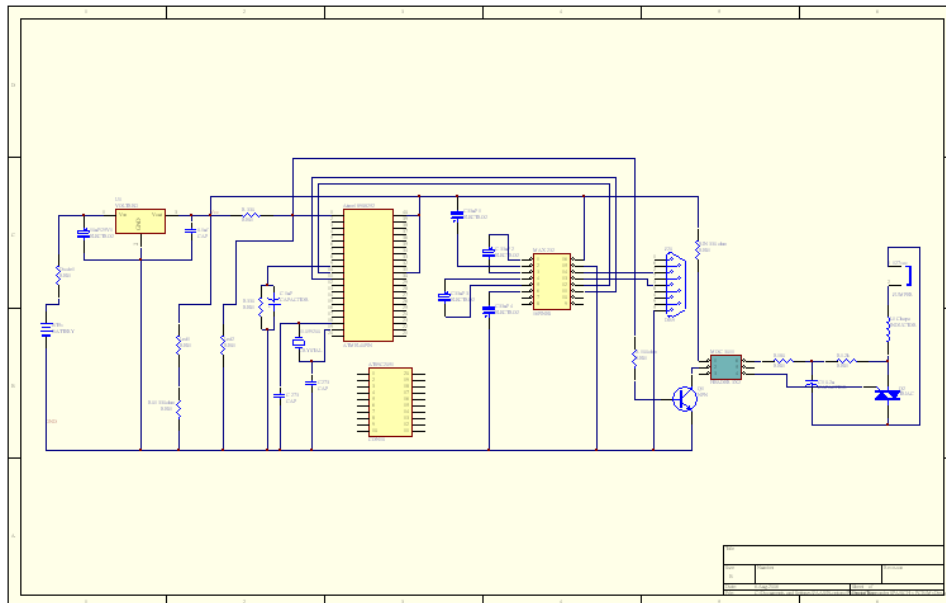


Figura 4.21.: Esquemático tarjeta de controladora.

# Capítulo 5. Diseño de Software

En este capítulo se estructuran las aplicaciones de software desarrolladas para la aplicación de acceso.

Inicialmente se plantean aplicaciones cliente servidor, que se ejecutan en la computadora host que está conectada a la red RS-485. Estas aplicaciones son responsables de la operación día a día de la aplicación. La arquitectura de desarrollo utilizada fue cliente servidor, por cuestiones de rendimiento, seguridad y disponibilidad.

Los requerimientos de estos programas, no justificaban el desarrollo de estos módulos en ambientes web, ya que esto no es diferenciador para los usuarios del sistema.

Usuarios del sistema:

- *Usuario administrador*, para configurar el sistema,
- *Usuarios operativos*, quienes dan de alta usuarios y realizan el registro de visitantes.
- *Usuarios finales*, quienes poseen una tarjeta RFID y realizan accesos.

Los usuarios operarios deben estar físicamente en el acceso principalmente y frente a la máquina host, por lo cual no es necesario plantear un sistema distribuido en varias capas. La información que genera el sistema en cuanto a ausencias y llegadas fuera de tiempo, es compartida con la gente administrativa a través de la generación de reportes.

Finalmente se desarrolló un aplicación web, orientada a usuarios administrativos, en la cual es posible visualizar los niveles de ausentismo y retardos por día, mes y año y entre departamentos. Adicionalmente es posible obtener información detallada de los accesos del día.

La justificación para realizar esta aplicación con una arquitectura web es la necesidad de soportar usuarios remotos, quienes deben tener acceso a esta información en todo momento.

En la Figura 5.1 se muestra la arquitectura general del sistema.

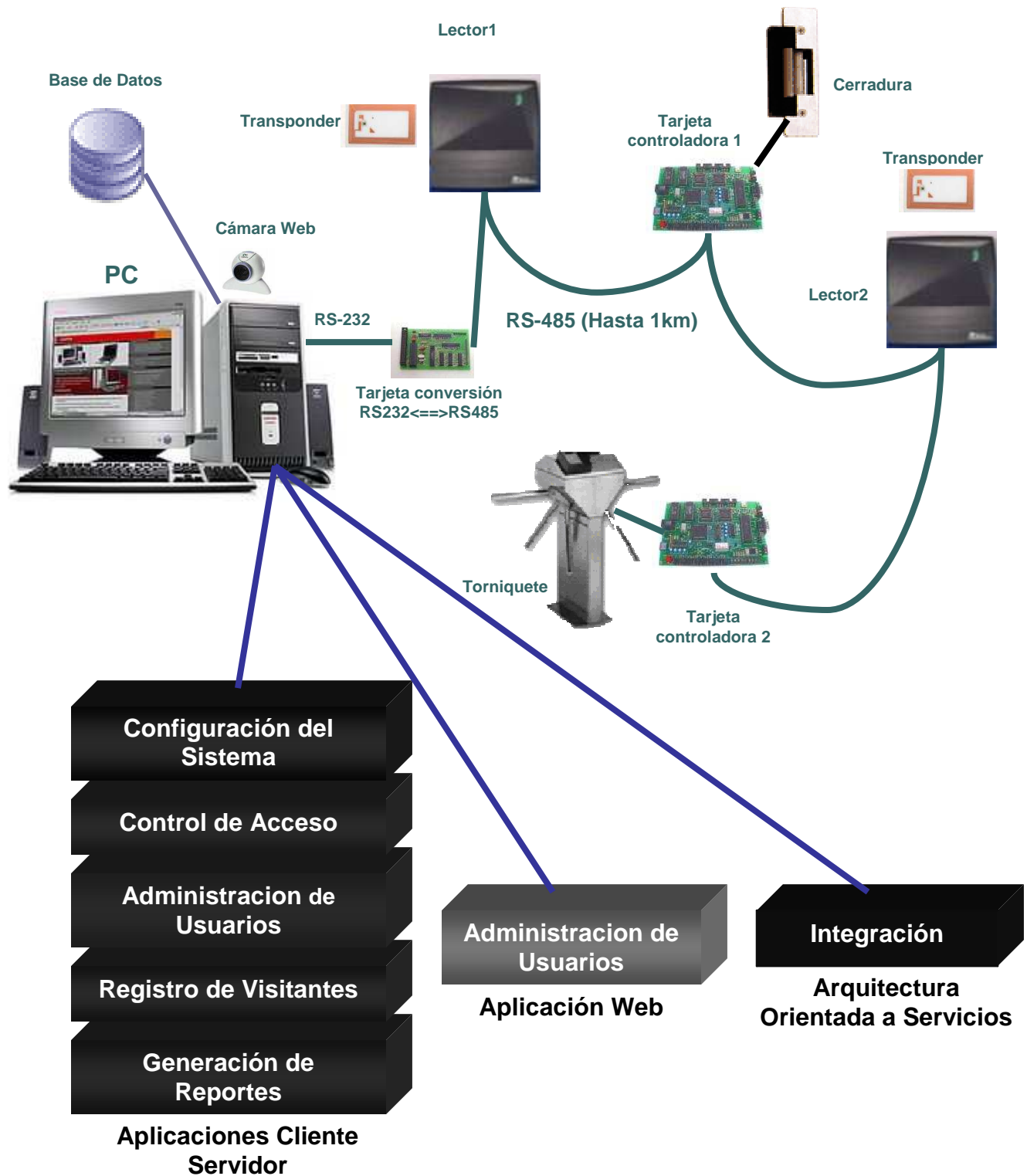


Figura 5.1.: Diagrama general del sistema de control de acceso.

## 5.1. Esquema de Base de Datos

La base de datos utilizada, inicialmente, fue desarrollada en MySQL, y posteriormente, fue migrada a SQL Server y a Oracle de modo que las aplicaciones fueran compatibles con los manejadores más populares actualmente. Esto se hizo con dos propósitos: hacer benchmarks para comparar el desempeño de los distintos manejadores (el benchmark no fue realizado) y tener un sistema capaz de correr con distintas manejadores de modo que pudiera implementarse en casos reales sin que el manejador represente un problema ya que la mayoría de las empresas ya usan alguno de estos 3 manejadores.

El esquema desarrollado está formado por 8 tablas:

- *Administrador.*

Contiene las credenciales del usuario administrador del sistema, para que pueda tener acceso a todos los módulos del sistema como superusuario.

- *Usuarios*

Almacena la información general de todos los usuarios del sistema.

- *Usuario\_Acceso*

Define información más detallada en cuanto al acceso del usuario como la fecha en que expira su acceso y su hora de entrada y salida, esta última es usada posteriormente para definir retardos.

- *Registros*

En esta tabla se almacenan todos los accesos realizados por usuarios y visitantes. Define el punto en donde se realizó el acceso, si entró o salió, el identificador de la tarjeta del usuario, fecha, hora, y si fue el primer acceso del día.

- *Tipo\_Acceso*

Aquí se especifican explícitamente los puntos para cada usuario: el acceso es “Aceptado” o “Denegado”

- *Nombre\_Acceso*

Esta tabla contiene todos los puntos en donde se tiene un dispositivo de RFID controlando el acceso.

- *Visitantes*

## Información general de los visitantes

- *Visitantes\_Acceso*

Información específica de cada visita de una persona. Aquí se define el identificador de la tarjeta que utilizará el visitante.

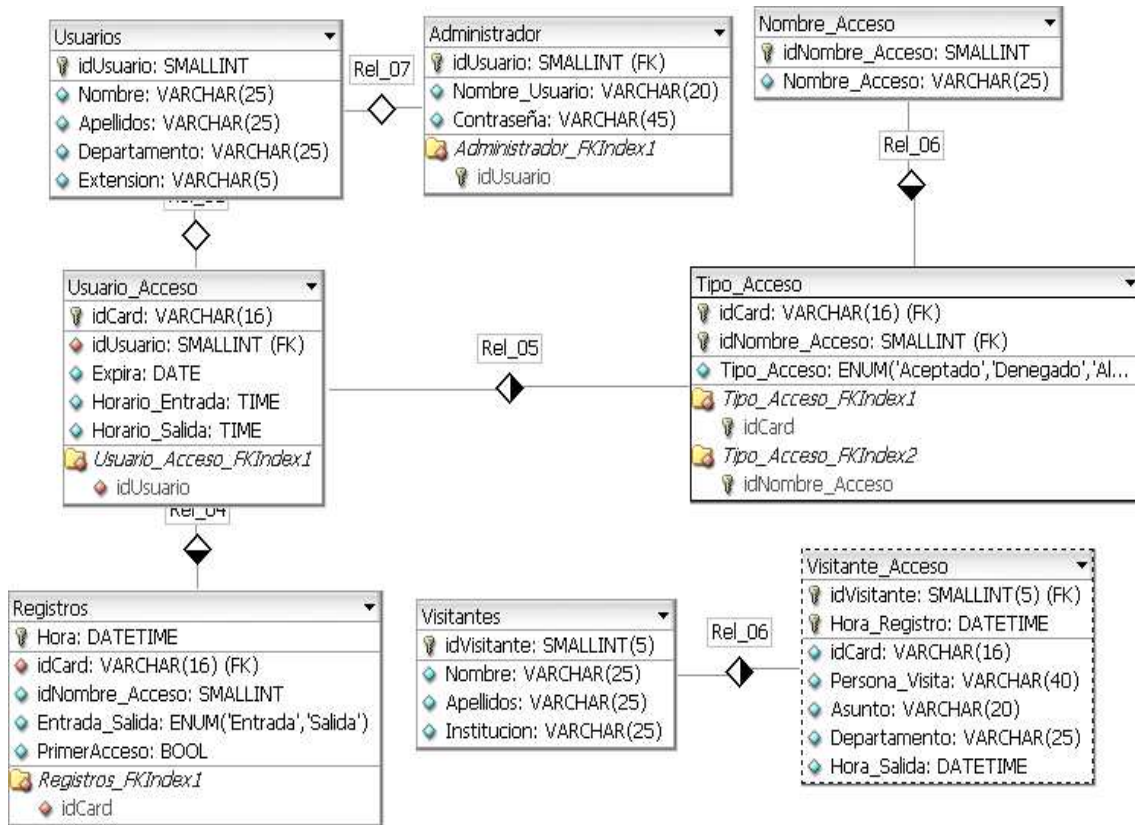


Figura 5.2.: Esquema de Base de Datos.

El diagrama mostrado en la Figura 5.2, fue realizado con ayuda de la herramienta dbdesigner de fabForce, la cual es gratuita.

Para poder interactuar con el manejador de base de datos Oracle, desde java, es necesario hacer uso de la librería oracle.jdbc.

## 5.2. Módulo de Configuración del Sistema desarrollado

Este módulo permite realizar la configuración del sistema de control de acceso. Incluye agregar, eliminar o modificar los distintos puntos de acceso que se controlarán con el sistema, así como el número, tipo e identificador de los dispositivos en dicho acceso. En un punto de acceso se pueden tener uno o dos lectores (dependiendo del caso, es posible controlar un punto de acceso con un solo lector de RFID) y un actuador. La ventaja de este módulo de configuración es que genera un archivo, el cual, es utilizado por el módulo de control de acceso cada vez que es inicializado. Lo anterior permite que el sistema se reinicialice sin perder la configuración actual. Esta configuración incluye también la introducción del identificador de cada dispositivo en la red, de modo que el módulo de control de acceso pueda comunicarse con todos los dispositivos en la misma.

Así mismo una vez creada la configuración del sistema, esta aplicación puede utilizarse para modificarla.

### Descripción general de la aplicación

Módulo para la configuración de lectores y actuadores del sistema mediante la utilización de serialización.

### Serialización de objetos en Java [40]

La serialización de un objeto consiste en obtener una secuencia de bytes que represente el estado de dicho objeto. Esta secuencia puede utilizarse de varias maneras: puede enviarse a través de la red, guardarse en un fichero para su uso posterior o utilizarse para recomponer el objeto original.

El estado de un objeto viene dado, por el estado de sus campos. Así, serializar un objeto consiste, básicamente, en guardar el estado de sus campos. Si el objeto a serializar tiene campos, que a su vez son objetos, habrá que serializarlos primero. Éste es un proceso recursivo que implica la serialización de todo un grafo (en realidad, un árbol) de objetos.

Además, también se almacena información relativa a dicho árbol, para poder llevar a cabo la reconstrucción del objeto serializado.

Un objeto serializable es un objeto que se puede convertir en una secuencia de bytes. Para que un objeto sea serializable, debe implementar la interfaz *java.io.Serializable*. Esta interfaz no define ningún método. Simplemente se usa para marcar aquellas clases cuyas instancias pueden ser convertidas a secuencias de bytes (y posteriormente reconstruidas). Objetos tan comunes como *String*, *Vector* o *ArrayList* implementan la interfaz *Serializable*, de modo que pueden ser serializados y reconstruidos más tarde.

Para serializar un objeto no hay más que declarar el objeto como serializable:



```
public class MiClase implements java.io.Serializable
```

El sistema de ejecución de Java se encarga de hacer la serialización de forma automática.

## Descripción específica de la aplicación

Al iniciar esta aplicación, se valida la existencia del archivo de configuración, si es la primera vez que se ejecuta la aplicación (o si el archivo fue borrado), el archivo se creará con el nombre de “ControlRFID.crfid” que se sitúa en la carpeta donde esté la aplicación.

Una vez validado el archivo se obtiene la siguiente pantalla:



Figura 5.3.: Módulo de configuración.

En esta pantalla se muestran cuatro botones con las acciones que se pueden realizar y un panel en donde se enlistarán los nombres de los nodos que sean creados.

A continuación se describe cada una de las operaciones que pueden ser realizadas en esta aplicación:

### Agregar Nodo

En esta opción se podrán seleccionar entre dos tipos de configuraciones que son:

- Agregar 1 Lector y 1 Actuador
- Agregar 2 Lectores y 1 Actuador

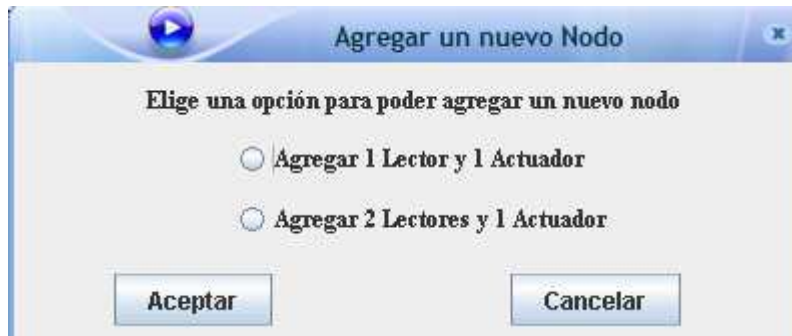


Figura 5.4.: Agregar un nodo.

Dependiendo de la opción que se elija se mandará llamar a dos clases distintas LectorC1 o LectorC2, por medio de su constructor por defecto.

A la hora de ingresar a cualquiera de las dos clases anteriores se declara un

```
LinkedList<Nodo> linkedListNodo=new LinkedList();
```

Este último es una lista ligada que va a contener a todos los nodos que creamos con el objetivo de serializarlo solo en un archivo ya mencionado.



Figura 5.5.: Configurar nodo.

Esta es la pantalla para agregar un actuador y un lector. Para poder guardar un nodo se deben de llenar todos los campos, de tal forma que los campos de código de lector y actuador sea un número hexadecimal (00 - FF).Cada vez que agrega un campo de código, se hace una validación para que sea un número hexadecimal, de ahí hacemos un cast a byte mediante la clase de conversión.

Una vez que los campos han sido llenados completamente, se pueden almacenar oprimiendo el botón de guardar, el cual realiza las acciones que se muestran en el siguiente diagrama de flujo.

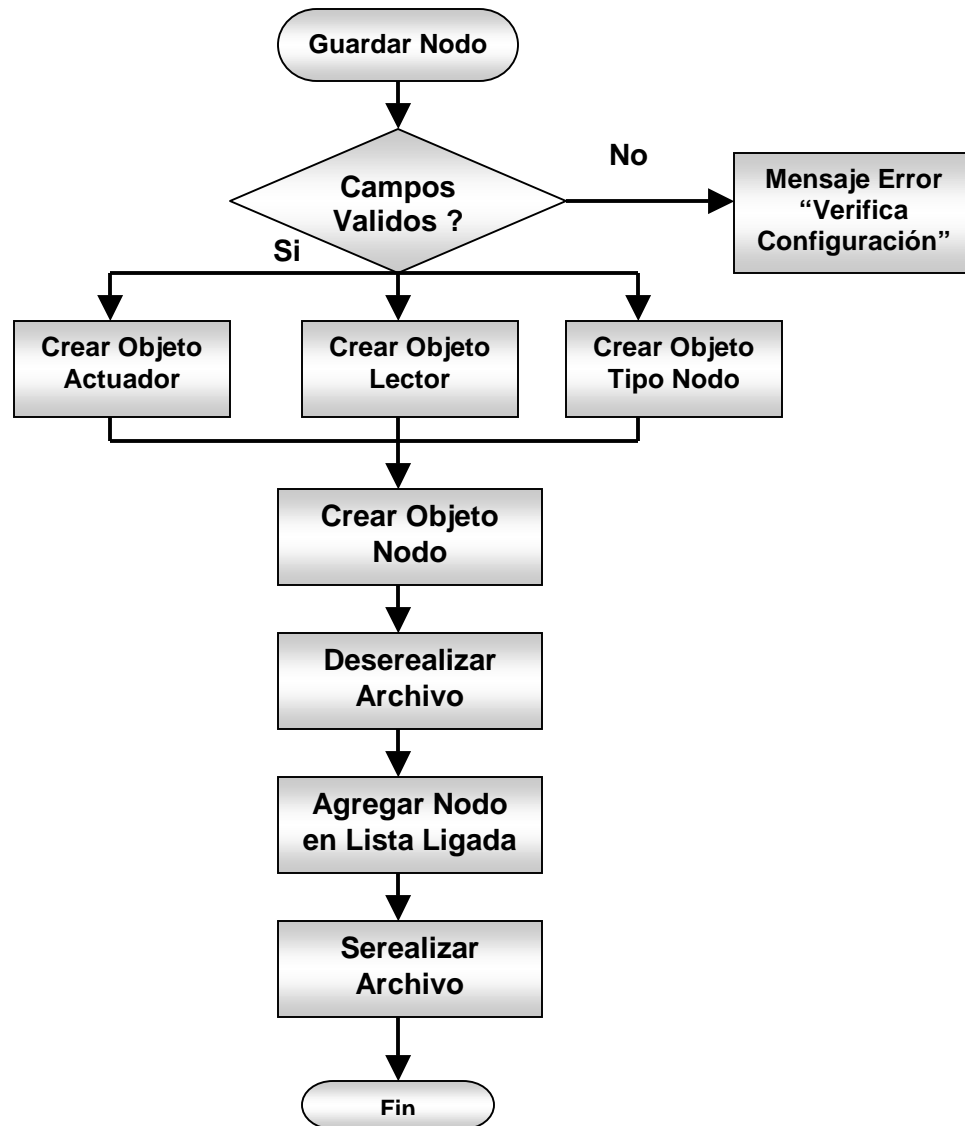


Figura 5.6.: Diagrama de flujo *agregar nodo*.

## Abrir Nodo

Para abrir un nodo se selecciona aquel que se desea editar. Para llevar a cabo esta acción, se deserializa el archivo para verificar el tipo de nodo que tiene con la finalidad de abrir la ventana correcta, correspondiente a las clases LectorC1 y LectorC2.



Figura 5.7.: Ventana abrir nodo.

Para editar un nodo, se hace click en el botón “editar nodo” para que habilite todos los campos. Una vez editados se oprime guardar y se realizan las mismas acciones que cuando se guarda, con la diferencia de que se reemplaza el nodo anterior por el nuevo.

## ELIMINAR NODO

Para eliminar un nodo se selecciona uno y haciendo clic en el botón “eliminar nodo” se obtiene un mensaje de advertencia que pedirá confirmar la acción de eliminar el nodo, ésto se hace mediante la utilización de un JDialog.

Para realizar esta acción de eliminar se deserializa el archivo y se elimina el nodo del linkedlist. Una vez hecho eso serializamos de nuevo el archivo.

La ventaja de la serialización es que permite la autoconfiguración del sistema, a partir de un archivo. Si la aplicación de control de acceso es detenida, esta recuperara su configuración original al iniciar de nuevo, ya que leera el archivo serializado para cargar su configuración.

## 5.3. Módulo de Control de Acceso

Este módulo hace las funciones de middleware de RFID para nuestro sistema. Es el encargado de enviar peticiones de lectura hacia los lectores y procesar las respuestas que se obtienen de éstos. Este módulo se autoconfigura, leyendo el archivo que se crea, a partir del módulo de configuración.

La interacción con la red RS-485, en donde se encuentran todos los dispositivos, se hace a partir del puerto RS-232, el cual se encuentra conectado hacia la tarjeta de conversión RS-232 a RS-485. El control del puerto RS-232 se realiza con ayuda de la API Commapi de java[13].

Este módulo realiza las siguientes funciones:

### Aplicación Control de Acceso

- Autoconfiguración a partir del archivo ControlRdfi.crfid
- Poleo de dispositivoLectores
- Verificar tipo de acceso de los usuarios (Aceptado-Denegado)
- Realizar operaciones con la Base de Datos
- Enviar comandos para liberar actuadores
- Verificar funcionamiento de los dispositivos

### Filtros y Reglas

- Lecturas Repetidas
- Lecturas Erróneas
- Obtener identificadores

### Funciones del Driver javax.comm

El API de comunicaciones de java, constituido por el paquete javax.comm, no forma parte del JDK, pero agrega soporte a Java para el manejo de dispositivos a través de los puertos serie y paralelo.

- Captura de paquetes
- Envío de paquetes

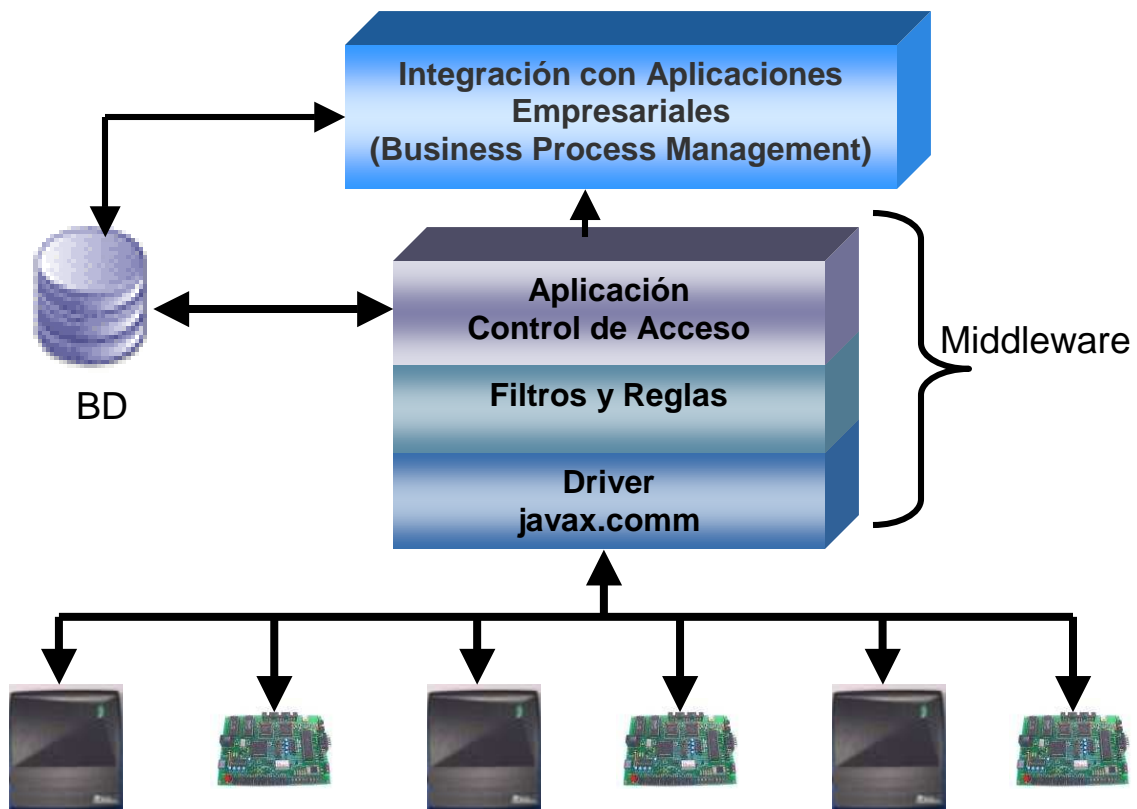


Figura 5.8.: Diagrama de la capa de Middleware.

Tareas específicas del módulo de Control de Acceso:

## 1. Inicializar Componentes

- Abrir archivo de configuración

Se abre el archivo de configuración, y se vacía todo su contenido en una lista ligada, para posteriormente, recorrerla y hacer el pooling entre todos los lectores.

- Inicializar Puerto Serial COM1

Se inicializa el puerto COM1 y se configura para trabajar a 9600 bps con un bit de parada y sin paridad.

- Se abre una conexión a la base de datos

Dependiendo del manejador de base de datos que se esté utilizando es posible abrir esta conexión hacia MySQL, Oracle y Microsoft SQL Server.

## 2. Pooling de Lectores

Una vez que se tiene la lista ligada con el contenido del archivo de configuración, el contenido real de esta lista corresponde a objetos de tipo Nodo. Un objeto de Tipo Nodo contiene Objetos de Tipo Lector, Objetos de Tipo Actuador, y un String con el nombre del nodo. A su vez un objeto Lector contiene un String con su ubicación y otro String con el comando necesario para que este efectúe una lectura

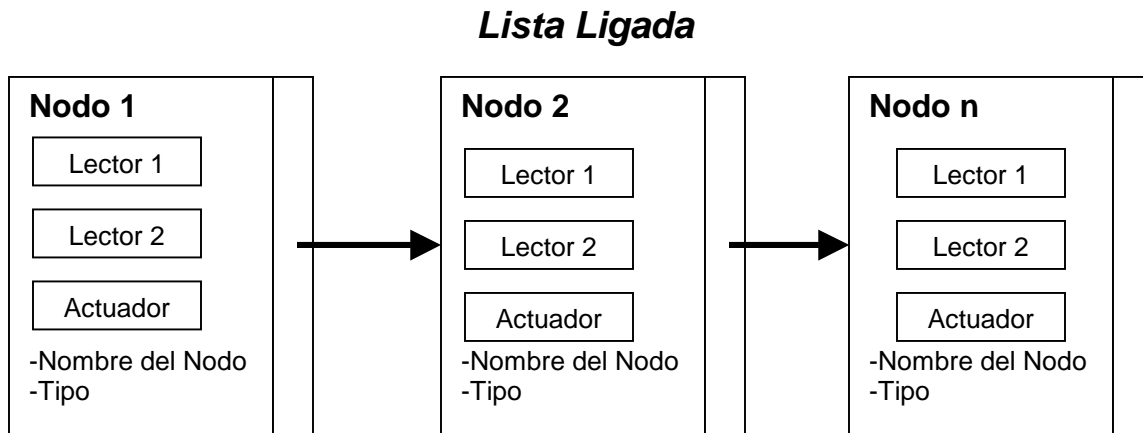


Figura 5.9.: Diagrama de Lista Ligada.

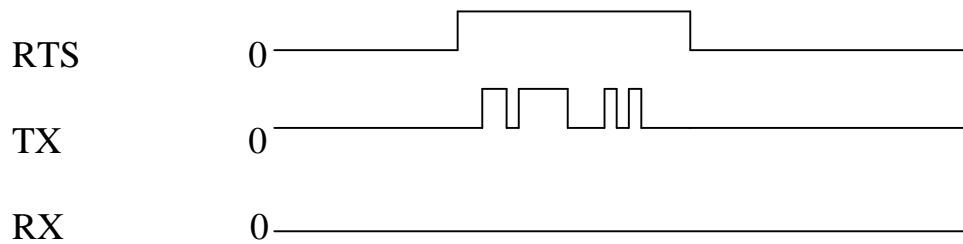
Entonces existe un método que recorre esta lista ligada, obtiene los objetos lector y sus comandos y les envía una señal de lectura. Si el lector contesta, con algún identificador, éste se procesa para tomar una acción específica, de lo contrario, se busca el siguiente lector en la lista, para enviarle el comando de lectura. Este proceso no se detiene.

La mayor complicación en este punto fue que la interacción con el puerto serial debe desarrollarse de la siguiente manera. El puerto RX siempre está escuchando y para esto el bit RTS (Ready To Send) debe estar en 0. A la hora de enviar un comando a un lector o un actuador, se debe poner el bit RTS en 1. El problema que se tuvo en este punto fue que cada vez que se envía un comando, se debe de hacer un RTS=true, durante el tiempo que dure el envío del comando y al finalizar este envío se debe hacer un RTS=false de modo que se pueda recibir correctamente la respuesta de los lectores o actuadores. Este proceso debe ser muy preciso, ya que si tarda un poco más de lo necesario en hacer el RTS=false, puede ser que en ese momento reciba la contestación de un lector, y se pierda parte de la información. El problema fue que en un principio se tomó por hecho que el control del bit RTS lo hacía explícitamente el controlador del puerto serial, lo cual no es así.

Este proceso causó un poco de confusión al principio, porque era posible emitir comandos, pero no se obtenía ninguna respuesta(RTS estaba por default en 1). Hasta que se hizo uso de un osciloscopio fue posible apreciar este fenómeno.

El tema de la precisión de la duración en que RTS debe estar en 0 ó 1 se resolvió analizando el tiempo que le tomaba a un comando enviarse por completo. Este tiempo se utilizó para crear rutinas *sleep*, las cuales se ejecutan simultáneamente al envío de un comando. Al finalizar el tiempo de las rutinas *sleep* restauran el valor de RTS haciendo un  $RTS=false$ . El comando que se envía a un lector es más grande que el que se envía a un actuador, por lo tanto, es necesario llamar a dos rutinas diferentes *sleep* con tiempos diferentes.

### Transmisión



### Recepción

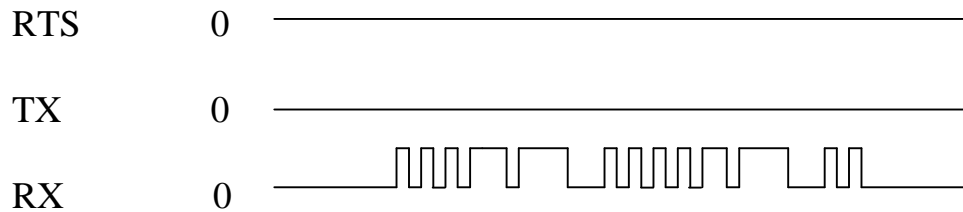


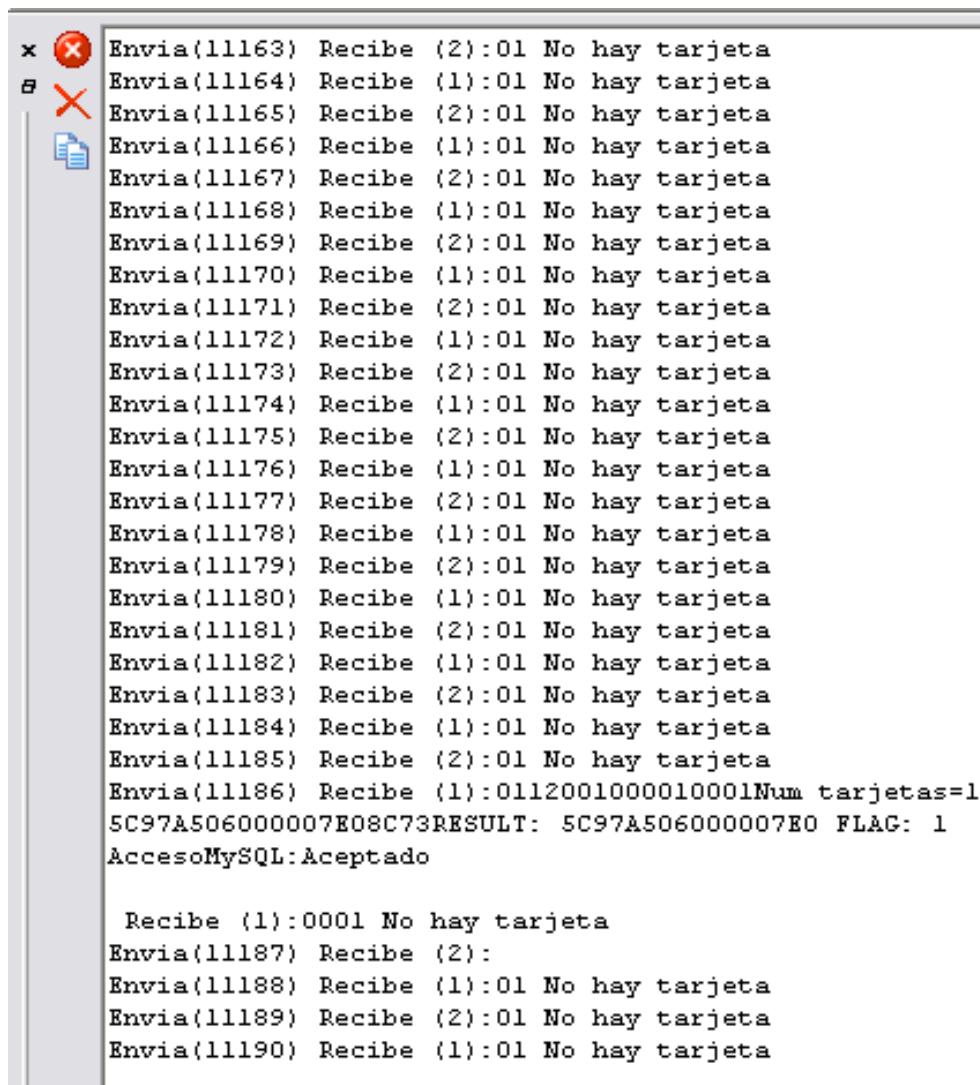
Figura 5.10.: Transmisión de datos RS-232.

Se hace uso de una tercera rutina de temporización, para dar una pausa entre el envío del comando entre lectores. Sin el uso de este temporizador, la computadora procesa la información demasiado rápido, y satura a los lectores con peticiones de lecturas, lo que ocasiona que éstos contesten con datos erróneos.



### 3. Recepción de Respuestas

Después de enviar un comando de lectura a un lector, éste contestará, independientemente de si realizó una lectura positiva o no. De este modo, es posible saber si un lector deja de funcionar, ya que no obtendremos ninguna respuesta de él. La función de este componente es estar escuchando en RX y esperar las respuestas de los lectores. Aquí se hace un filtrado de errores y de lecturas duplicadas (Si la tarjeta de RFID no es retirada rápidamente del rango del lector, es posible obtener múltiples lecturas de un mismo tag, las cuales hay que filtrar), y una vez que se obtiene una lectura positiva, el lector envía al host el identificador de la tarjeta leída. En el host, el módulo de control de acceso analiza el identificador recibido, verifica que sea auténtico y manda llamar al proceso que verifica que el acceso solicitado, sea válido. Esto es que el usuario exista, y que tenga permitido el acceso que esta solicitando.



```
x [X] Envia(11163) Recibe (2):01 No hay tarjeta
B [X] Envia(11164) Recibe (1):01 No hay tarjeta
  [X] Envia(11165) Recibe (2):01 No hay tarjeta
  [X] Envia(11166) Recibe (1):01 No hay tarjeta
  [X] Envia(11167) Recibe (2):01 No hay tarjeta
  [X] Envia(11168) Recibe (1):01 No hay tarjeta
  [X] Envia(11169) Recibe (2):01 No hay tarjeta
  [X] Envia(11170) Recibe (1):01 No hay tarjeta
  [X] Envia(11171) Recibe (2):01 No hay tarjeta
  [X] Envia(11172) Recibe (1):01 No hay tarjeta
  [X] Envia(11173) Recibe (2):01 No hay tarjeta
  [X] Envia(11174) Recibe (1):01 No hay tarjeta
  [X] Envia(11175) Recibe (2):01 No hay tarjeta
  [X] Envia(11176) Recibe (1):01 No hay tarjeta
  [X] Envia(11177) Recibe (2):01 No hay tarjeta
  [X] Envia(11178) Recibe (1):01 No hay tarjeta
  [X] Envia(11179) Recibe (2):01 No hay tarjeta
  [X] Envia(11180) Recibe (1):01 No hay tarjeta
  [X] Envia(11181) Recibe (2):01 No hay tarjeta
  [X] Envia(11182) Recibe (1):01 No hay tarjeta
  [X] Envia(11183) Recibe (2):01 No hay tarjeta
  [X] Envia(11184) Recibe (1):01 No hay tarjeta
  [X] Envia(11185) Recibe (2):01 No hay tarjeta
  [X] Envia(11186) Recibe (1):0112001000010001Num tarjetas=1
  [X] 5C97A506000007E08C73RESULT: 5C97A506000007E0 FLAG: 1
  [X] AccesoMySQL: Aceptado

  [X] Recibe (1):0001 No hay tarjeta
  [X] Envia(11187) Recibe (2):
  [X] Envia(11188) Recibe (1):01 No hay tarjeta
  [X] Envia(11189) Recibe (2):01 No hay tarjeta
  [X] Envia(11190) Recibe (1):01 No hay tarjeta
```

Figura 5.11.: Recepción de repuestas de lectura.

## 4. Consulta a la Base de Datos

Esta parte del proceso, recibe el identificador obtenido y el acceso al que se quiere ingresar. Se realizan las siguientes acciones:

### a. Verificar el tipo de acceso

Lo primero que se hace es verificar si el usuario en cuestión tiene un tipo de acceso “Aceptado” en el punto donde quiere ingresar. Para esto, se hace una consulta en la base de datos, y si el acceso es aceptado, se manda llamar a la función que libera a los actuadores.

### b. Verificar primer acceso

Si se da un acceso positivo, se hace otra consulta para verificar si se trata del primer acceso del día y de ser así se ejecutan las acciones descritas en el siguiente paso (c).

De no tratarse del primer acceso del día, se ingresa la misma información, con la diferencia de que la columna de *primer acceso* de la tabla *registros* se deja vacía.

### c. Verificar retardo

Si se trata del primer acceso, se hace una última consulta para verificar la hora de llegada estipulada en el sistema para dicho usuario (esta hora está definida en la columna *horario\_entrada*, en la tabla *Usuario\_Acceso* de la base de datos) y la hora a la que se registró el acceso. Si el usuario realiza el ingreso antes de la hora estipulada por el sistema, se ingresa el registro en la base de datos, detallando el identificador del usuario, hora y fecha del ingreso, el punto de acceso por donde ingreso, y existe una columna llamada primer acceso, en la cual se pone un número 1.

Si se detecta que el usuario está llegando después de la hora definida en el sistema, se ingresará el registro con la misma información que en el punto anterior, con la diferencia que en la columna de primer acceso, se ingresará un 2 para identificar que se trata de un retardo.

Esto facilita en gran medida la generación de reportes de asistencia y retardos, ya que solo es necesario verificar quien no tiene 1 ó 2 en un día específico para obtener la lista de ausencias, y verificar quien tiene un 2 para obtener los retardos.

	HORA	IDCARD	IDNOMBRE_ACCESO	ENTRADA_SALIDA	PRIMERACCESO
1	2006-11-29.7.0.0.0	9F93A506000007E0	1	1	1
2	2006-11-29.12.0.0.0	9F93A506000007E0	1	1	
3	2006-11-29.11.0.0.0	9F93A506000007E0	1	2	
4	2006-11-29.12.20.0...	9F93B635000007E0	1	1	1
5	2007-11-29.7.52.45...	9F93C735000007E0	1	1	1
6	2007-11-29.17.14.4...	9F93A506000007E0	1	1	2
7	2007-11-29.18.14.4...	9F93A506000007E0	1	2	
8	2007-11-30.7.52.45...	9F93C735000007E0	1	1	1
9	2007-12-1.18.14.4.0	9F93C735000007E0	1	1	1
10	2006-11-26.12.0.0.0	9F93C735000007E0	1	1	1

Figura 5.12.: Tabla Registros.

## 5. Libera Actuador

Este método se manda llamar en caso de que se verifique un acceso positivo para dicho usuario en el punto de acceso en cuestión. Entonces se obtiene el comando de la tarjeta controladora correspondiente y se le envía una petición para que libere el actuador. Aquí también es necesario interactuar con una rutina que permite manejar los niveles de la señal RTS para no perder datos.

## 5.4. Módulo de control de acceso para Visitantes

### Descripción

La función de esta aplicación es llevar el control de acceso para los visitantes, a los cuales se les otorga una tarjeta RFID que les restringe el acceso a diversas áreas, de modo que solo puedan entrar a los lugares que ellos visitan. Además se lleva un historial almacenando los datos más relevantes de dichas personas, como lo son: nombre, apellido, hora de entrada, hora de salida, lugar y persona que visitó. Los datos anteriormente mencionados son guardados en una Base de Datos. También se almacena una fotografía del visitante, la cual es tomada por una webcam controlada por esta aplicación.

### Descripción de las API's usadas

Esta aplicación requirió de la utilización de la API **JMF** proporcionada por los mismos realizadores de Java y de el driver `mysql-connector` que fue necesario para poder acceder a la base de datos.

## JMF[42]

La Java Media Framework (JMF) permite procesar fuentes de datos multimedia con solo unas líneas de código. JMF es una extensa y versátil API usada principalmente para procesar *media* en tiempo real, los cuales suelen ser datos que cambian respecto del tiempo, como el audio, el video, las secuencias MIDI, y animaciones.

Algunos de los diversos usos que tiene JMF son:

- Reproducir varios archivos multimedia en una aplicación de Java.
- Reproducir *media* desde Internet.
- Capturar audio o video desde dispositivos como micrófonos y webcams, además de poder almacenar los datos obtenidos en alguno de los formatos soportados.
- Transmitir audio y video en tiempo real a través de Internet.

Los formatos soportados por esta API incluyen AU, AVI, MIDI, MPEG, QuickTime y WAV.

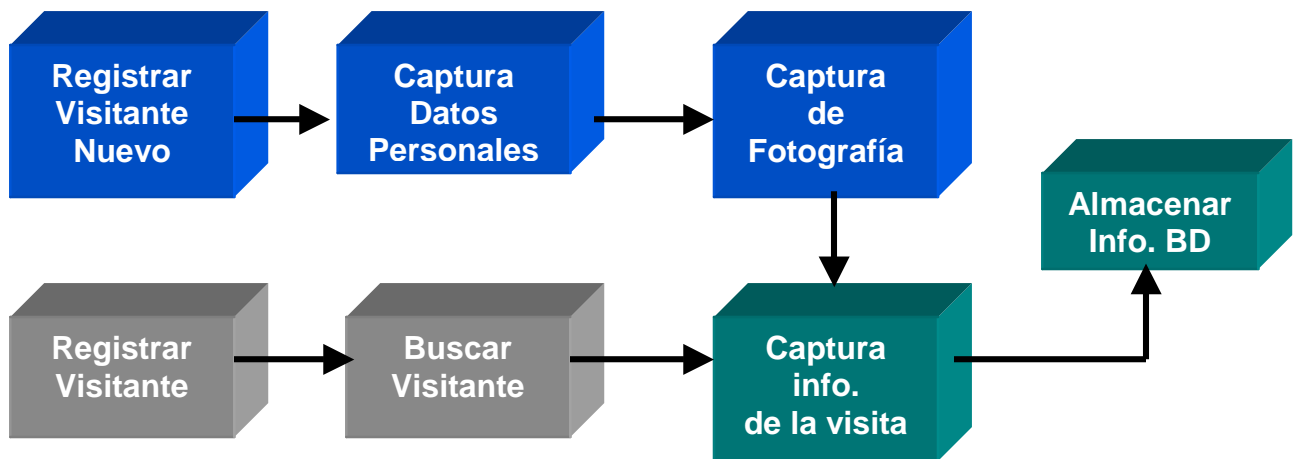


Figura 5.13.: Diagrama de bloques del módulo registro de visitantes.

En la Figura 5.13, se muestra un diagrama de bloques con el flujo que pueden tomar los registros de visitantes. Se intenta hacer este proceso lo más eficiente posible, de modo que en la primera visita se recauda la información personal y se toma una fotografía. Esta información es almacenada en la base de datos y es utilizada para accesos posteriores que pueden ser en distintas fechas. De modo que en una segunda visita, solamente se registra información básica, como el departamento que se visita, persona que se visita, asunto y se hace la asignación de una tarjeta RFID. Así se agilizan los accesos de los visitantes frecuentes.

## 5.5. Módulo de Administración de Usuarios

### Descripción general de la aplicación

Módulo para el registro y la administración de usuarios que permite realizar operaciones básicas como agregar, editar, eliminar y buscar usuarios.

### Descripción de las API's utilizadas

Para la realización de este módulo se utilizó una API para llevar a cabo la conexión con la base de datos. Esta API es la “mysql-connector-java-5.0.0-beta-bin”.

### Descripción específica de la aplicación

La aplicación inicia con la búsqueda de la base de datos. Una vez que se verificó que la base de datos existe y no se generó ningún error, la aplicación nos muestra la siguiente pantalla:



Figura 5.14.: Módulo de administración de usuarios.

A la hora de mostrar esta pantalla que está dentro de la clase Registro\_Usuario la JTable de datos generales se llena a través de la función LoadDB\_DatosGral. Una vez llenada la tabla el administrador puede seleccionar al usuario correspondiente para ver su información de asistencia y de accesos permitidos. Cada vez que se selecciona a un usuario de la JTable de datos generales, la JTable de datos de asistencia y la de accesos permitidos se llenan a través de las funciones de LoadDB\_DatosAsisExp y

LoadDB\_AccesosPer que realizará una consulta a través de los campos de idUsuario e idCard.

A continuación se describirán las acciones que se pueden realizar en esta pantalla.

## Búsquedas

Para la realización de búsquedas se tomaron en cuenta cuatro parámetros los cuales son:

- Todos
- Nombre
- Apellido
- Departamento

The screenshot shows a software window titled "Registro de Usuarios". It features a search section with a text input field labeled "Busqueda por Todos" and a dropdown menu for "Tipo de Busqueda" with options: Todos, Nombre, Apellido, and Departamento. To the right are four icons: "Busqueda" (magnifying glass), "Nuevo Usuario" (palette), "Editar Usuario" (pencil), and "Eliminar Usuario" (trash). Below is a table with columns: Nombre(s), Apellido(s), Departamento, and Extensión. The first row contains: [blank], Sánchez Noya, Comunicaciones, 55555. Below this is a section "Datos de Asistencia y Expiración" with a table:

Número de tarjeta	Fecha de Expiración	Hora de Entrada	Hora de Salida
12345	2007-01-01	14:00:00	18:00:00

At the bottom, there is a section "Accesos Permitidos" with a table:

Nombre Acceso	Tipo Acceso
Puerta Principal	Aceptado
Laboratorio 1	Aceptado
Laboratorio 2	Denegado
Laboratorio 3	Denegado

Finally, there are "Cancelar" and "Salir" buttons at the bottom right.

Figura 5.15.: Búsqueda de usuarios.

Para llevar a cabo la búsqueda utilizamos la función Query que recibe como parámetros dos cadenas, una que indica el contenido de la caja de textos y la otra que especifica el tipo de búsqueda que se va a realizar. Cuando se realiza una búsqueda incorrecta no aparece nada en el JTable de datos generales lo cual indica que no se encontró al usuario y en caso de que haya ocurrido un error en la búsqueda nos manda un mensaje de error.

## Nuevo Usuario

Para agregar un nuevo usuario se manda llamar a la clase de Datos\_Usuario a través de su constructor por default que nos genera esta pantalla:

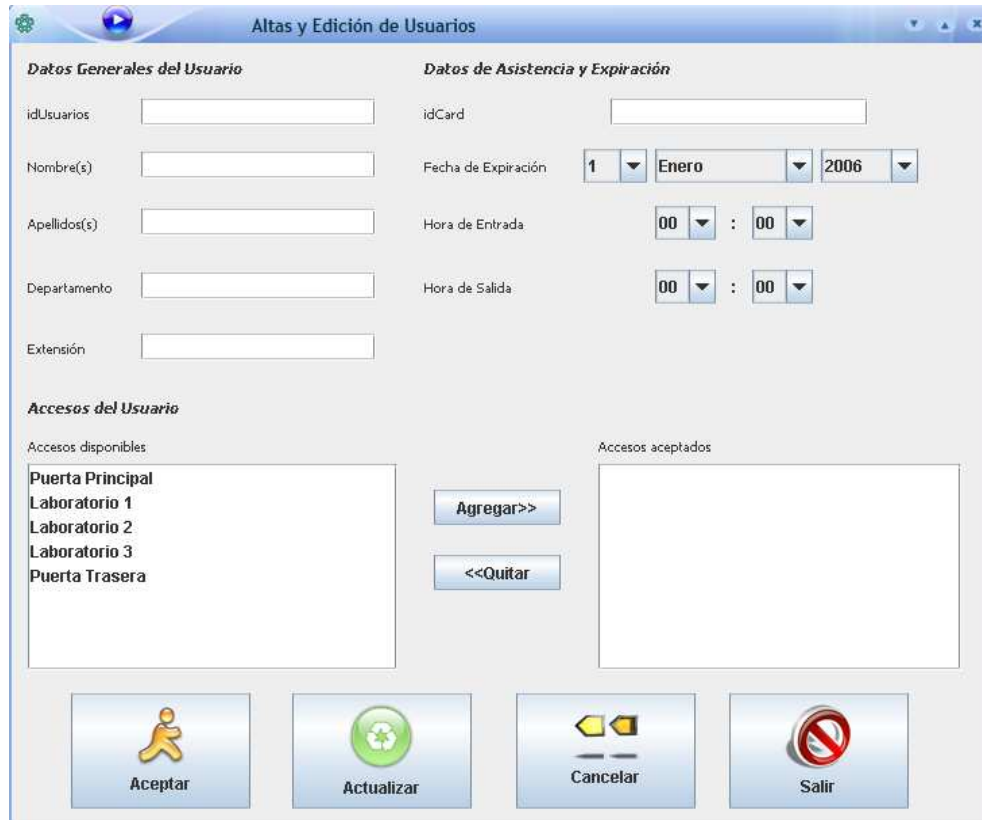


Figura 5.16.: Agregar usuario.

Lo primero que se hace en esta pantalla es llenar el JComboBox y el Jlist. Este último con información de la base de datos. Para llenar estos campos se hace uso de las siguientes funciones:

- **Fecha de expiración:** Se ocupa una función para llenar los campos de los días, los meses y los años
- **Hora de Entrada/Salida:** Estos se llenan por default, así que no se utiliza ninguna función.
- **Accesos de usuario:** Es importante que en la base de datos la tabla "Nombre\_Accesos" contenga datos para que se pueda llenar el Jlist con los puntos en donde se concederá el acceso.

Siguiendo el procedimiento de la creación de un nuevo usuario, una vez que se han realizado los pasos anteriores, el usuario administrador puede capturar el resto de los datos. No se deben de dejar espacios o cajas de texto en blanco porque se recibirá una advertencia. Una vez llenados todos los campos se pueden realizar tres acciones que se especifican a continuación:

- **Acceptar:** Cuando se presiona el botón de Aceptar se verifica que no haya campos vacíos y se ejecutan las siguientes funciones InsertDatosGral, InsertDatosAsisExp y InsertAccess que toman los datos de las cajas de texto, combo box y listas para insertar los datos.
- **Cancelar:** El botón cancelar realiza la función de limpiar todos los campos y ejecuta de nuevo el Agregar\_Accesos.
- **Salir:** El botón de salir nos regresa a la anterior pantalla sin realizar ninguna acción.

## Editar Usuario

Para poder editar a un usuario se debe de seleccionar este de la Tabla de datos generales que aparece en la ventana principal, una vez seleccionado se presiona el botón "Editar Usuario" este manda a llamar a la clase de Datos\_Usuario mediante un constructor específico que recibe los datos del usuario seleccionado para poder mostrarlos en esta pantalla y editarlos.

En esta ocasión la novedad es que podemos modificar casi todo. Lo que no se podrá modificar es el identificador del Usuario. Además se habilita otro botón que es el de Actualizar que también verifica los campos vacíos y ejecuta la función Update que actualiza los datos en la bases de datos.

## Diagrama de actividades

Se muestran todas las actividades que se llevan a cabo en la aplicación.



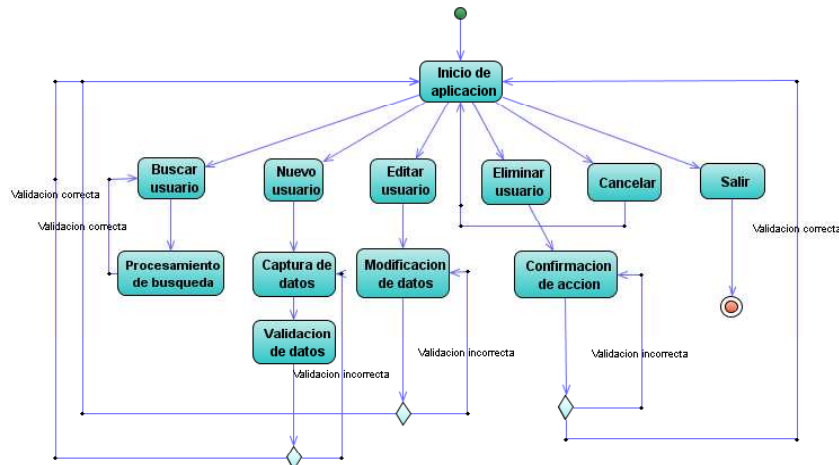


Figura 5.17.: Diagrama de actividades administración de usuarios.

- Inicio de aplicación: Básicamente es el inicio de la aplicación y la carga de la base de datos.
- Buscar usuario: Realiza la búsqueda de un usuario de acuerdo a ciertos parámetros.
- Procesamiento de búsqueda: De acuerdo a los parámetros obtenidos se realiza la búsqueda.
- Nuevo usuario: Abre la ventana para la captura de datos.
- Captura de datos: Se realiza la captura de los datos.
- Validación de datos: Se validan los datos accedidos.
- Editar usuario: Accesa a la pantalla para modificar la información de un usuario.
- Modificación de datos: Se modifican los datos en la base de datos.
- Eliminar usuario: Abre la ventana de confirmación de eliminación.
- Confirmación de acción: Se confirma la eliminación del usuario.

## 5.6. Módulo Generador de Reportes en PDF y en EXCEL

Aplicación que genera automáticamente reportes tanto en formato pdf como xls a través de consultas en una base de datos. La aplicación crea reportes diarios de usuarios, retardos e inasistencias.

API's utilizadas para el generador de reportes.

Esta aplicación se fundamenta en dos API'S que van a servir para crear los reportes en formato PDF y en Excel, la otra api utilizada nos sirve para mandar estos reportes vía correo electrónico a un remitente. Las API'S utilizadas son las siguientes:

- JasperReport
- JavaMail

## JasperReport [27]

JasperReports es una biblioteca open source para crear reportes de una manera simple y flexible. JasperReport tiene la habilidad de entregar contenido amplio en la pantalla ya sea con el Printer de JasperReports o en diferentes formatos (PDF, HTML, XLS, CSV XML).

Esta herramienta está completamente escrita en Java y puede usarse en una gran variedad de aplicaciones, incluyendo J2EE y aplicaciones WEB con la opción de generar contenido dinámico.

La definición de los reportes creados por JasperReport persiste en un estándar de Web abierto basado en un formato XML conocido como JRXML.

Para crear y modificar reportes sobre este estándar se pueden utilizar los siguientes métodos:

- Usando la API de JasperReport.
- Usando cualquier editor de texto.

## JAVAMAIL [42]

El API JavaMail es un paquete opcional (extensión estándar) para leer, componer, y enviar mensajes electrónicos.

El propósito principal de JavaMail es transportar, enviar, o re-enviar mensajes como sendmail u otros programas del tipo MTA (Mail Transfer Agent). En otras palabras, los usuarios interactúan con los programas para leer y escribir e-mails. Los programas MUA tratan con los programas MTA para el envío real.

El API JavaMail está diseñado para proporcionar acceso independiente del protocolo para enviar y recibir mensajes dividiéndose en dos partes:

- Enviar y recibir mensajes independientemente del proveedor/protocolo.

- La segunda parte habla de lenguajes específicos del protocolo como SMTP, POP, IMAP, y NNTP. Con el API JavaMail para poder comunicar con un servidor, necesitamos un proveedor para un protocolo.

## Generación de reportes

El módulo generador de reportes consta de un demonio que se encuentra corriendo constantemente, y cada 12:00:00 am, ejecuta el proceso de generación de reportes. Se realizan 3 consultas a la base de datos, para obtener la asistencia, retardos y ausencias. Estos resultados son pasados a jasper reports para generar los reportes en pdf y excel. Finalmente se envían por correo electrónico como archivo adjunto.

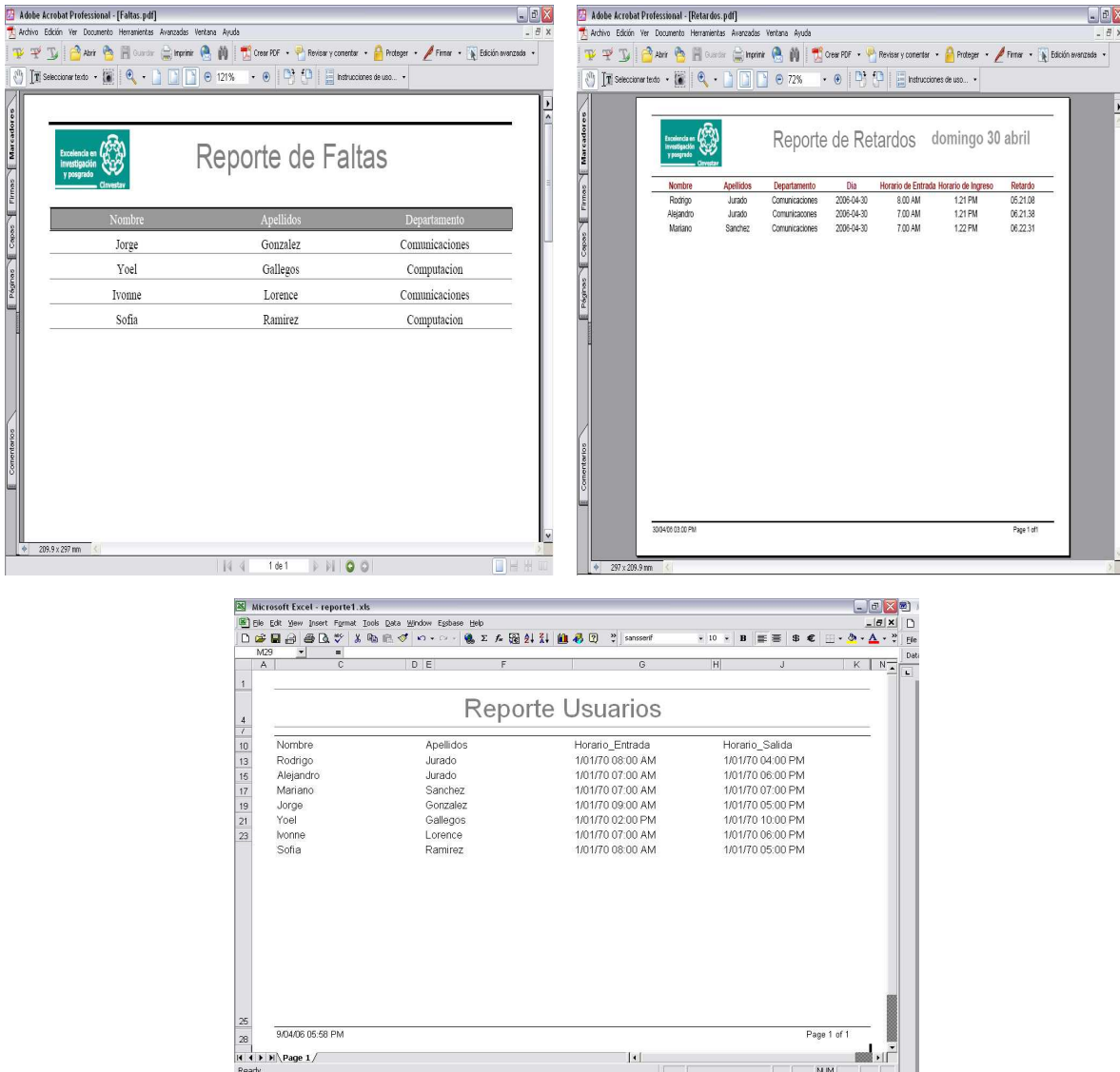


Figura 5.18.: Reporte de faltas, retardos en pdf y asistencia en excel.

## 5.7. Aplicación WEB

La evolución de las arquitecturas de sistemas en los últimos 30 años ha sido muy rápida, se inicia con sistemas de una capa, orientadas a las tareas de un solo usuario, sin la existencia de ambientes colaborativos. Después vinieron los sistemas de dos capas o cliente-servidor, los cuales permiten compartir información. Estos sistemas interactúan directamente con los usuarios finales, la lógica de negocio y la capa de presentación permanecen en el cliente, y los datos se encuentran en un servidor remoto. Aunque este esquema se sigue utilizando en la actualidad, tiene bastantes limitantes, como falta de flexibilidad del diseño y la portabilidad.

La siguiente generación de sistemas es la de aplicaciones web, basadas en sistemas de varias capas, que permiten el acceso remoto a aplicaciones a través de navegadores de internet. Estas aplicaciones, en un inicio, mostraban solo contenido estático, pero ahora este contenido es cada vez más dinámico y la tendencia actual es tener aplicaciones RIA (Rich Internet Applications) o basadas en Web 2.0. Estas aplicaciones tienen como objetivo mejorar la experiencia del usuario final, mediante interfaces más vistosas, con mayor funcionalidad, facilidad de uso y mejor desempeño. Para esto es posible utilizar componentes como JSF y AJAX[30].

JSF (Java Server Faces) es un framework que simplifica el desarrollo de aplicaciones web. Es la evolución de los struts (struts es un framework de desarrollo para aplicaciones web basadas en el patrón MVC) y es un componente del framework de Java 2 Enterprise Edition (J2EE).

### AJAX

Ajax (Asynchronous JavaScript + XML) no es una tecnología en sí mismo. En realidad, se trata de la unión de varias tecnologías que se desarrollan de forma autónoma[33]:

- XHTML (o HTML) y hojas de estilos en cascada (CSS) para el diseño que acompaña a la información.
- Document Object Model (DOM) accedido con un lenguaje de scripting por parte del usuario, especialmente JavaScript, para mostrar e interactuar dinámicamente con la información presentada.
- El objeto XMLHttpRequest para intercambiar datos asincrónicamente con el servidor web.
- XML es el formato usado comúnmente para la transferencia de vuelta al servidor.

Desarrollar aplicaciones AJAX requiere un conocimiento avanzado de todas y cada una de las tecnologías anteriores.

En las aplicaciones web tradicionales, las acciones del usuario en la página web (hacer click en un botón, seleccionar un valor de una lista, etc.) desencadenan llamadas al servidor. Una vez procesada la petición del usuario, el servidor devuelve una nueva página HTML al navegador del usuario.

Este tipo de aplicaciones web funcionan correctamente, pero constantemente tienen problemas de desempeño y no crean una buena impresión al usuario. Al realizar peticiones continuas al servidor, el usuario debe esperar a que se recargue la página con los cambios solicitados. Si la aplicación debe realizar peticiones continuas, la aplicación web se hace muy lenta.

AJAX permite mejorar completamente la interacción del usuario con la aplicación, evitando las recargas constantes de la página, ya que el intercambio de información con el servidor se produce en un segundo plano.

Las aplicaciones construidas con AJAX eliminan la recarga constante de páginas mediante la creación de un elemento intermedio entre el usuario y el servidor[34]. La nueva capa intermedia de AJAX mejora la respuesta de la aplicación, ya que el usuario nunca se encuentra con una ventana del navegador vacía esperando la respuesta del servidor.

Las peticiones HTTP al servidor se transforman en peticiones JavaScript que se realizan al elemento encargado de AJAX. Las peticiones más simples no requieren intervención del servidor, por lo que la respuesta es inmediata. Si la interacción requiere la respuesta del servidor, la petición se realiza de forma asíncrona mediante AJAX. En este caso, la interacción del usuario tampoco se ve interrumpida por recargas de página o largas esperas por la respuesta del servidor.

## Arquitectura de la aplicación WEB

La aplicación web para visualizar información de los usuarios, sus ausencias y retardos, fue desarrollada con ayuda del framework ADF[35]. ADF se compone de una serie de librerías que agilizan el desarrollo de aplicaciones java, tienen componentes preconstruídos que aceleran el tiempo de desarrollo, y utiliza múltiples patrones de diseño, y como resultado se pueden crear aplicaciones J2EE con excelente desempeño. Otra ventaja de este framework es que permite utilizar componentes AJAX, sin necesidad de programarlos desde cero.

La arquitectura de la aplicación desarrollada es similar a la mostrada en la Figura 6.18, en donde se tiene una capa de servicios de negocio, la cual es la capa de persistencia, que permite transformar las tablas de la base de datos, hacia clases java y se encarga del manejo de transacciones.

Esta capa de servicios de negocio está formada por tres componentes, Entity Objects, View Objects y Application Module:

- Entity Objects.- componentes que establecen una transformación entre las tablas de la base de datos, y las clases de java.
- View Objects.- componente construido sobre los entity objects que permite crear consultas para después utilizarlas en los componentes visuales, como los jsps[31].
- Application Module.- este componente se encarga del manejo de transacciones con la base de datos.

Estos elementos permiten crear componentes reutilizables, ya que se crea un entity object por cada tabla que se tenga en la base de datos. Sobre los entity objects se crean tantos view objects como consultas sean necesarias.

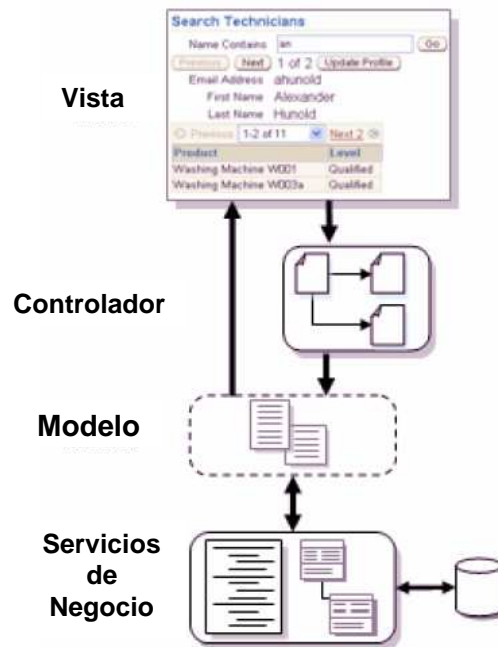


Figura 5.19.: Arquitectura de la aplicación web.

Por encima de la capa de servicios de negocio, se tiene un esquema basado en el patrón de diseño MVC (modelo-vista-controlador). La capa de modelo permite el acceso a los datos y ahí se encuentra la lógica de negocio. Como controlador y vista se utilizó JSF.

En la Figura 6.19 se muestra el diagrama del controlador, en donde se puede ver el esquema de navegación y las distintas páginas que conforman el sistema.

El sistema inicia en la página de Login, y si este es exitoso, se obtiene la página de inicio, de lo contrario será direccionado al jsp de error.

A partir de la página de inicio es posible acceder a las distintas búsquedas en el sistema.

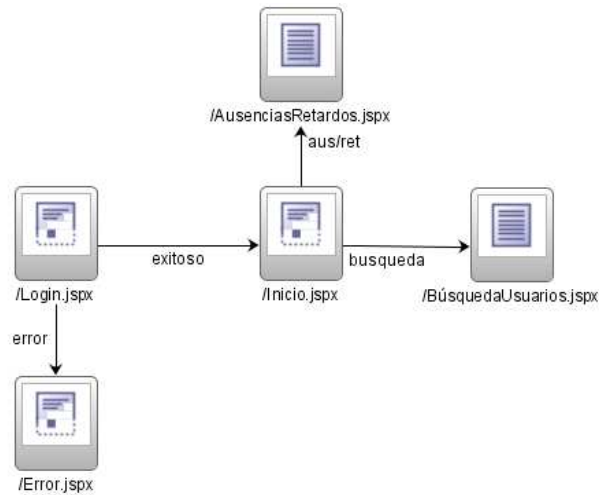


Figura 5.20.: Controlador de la aplicación web.

Esta aplicación fue desarrollada con ayuda de Jdeveloper 11, y fue publicada en un servidor de aplicaciones OC4J.

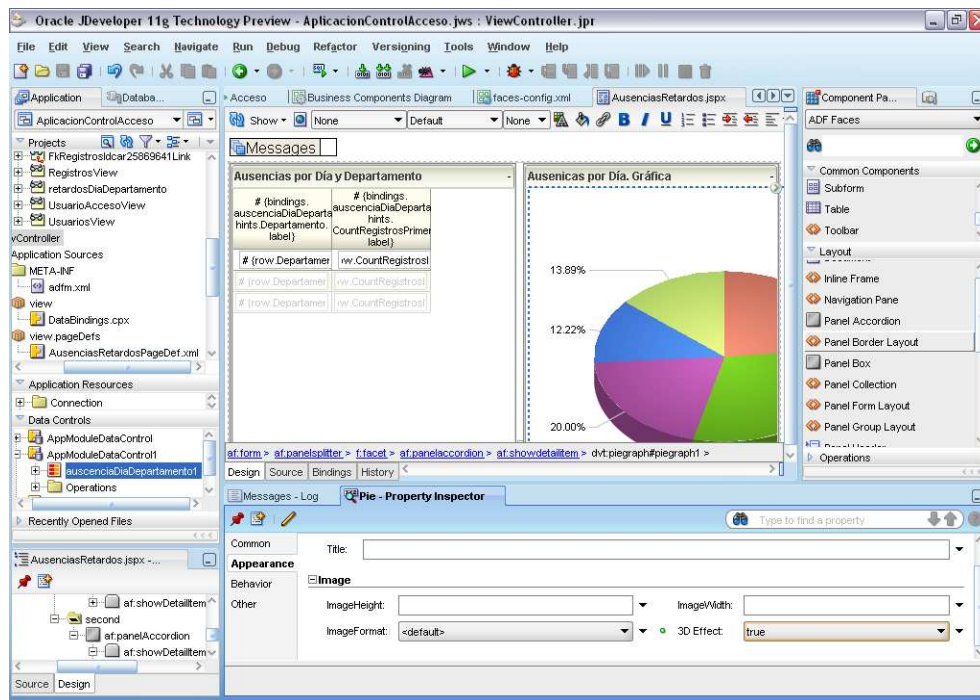


Figura 5.21.: Ambiente de desarrollo.

El objetivo de esta aplicación es permitir a la gerencia conectarse de manera remota, y obtener información del comportamiento del sistema.

Para esto se crearon consultas para obtener las faltas del día y los retardos. Estas consultas SQL tienen la siguiente forma:

Consulta para obtener las faltas del día.

```
select usuarios.idusuario, nombre, apellidos, departamento, extension from usuarios
, usuario_acceso where usuarios.idusuario not in(select usuarios.idusuario from
registros, usuarios, usuario_acceso, dual where
usuarios.idusuario=usuario_acceso.idusuario and
usuario_acceso.idcard=registros.idcard and to_char(hora,'dd-mm-
yyyy')=to_char(sysdate,'dd-mm-yyyy') and (registros.primeracceso=1 or
registros.primeracceso=2)) and usuarios.idusuario=usuario_acceso.idusuario
```

Esta consulta, es empleada para crear una tabla con la información de ausencias del día.

Retardos por día por departamento

```
Select usuarios.departamento , count(registros.PRIMERACCESO) from
usuarios, usuario_acceso, registros, dual where
usuarios.IDUSUARIO=usuario_acceso.IDUSUARIO and
usuario_acceso.IDCARD=registros.IDCARD and
registros.PRIMERACCESO=2 and
to_char(hora,'dd-mm-yyyy')=to_char(sysdate,'dd-mm-yyyy')
group by departamento
```

Esta consulta, se utiliza para generar un gráfico de pay, con la estadística de retardos, agrupado por departamentos.

A partir de estas consultas es posible generar modificaciones muy sencillas, para obtener estadísticas por día, mes y año.

## Interfaces gráficas

Para el desarrollo de la interfaz gráfica, se hizo uso de algunos componentes ajax, los cuales le permiten al usuario modificar el comportamiento de la página web, es decir, es posible modificar el orden de las columnas en las tablas, es posible tener ventanas tipo acordeon y estar cambiando entre ellas, o modificar su tamaño para tener una mejor visibilidad de algún componente. Y estas páginas hacen uso de partial rendering, lo que



permite al usuario actualizar componentes aislados, sin la necesidad de refrescar toda la pantalla. Lo que hace que estas sean más rápidas.

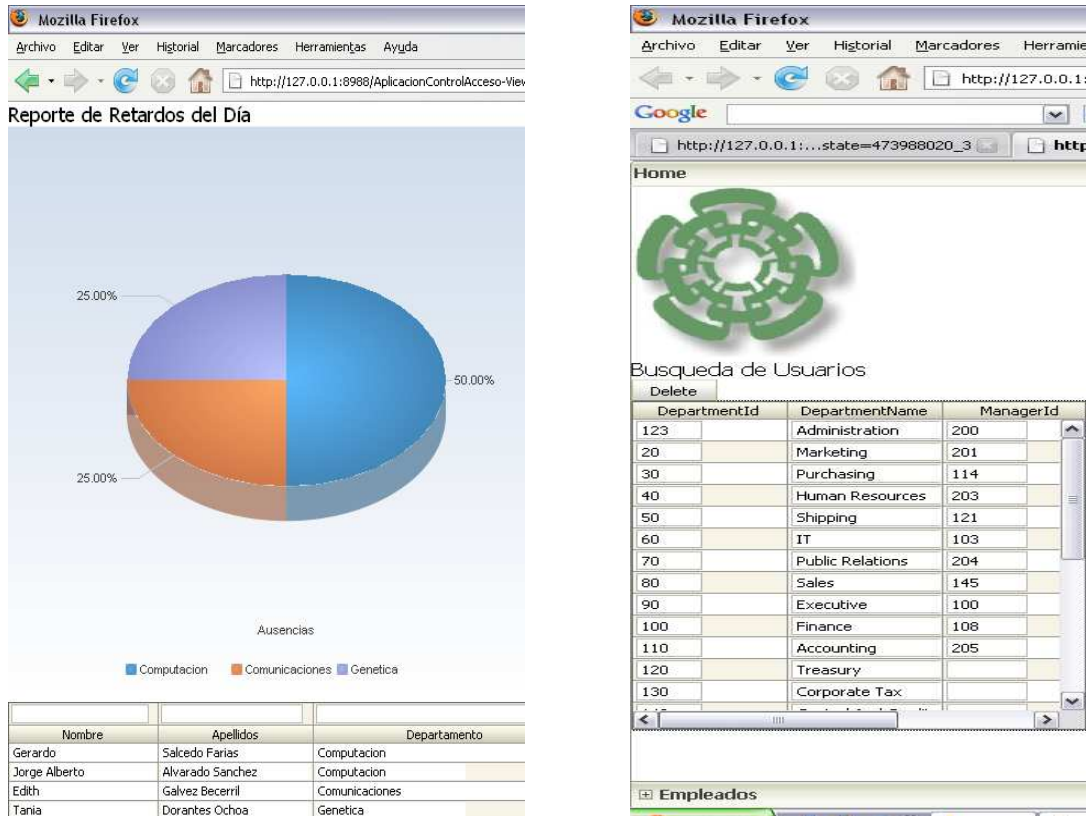


Figura 5.22.: Reporte de retardos del día y búsqueda de usuarios.

Aquí se muestran algunos ejemplos de las vistas que se pueden obtener del sistema.

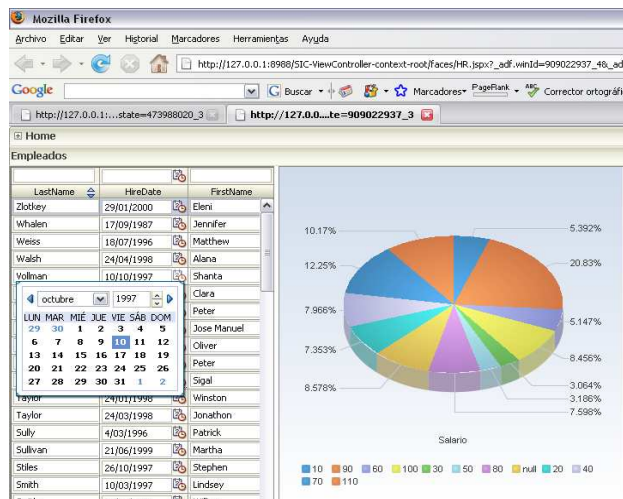


Figura 5.23.: Reporte de retardos del mes agrupados por departamento.

# Capítulo 6. Conclusiones y Trabajo Futuro

## 6.1. Conclusiones

### Tecnología RFID

RFID es una tecnología que presenta atractivas ventajas contra otras tecnologías de autoidentificación, en la actualidad todavía no llega a una madurez total, ya que falta terminar de definir estándares. Además existen retos y limitantes propias de RFID en donde todavía hay mucho por investigar, como es caso de limitantes de lectura en líquidos o metales. Las regulaciones en cuanto al impacto que existirá en la sociedad, en cuestiones de privacidad y seguridad todavía no han sido definidas. En un corto plazo, no se espera que RFID sustituya a otras tecnologías más bien están surgiendo nuevas áreas de aplicación para esta tecnología, se piensa que en un principio RFID complementa a otras tecnologías, que en algunos otros caso ni siquiera sea considerada y que en un futuro tal vez remplace algunas tecnologías de autoidentificación.

El interés por la tecnología de RFID se ha incrementando con rapidez. Muchas empresas y gobiernos están buscando aumentar la eficiencia de sus operaciones y reducir costos a través de esta tecnología. La oferta de este tipo de soluciones cada vez es mayor, en el mercado existen diversos fabricantes del hardware de RFID, y están empezando a desarrollarse empresas dedicadas a la implementación de RFID, con aplicaciones empaquetadas o con desarrollos a la medida.

Se espera que en aproximadamente 10 ó 15 años la tecnología RFID sea ubicua, ya que el costo de uno de sus componentes, el tag, ha venido bajando en los últimos años y para finales de 2008, se espera que el costo de las etiquetas llegue a US\$0.05 cada una, lo cual permitirá el empleo masivo de esta tecnología y el surgimiento de muchas aplicaciones que aprovechen sus bondades.

El poder de RFID se encuentra principalmente en 3 cualidades: la capacidad de poder leer etiquetas a distancia y sin necesidad de línea de vista, la capacidad de lectura/escritura y el poder identificar a elementos como únicos. Estas características son claves y representan un gran diferenciador al comparar RFID con otras tecnologías de autoidentificación. Hace un par de años se hablaba de RFID como el sustituto del código de barras. En la actualidad, se piensa más en una convivencia entre las distintas tecnologías, explotando las ventajas de ambas. Pero por ahora, existen aplicaciones en que RFID no resulta ser la solución más adecuada, por tema de precio o funcionalidad.

### Sistema de Control de Acceso

El sistema desarrollado, intentó abarcar todos los elementos involucrados en un desarrollo de RFID. El resultado fue un sistema funcional, que permite controlar el

acceso en determinados puntos y una fácil configuración del sistema, para, agregar, quitar o modificar puntos de acceso, hasta donde el hardware lo permite. Por limitantes del protocolo RS-485, solo se pueden tener hasta 32 dispositivos conectados en una red. Si se requiere agregar más dispositivos, la solución sería agregar una tarjeta RS232 al host, o agregar más hosts.

Se seleccionó el protocolo RS-485 por sus ventajas en cuanto a distancia (más de 1 km), pensando en poder controlar lectores y/o actuadores a estas distancias. Pero la tendencia actual es muy fuerte hacia utilizar componentes RFID interconectados en redes Ethernet y wifi. Esto tiene la ventaja de no tener la limitante de un número máximo de dispositivos en la red (más bien la cantidad de tráfico que los equipos de red soporten). Diversos analistas, establecen que en unos años, el tráfico de redes principalmente será generado por dispositivos de RFID[3]. Otra ventaja de utilizar ethernet o wifi, es que se puede recaer en los esquemas de verificación de errores de TCP/IP, y en caso de existir colisiones o errores en la transmisión, estas capas se encargan de solicitar la retransmisión de paquetes, en cambio en una red RS-485, este control debe ser hecho en capas más altas por la aplicación.

Para este proyecto, se tenían 2 lectores de RFID (inicialmente eran 3, pero uno se descompuso en la etapa de desarrollo), y se creó una tarjeta de conversión RS232 a RS485 y una tarjeta controladora. Por lo cual no fue posible probar el sistema con una carga mayor, pero las pruebas realizadas con estos componentes, fueron satisfactorias.

Para la implementación del control de acceso, se hizo uso de distintas arquitecturas de software, cliente/servidor, web y SOA. Para explotar las distintas características de cada una de estas.

El mundo digital en el que vivimos actualmente ha evolucionado a niveles en los que la información es muy valiosa. El tener la información adecuada, en el momento preciso, para tomar la decisión correcta, puede llegar a ser la diferencia entre una empresa exitosa y otra que no lo es. Esa es la razón por la cual en una empresa, es necesario poder consolidar la información que se genera a partir de los distintos sistemas. Por eso surge la necesidad actual del uso de componentes de software que permiten exponer la funcionalidad de ciertos sistemas como servicios que pueden ser consumidos por otras aplicaciones o sistemas.

Aquí es donde la arquitectura orientada a servicios toma importancia, y apoyada de estándares como BPEL, da un gran poder de integración y de generación de procesos de negocio. El resultado de este tipo de herramientas son arquitecturas estándares, flexibles y fáciles de mantener.

## 6.2. Trabajo Futuro

Los dispositivos de RFID utilizados son de alta frecuencia y tienen ciertas limitaciones en la distancia máxima de lectura. El siguiente paso, sería utilizar dispositivos que trabajen en el rango de frecuencias UHF y desarrollar una solución orientada hacia la cadena de suministro, donde RFID promete tener el mayor impacto. Al inicio de este proyecto, se planteó el uso de esta tecnología, pero los costos de estos dispositivos era mucho mayor. Una aplicación interesante es el manejo de inventarios inteligentes, en donde es posible cubrir un almacén con un arreglo de lectores, que permita tener un inventario preciso y en tiempo real.

El sistema de software desarrollado, permite el control y cierto nivel de configuración de los dispositivos lectores utilizados. Lo ideal, sería tener un módulo de software que pudiera trabajar con distintos modelos de lectores y fabricantes. Y que adicionalmente permitiera la configuración de estos dispositivos.

Los lectores y *tags* utilizados, tienen capacidades de lectura/escritura, esto podría ser utilizado para crear otro tipo de aplicaciones alrededor de este sistema. Un ejemplo podría ser un monedero electrónico, que pudiera ser utilizado en el comedor de la empresa.

Un sistema como el propuesto en esta tesis, debe estar disponible en todo momento, es decir, no se puede tolerar una caída del sistema. Para lo cual se podría pensar en esquemas de alta disponibilidad, con replicación de la Base de Datos y un host de respaldo en caso de fallas.

El modelo EPC como capa middleware en sistemas RFID, está tomando mucha fuerza, y se está convirtiendo en el estándar de la industria, sería muy interesante desarrollar una arquitectura de este tipo para probar sus capacidades.



# Apéndice A.

## Arquitectura Orientada a Servicios

Un sistema de RFID, no sería de utilidad, si no se explotara a fondo toda la información que se genera a través de los tags. En los capítulos anteriores se mostraron aplicaciones cliente-servidor y web desarrolladas como una solución integral de control de acceso con Tecnología RFID. Pero para llevar este tipo de soluciones a un nivel de mayor sofisticación, el sistema debería estar desarrollado de modo que su integración con otros sistemas sea lo más transparente posible. Un ejemplo de esta integración podría darse en una empresa en donde se cuente con algún sistema de nómina o de recursos humanos, con el cual sea necesario conectar el sistema de control de acceso, para pasar diariamente la lista de ausencias del personal y automatizar los ajustes necesarios en la nómina o en donde se requieran, sin necesidad de interacción humana.

En concreto lo que aquí se realizó fue exponer un servicio web con la funcionalidad de poder obtener las ausencias del día y poder compartir esta información con otras aplicaciones, con alguna interacción humana, con ayuda de herramientas SOA como BPEL.

### A.1. Arquitectura de integración del Sistema Desarrollado

La aplicación desarrollada permite el realizar un control de acceso, pero funciona como una unidad independiente, es decir, no se esta comunicando con ninguna otra aplicación o sistema que pudiera existir en un ambiente empresarial como cualquier elemento de tecnología como FTP, archivos, bases de datos, colas de mensajes, aplicaciones ERP, CRM o cualquier aplicación propietaria. Lo que aquí se propone, es agregar una capa de servicios web al módulo de Control de Acceso, de modo que utilizando a BPEL como orquestador de servicios, sea posible definir procesos, en los cuales se interactue con servicios expuestos por cualquiera sistema existente en la institución. Y de esta forma poder compartir la información generada por el sistema de control de acceso con otras aplicaciones.

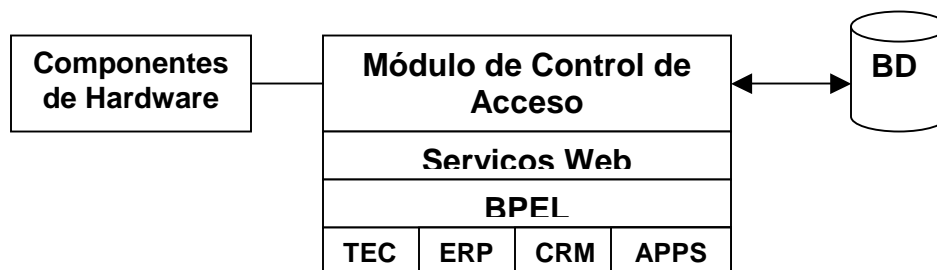


Figura A.1.: Arquitectura de integración del sistema desarrollado.

## **A.2. Conceptos SOA**

### **SOA (Service Oriented Architecture)**

Actualmente, al interior de las empresas, se puede encontrar un gran número de sistemas heterogéneos, esto es aplicaciones hechas en casa, sistemas ERP, CRM, mainframes, bases de datos, herramientas de colaboración, sistemas de inteligencia de negocio, etc., todos interactuando entre sí. Comúnmente la integración entre estos sistemas se realizaba en un esquema EAI (Enterprise Application Integration), que es una integración puntual, lo que da como resultado componentes fuertemente ligados y arquitecturas difíciles de mantener.

Los servicios, de manera muy parecida a los componentes, son bloques construidos de manera independiente que en conjunto crean procesos o aplicaciones empresariales. Al contrario de las arquitecturas orientada a objetos, SOA está formada por servicios de aplicación débilmente acoplados y altamente interoperables y, a diferencia de los componentes de software, los servicios tienen un conjunto de características que les permiten ser parte de una arquitectura orientada a servicios[36].

Una de estas características es su completa autonomía de otros servicios y la ventaja de estos es que pueden ser consumidos desde otras plataformas, esto es, no se requiere compatibilidad de sistemas operativos ni de lenguajes de programación. Así, una arquitectura SOA permite integrar múltiples servicios, con la ventaja de que estos componentes están débilmente acoplados, esto permite tener una arquitectura flexible, fácil de mantener y extender o modificar en menor tiempo y costo.

### **WebServices**

El término “Web Services” se refiere a un grupo de estándares de interoperabilidad (WSDL, XML y XML Schema, SOAP, JMS, JCA, etc) que simplifican la integración entre sistemas heterogéneos.

### **BPEL (Business Process Execution Language)[38]**

BPEL es un estándar industrial dirigido por la organización OASIS que permite orquestar y ejecutar procesos de negocio.

Los lenguajes de alto nivel como BPEL o WS-coordinación llevan el concepto de servicio un paso adelante, al proporcionar métodos de definición y soporte para flujos de trabajo y procesos de negocio[39].

BPEL permite enviar y recibir mensajes XML con servicios remotos, manipular datos en estructuras XML, recibir mensajes XML de forma asíncrona de servicios remotos, manejo de eventos y excepciones, definir flujos paralelos e introducir tareas humanas.

Orquestar se refiere a secuenciar los servicios y proveer la lógica adicional para procesar datos.

### **A.3. Descripción de la integración de la aplicación de control de acceso con otros sistemas**

Para resolver este planteamiento, se desarrolló una solución basada en una arquitectura orientada a servicios.

Se desarrollaron los siguientes componentes con ayuda de Jdeveloper versión 10.1.3 y el servidor de aplicaciones OC4J:

#### **1. Clase Java “ControlDeAsistencia” que obtiene ausencias del día.**

Basados en el diseño de la base de datos con la que se trabajo, es posible verificar los usuarios o empleados que no registraron un acceso a las instalaciones, con ayuda de la siguiente consulta SQL:

```
select nombre, apellidos ,idusuario from usuarios where  
idusuario not in(select usuarios.idusuario from registros,usuarios,usuario_acceso,dual  
where usuarios.idusuario=usuario_acceso.idusuario and  
usuario_acceso.idcard=registros.idcard and  
to_char(hora,'dd-mm-yyyy')=to_char(sysdate,'dd-mm-yyyy') and  
(registros.primeracceso=1 or registros.primeracceso=2))");
```

El módulo de control de acceso mostrado en el capítulo 6, cada vez que registra un acceso, se verifica si es el primer registro del día de ese usuario. De ser así, se marca el registro de acceso en la base de datos como el primer acceso de día. Adicionalmente este módulo verifica la hora de entrada establecida para ese usuario, y si el primer acceso del día se da después de esa hora, marca el registro como primer acceso con retardo. Estas acciones facilitan la creación de los reportes de asistencia y retardos.

Dentro de la clase de java, se desarrollaron dos métodos: uno que obtiene los nombres de las personas que se ausentaron en el día, y otro que obtiene sus identificadores (su idusuario como son reconocidos por el sistema). Estos métodos realizan la consulta anterior para obtener los nombres e identificadores de los ausentes e introducen cada uno de estos registros como elementos de una lista de java. De modo que si se manda llamar al método “AusenciaUsuario”, se obtiene como respuesta una lista con nombre y si se manda llamar al método “AusenciaUsuarioID”, la respuesta es una lista con identificadores.



## 2. La Clase “ControlDeAsistencia” es expuesta como un servicio web.

Se creó un documento wsdl que describe la funcionalidad, ubicación, tipos de datos, parámetros de entrada y de salida del servicio. Como la aplicación java tenía dos métodos, en el wsdl, se describen los dos servicios.

En la Figura A.2 se muestra una representación gráfica del archivo wsdl.

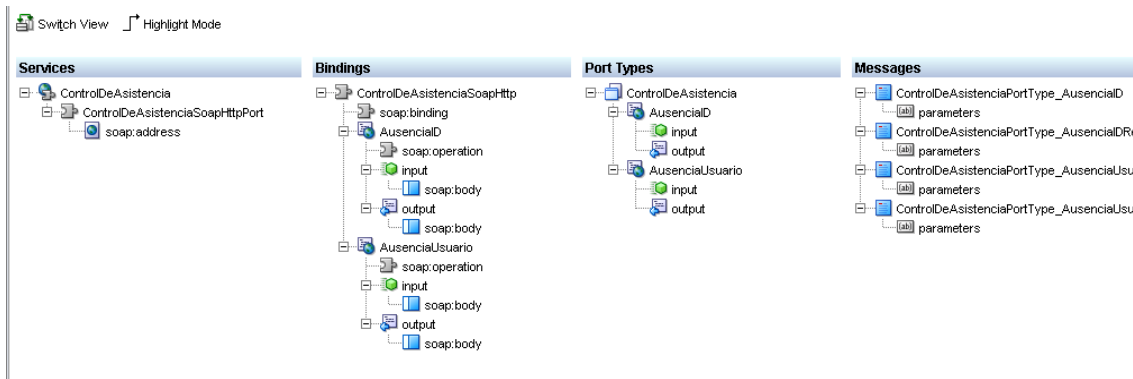


Figura A.2.: Representación del archivo WSDL.

Posteriormente se publicó este servicio web en un servidor de aplicaciones OC4J. En la siguiente figura, se aprecia como quedaron publicados los 2 métodos de la clase java, que son vistos como operaciones dentro del servicio web (Ausencia Usuario y AusenciaID).

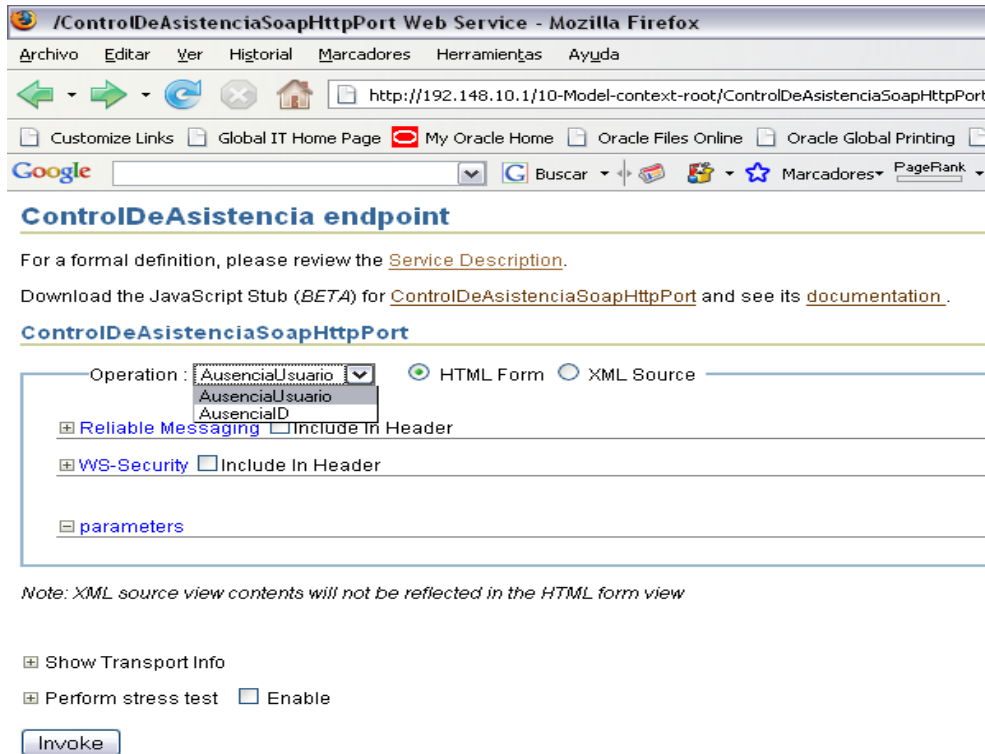
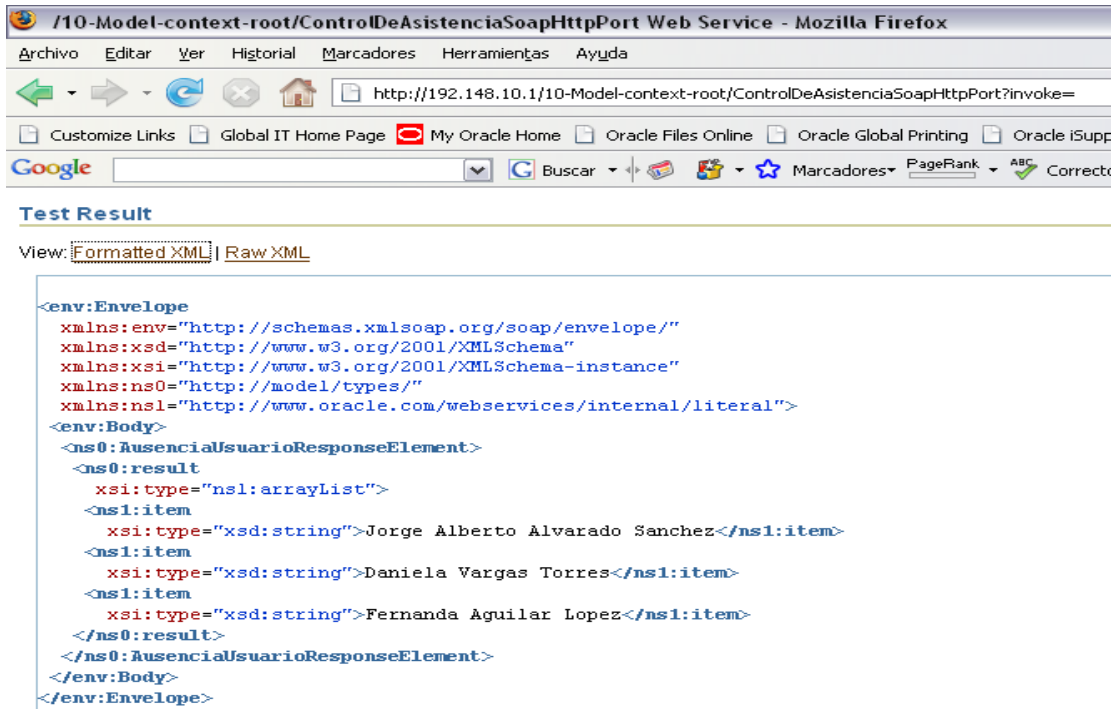


Figura A.3.: Página de prueba del Servicio Web.

Al invocar la operación AusenciaUsuario, ejecuta la consulta sql y responde con los nombres de las personas que se ausentaron. Esta información es regresada en unos formatos XML, lo cual facilita, en gran medida, la manipulación de la información.



### 3. Desarrollar documentos XML schema

Una vez que tenemos expuesto el servicio, vamos a extender su funcionalidad, buscando una arquitectura SOA a través de BPEL. Antes, se debe desarrollar un documento XML Schema, el cual describe la estructura de los datos que se van a pasar a través de del flujo de BPEL. Lo interesante de esta tecnología es que podemos crear nuestras propias estructuras, por ejemplo: si este sistema de control de acceso se utilizara en múltiples sucursales, sería muy conveniente consolidar toda esta información. Para ello se podría agregar funcionalidad al servicio web, creado anteriormente, para que adicionalmente tuviera una operación que contestara con la dirección de la sucursal en cuestión. Para poder manipular esta información desde BPEL, se desarrollo el siguiente documento XML Schema (XML schema, es una representación basada en xml, pero gráficamente se podría ver como en la siguiente figura.)

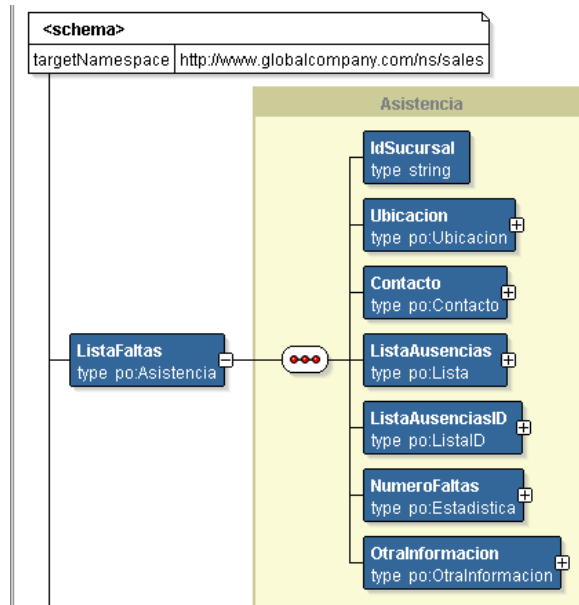


Figura A.5.: XML Schema.

La ventaja de esta tecnología es que permite crear tipos de datos complejos, de modo que es posible crear un tipo “ListaFaltas”, que en su interior contenga otro más complejo como dirección, que a su vez, está formado por: calle, municipio, estado, código postal y país (estos últimos todos de tipo cadena o string).

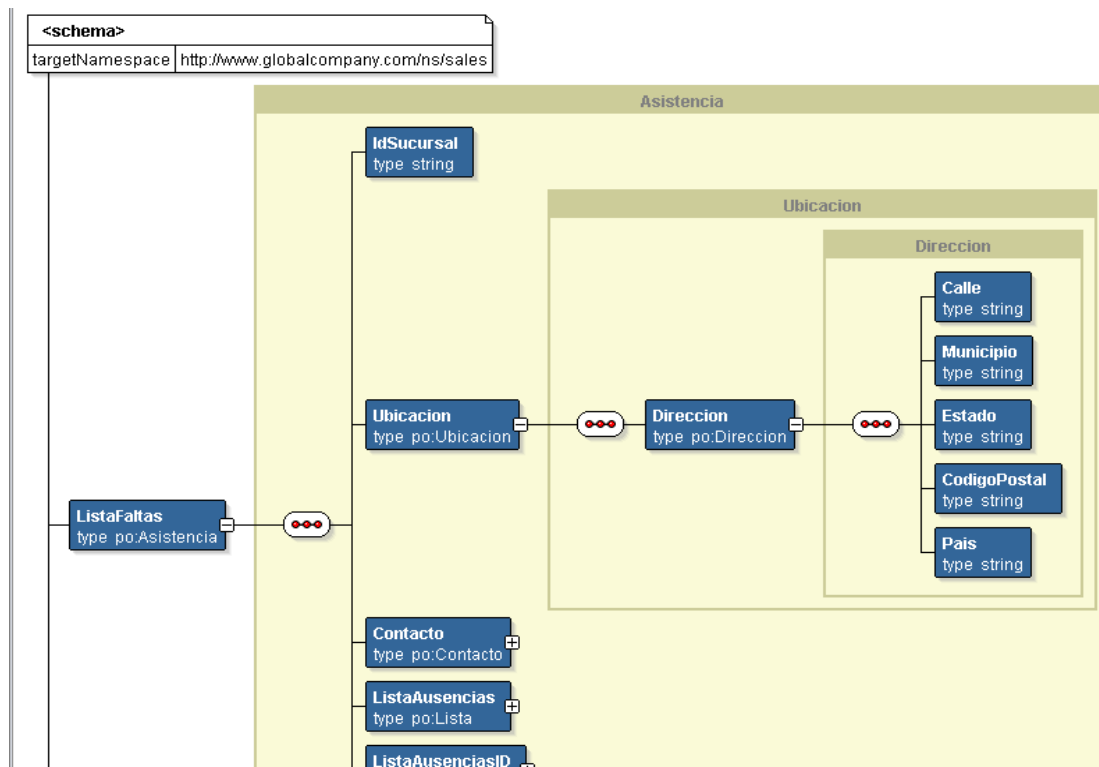


Figura A.6.: Tipo de dato complejo XML Schema.

De modo que cuando se trabaje con BPEL, debe utilizarse un documento xml que utilice el formato del xml schema creado, para que los distintos sistemas, puedan entenderse. Esto quedará más claro en el siguiente punto.

#### **4. Desarrollo de proceso BPEL**

BPEL o Business Process Execution Language es un estándar (soportado por OASIS) de la industria con gran aceptación, el cual funciona como orquestador de servicios. Es un lenguaje basado en XML, que surgió tras 10 años de desarrollo conjunto entre Microsoft (XLANG) e IBM (WSFL, FDML). Este tipo de tecnología es conocida como programación en largo, ya que prácticamente lo que se hace con ellas es programar, pero a diferencia de la programación convencional como java, c o c++ (conocida como programación en corto) en la que se utilizan funciones o subprogramas, en este estilo se aplican servicios, que en la mayoría de los casos pueden ser vistos como servicios web.

Este tipo de soluciones son de gran valor, ya que permiten la reutilización de componentes (es posible integrar sistemas legado o mainframe). Dan mayor flexibilidad a los procesos.

Adicionalmente, otra ventaja de este estándar, es que una vez publicado, es visto, así mismo, como un servicio web, de modo que puede ser consumido de igual forma por cualquier otro sistema.

Por ejemplo, basado en la solución de control de acceso, se podría tener un flujo de BPEL, que al iniciar, mandara llamar la operación del servicio web que se describió en el punto 2, para obtener los nombres de las personas que se ausentaron en el día. Pensando en que se tuvieran muchas sucursales, se podría consumir otro servicio web que contestara con el identificador de la sucursal, y en base al diseño, se podría recuperar cualquier tipo de información que se requiriera, como dirección, información de contacto, responsable de la sucursal, teléfono, entre otros.

La idea es facilitar la interacción entre sistemas heterogéneos, a partir de BPEL, de modo que sea posible intercambiar información entre aplicaciones, servicios web, bases de datos, archivos, servidores ftp, colas de mensajes, sistemas legados, aplicaciones tipo ERP(Planificación de Recursos Empresariales) o CRM(Administración de la Relación con los Clientes), entre otros.

Una vez que el sistema de control de acceso genera la información de las faltas. Con la información obtenida de las sucursales y de las personas que tuvieron inasistencia se pueden realizar varios procesos tales como: enviar un informe a los sistemas de nómina, para efectuar los descuentos correspondientes, enviar un reporte a los sistemas de recursos humanos para actualizar el historial de los usuarios.

Sería posible interactuar con una aplicación que verificara si la persona en cuestión se encontraba de vacaciones, de incapacidad o si se trataba de una falta justificada. Inclusive

se podría escribir un registro en otra base de datos, o enviar el total de faltas del día por correo electrónico a determinada persona y adicionalmente se podría solicitar la interacción humana de algún supervisor, para que evaluara si es necesario tomar alguna acción específica.

## **Proceso BPEL Control de Asistencia**

A pesar de que BPEL es un estándar basado en xml, existen herramientas que permiten desarrollar flujos BPEL de alto nivel como utilización y configuración de componentes gráficos, sin necesidad de trabajar a nivel de xml. En este caso se utilizó Jdeveloper.

Proceso BPEL:

### **1. Cargar xml Schema**

Se utilizó el documento xml schema, para este proceso BPEL, de modo que a lo largo del proceso se pudieran ir completando todos los campos mediante la consulta de diferentes servicios.

### **2. Crear proyecto de BPEL**

Se creó un proyecto de BPEL, inicialmente, sólo la parte del flujo que permite consumir la operación, para obtener los nombres de las personas que faltaron, a partir del servicio web mostrado al inicio de este capítulo.

### **3. Obtener Lista con nombres**

Se crea un componente llamado partner link (en la Figura 7.6 es el componente que se observa del lado derecho), al cual se le pasa la ubicación del documento wsdl que describe a nuestro servicio web, en este caso: <http://192.148.10.1/10-Model-context-root/ControlDeAsistenciaSoapHttpPort?WSDL>.

El componente reconoce automáticamente todas las operaciones existentes en dicho servicio.

Como se mencionó anteriormente este proceso es muy parecido a la programación, que se hace a continuación es utilizar 3 componentes. El primero llamado invoke para que invoque al servicio (mande llamar la funcionalidad que existe en él). El segundo llamado assign para que pase los parámetros necesarios para ejecutar dicho servicio (por ejemplo en el caso de un servicio web con el servicio de verificación de crédito, en este punto se tendrían que pasar el numero de la tarjeta de crédito y la fecha de expiración). Finalmente, se utiliza otro componente assign para obtener la respuesta del servicio invocado (en el caso de la verificación de crédito, este último componente, obtendría el monto total de crédito disponible para dicho usuario).

Lo que hace el proceso es simplemente obtener una lista de nombres, para poder compartir esta información con alguna otra aplicación.

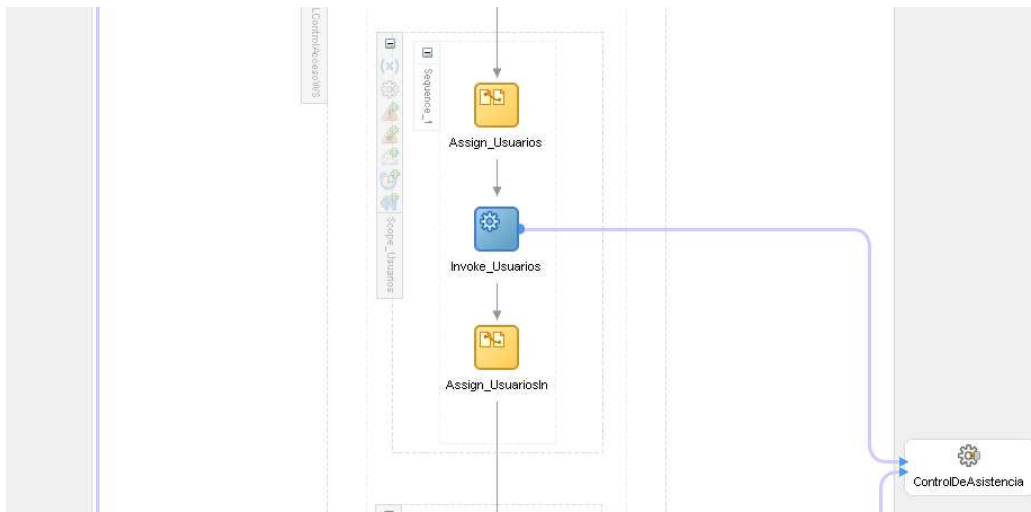


Figura A.7.: Proceso BPEL que invoca un servicio web.

#### 4. Obtener Lista con identificadores

Este punto es muy parecido al anterior, la única diferencia es que se consume la operación “AusenciaID” del servicio web, la cual regresa una lista con los identificadores de los usuarios que no llegaron. Se piensa que algunas aplicaciones pudieran trabajar con estos identificadores.

#### 5. Tarea Humana

Como se mencionó anteriormente, esta tecnología tiene la capacidad de crear interacciones humanas, de modo que por ejemplo algún supervisor reciba una notificación con la lista de personas que obtendrían un descuento de nómina en el mes, por haber excedido el número de faltas autorizadas, además que sea capaz de autorizar o rechazar dicha operación.

En este caso simplemente se creó una interacción humana de modo informativa para que el supervisor esté enterado de las personas que llegan tarde. En esta interacción humana, es necesario configurar, quién debe realizar esta tarea, quién es el responsable de la tarea, qué información se le debe de mostrar al supervisor, qué acciones puede tomar el (autorizar, rechazar, delegar, escalar, etc.) y definir cómo es que va a expirar la tarea si no se cumple en cierto tiempo.



Figura A.8.: Componente BPEL de interacción humana.

Este componente viene integrado con una aplicación web, de modo que las solicitudes de petición son redireccionadas a este sitio para que el supervisor tome acción sobre de ellas.

## 6. Escribir un Archivo

Finalmente se utiliza un adaptador para generar un archivo que contenga toda la información generada durante el proceso. Este es un archivo txt (o la extensión que se requiera), que es guardado en cualquier carpeta dentro del servidor. Esta tarea se realizó con el objetivo de mostrar los alcances de esta tecnología.

Para lo antes mencionado, se creó el adaptador de archivo, en donde se define el nombre de los archivos que se van a generar (en esta caso el nombre de los archivo, será la fecha del día), así mismo, se define la extensión del archivo, y en dónde se va a guardar.

Posteriormente, se utiliza un invoke (componente de lado izquierdo abajo) el cual hace el llamado del servicio de escritura de archivo.



Figura A.9.: Adaptador de archivo.



## A.4. Ejecución del Proceso BPEL

Una vez publicado el proceso BPEL en el servidor de aplicaciones, es posible ejecutarlo para constatar su funcionamiento correcto. Como se mencionó anteriormente, un proceso BPEL puede ser consumido como un servicio web o por medio de APIs. En este caso, se interactuará con él como servicio web (existen múltiples herramientas en el mercado que sirven para probar servicios web y analizar los mensajes SOAP que son intercambiados. En esta ocasión se utilizará una consola que viene integrada con la herramienta de BPEL).

The screenshot shows the 'Initiate' tab of a BPEL console. At the top, it displays 'BPEL Process: BPELControlAccesoWS', 'Version: 1.0', and 'Lifecycle: Active'. Below this, there are statistics for '0 Open Instances' and '11 Closed Instances'. The main area is titled 'Initiating a test instance' and contains a form with the following fields:

- Operation:
- Form type:  HTML Form,  XML Source
- WS-Security:  Include In Header
- WS-Addressing:  Include In Header
- payload:
  - IdSucursal:  xsd:string
  - Ubicacion:
  - Direccion:
    - Calle:  xsd:string
    - Municipio:  xsd:string
    - Estado:  xsd:string
    - CodigoPostal:  xsd:string
    - Pais:  xsd:string
  - Contacto:
    - NumeroTelefono:  xsd:string
    - CorreoElectronico:  xsd:string

Figura A.10.: Consola de prueba de BPEL.

Después de invocar el servicio, la consola de BPEL, permite monitorear los procesos y ver en el estatus que se encuentran, en esta consola los componentes del proceso, como se muestra en la siguiente figura y es posible hacer click, en los componentes para visualizar la información que paso por ahí.

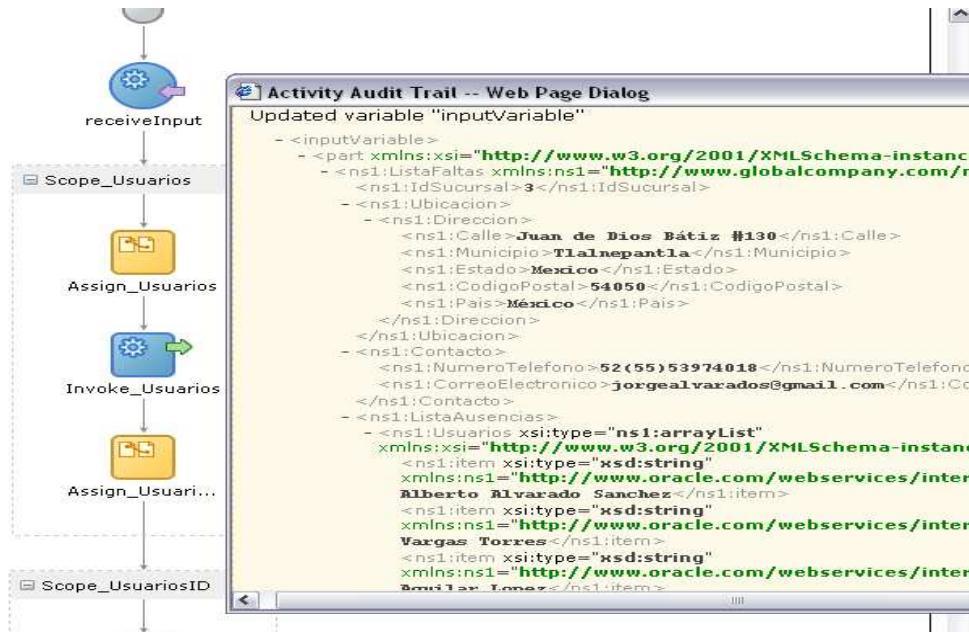


Figura A.11.: Visualizaci3n de la informaci3n en el proceso BPEL.

En este punto el proceso BPEL, ya recuper3 la lista de nombres y de identificadores, a partir del servicio web, y se encuentra esperando la interacci3n humana.

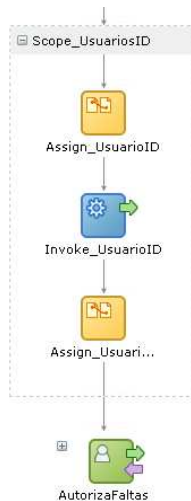


Figura A.12.: Proceso BPEL en espera de interacci3n humana.

Para poder interactuar con el proceso de manera manual, es necesario entrar a la aplicaci3n web, que permite hacer esto e iniciar una sesi3n con las credenciales de alg3n supervisor, para tener acceso al siguiente sitio.

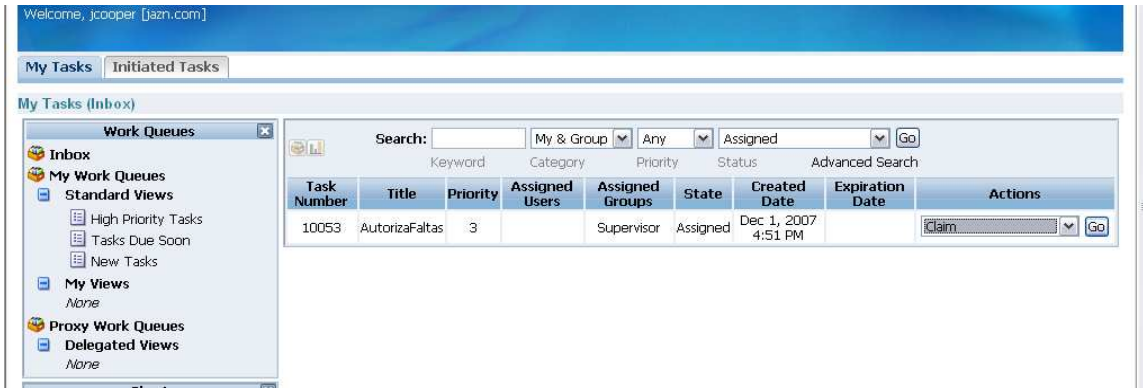


Figura A.13.: Aplicación Web de interacción humana.

En el esquema es posible ver todas las tareas pendientes, y hacer click en ellas para obtener más detalle de la información para finalmente tomar acciones al respecto.

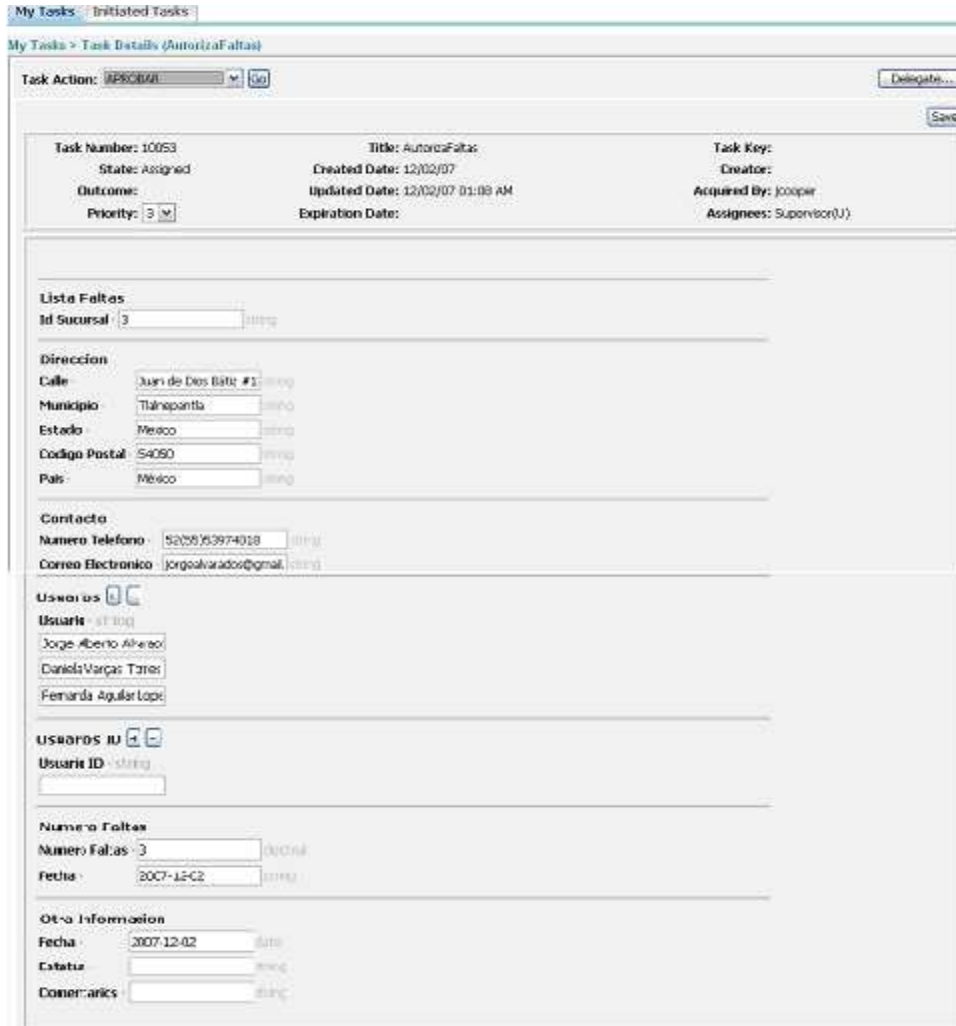


Figura A.14.: Interacción humana, detalle del proceso.

En este caso, el supervisor tiene la posibilidad de aprobar o rechazar la solicitud. Después del componente de interacción humana, existe un componente switch(como se mencionó anteriormente, este esquema es muy parecido a la programación convencional y tiene componentes similares), con el cual es posible ejecutar diferentes acciones dependiendo si se selecciona “aprobar” o “rechazar”.

Finalmente, el proceso escribe un archivo con la información almacenada a lo largo del proceso y termina.

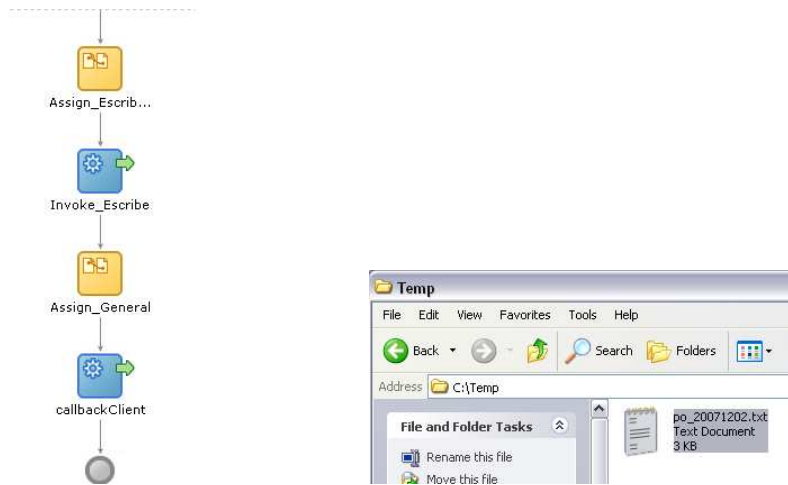


Figura A.15.: Escritura de archivo.

La funcionalidad de escribir un archivo puede utilizarse como una forma de registro, como otra forma de reportar la información, o inclusive como forma de integración con otras aplicaciones, por ejemplo alguna aplicación podría procesar el archivo.

Otro posible uso de este tipo de herramientas, sería que en el momento que el personal de recursos humanos diera de baja a algún usuario del sistema propio de RH, se iniciara un proceso de BPEL, el cual diera de baja al usuario de la base de datos de control de acceso, o cambiara todos sus accesos a “Denegado”.



# Bibliografía

- [1] R. Weinstein, *RFID: a technical overview and its application to the enterprise*, & IT Professional, Volumen 7(3): 27-33, Junio 2005.
- [2] Garfinkel, S.L., Juels, A., Pappu, R., *RFID privacy: an overview of problems and proposed solutions*, & Security and Privacy Magazine, IEEE Volume 3(3):34-43, Mayo-Junio, 2005.
- [3] Philipose, M. Smith, J.R. Jiang, B. Mamishev, A. Sumit Roy Sundara-Rajan, K. *Battery-free wireless identification and sensing*, Pervasive Computing, IEEE, Volumen 4(1): 37-45, Marzo 2005.
- [4] *Texas Instruments S6400 Reference Manual*. Guide for System Integrators RI-H4R-S5H3, Agosto 2003.
- [5] Technical Reference Texas Instruments Tag-it HF-I Transponder Inlay Extended Commands and Options, Mayo 2002.
- [6] J. Axelson; "Serial port complete: programming and circuits for RS-232 and RS-485 links and networks"; Madison, WI: Lakeview Research, 1998.14 [1.1]
- [7] RFID Essentials, Himanshu Bhatt, Bill Glover, O'Reilly, January 2006
- [8] V. Daniel Hunt, Albert Puglia, Mike Puglia, *RFID A guide to radio frequency identification*. Ed. Wiley 2007.
- [9] <http://www.macsema.com/buttonmemory.htm>
- [10] Patrick J. Sweeney, *RFID for Dummies*, Wiley Publishing, Inc 2005
- [11] Hassan, T. and Chatterjee, S., *A Taxonomy for RFID System Sciences*, 2006. HICSS '06. Proceedings of the 39th Annual Hawaii International Conference on Volume 8, 04-07 Jan. 2006 pp:184b - 184b
- [12] <http://www.epc.org.mx/view.php?id=1>
- [13] <http://www.itapizaco.edu.mx/paginas/JavaTut/froufe/parte19/cap19-1.html>
- [14] Sharyn Leaver with Tamara Mendelsohn, Christine Spivey Overby, and Esther H. Yuen, *Evaluating RFID Middleware Picking The Right Solution For Integrating RFID Data Into Business Applications*, Forrester Research, Inc. August 13, 2004

- [15] Ken Traub, *ALE: A New Standard for Data Access*, RFID JOURNAL Apr. 18, 2005
- [16] Joseph E. Hoang and Craig W. Thompson, *Architecting RFID Middleware*, IEEE Internet Computing, 2006 pp. 88-92
- [17] Tom Miller, *RFID Insider*, January 05, 2006 - RFID Connections
- [18] Manish Bhuptani and Shahram Moradpour, *RFID Field Guide: Deploying Radio Frequency Identification Systems*, Prentice Hall Professional Technical Reference February 2005
- [19] Phillips, T.; Karygiannis, T.; Kuhn, R.; *Security standards for the RFID market*, Security & Privacy Magazine, IEEE, Volume 3, Issue 6, Nov.- Dec. 2005 Paginas: 85 - 89
- [20] Michael, K.; McCathie, L.; *The pros and cons of RFID in supply chain management*, Mobile Business, 2005. ICMB 2005. International Conference on 11-13 July 2005 pp:623 - 629
- [21] Taesung Kim; Howon Kim; *Access Control for Middleware in RFID Systems*, Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference, Volume 2, 20-22 Feb. 2006 pp1020 -1022
- [22] V. Daniel Hunt, Albert Puglia, Mike Puglia, *RFID A guide to radio frequency identification*. Ed. Wiley 2007.
- [23] The Missing Piece, Peter Winer, Frontline Solutions, July 1, 2004, [www.frontlinetoday.com](http://www.frontlinetoday.com)
- [24] U.S. Agency Clears Implantable Microchips, Barnaby J. Feder and Tom Zeller, Jr, *The New York Times*, October 15, 2004.
- [25] Myerson M. Judith, *RFID in the Supply Chain, A guide to Selection and Implementation*, Auerbach Publications 2007.
- [26] Sandip Lahiri, *RFID Sourcebook*, Prentice Hall PTR, August 31, 2005
- [27] [http://www.jasperforge.org/jaspersoft/opensource/business\\_intelligence/jasperreports](http://www.jasperforge.org/jaspersoft/opensource/business_intelligence/jasperreports)
- [28] SN65176B, SN75176B DIFFERENTIAL BUS TRANSCEIVERS, Texas Instruments data sheet.
- [29] <http://www.tiris.com>
- [30] JavaServerFaces, *Ajax and Flash: Next Generation User Interfaces*, An Oracle White Paper, October 2006

- [31] otn.oracle.com
- [32] *Developing Ajax-Based User Interfaces with JSF: An Introduction to ADF Faces Rich Client Components Page*, otn.oracle.com
- [33] Javier Eguíluz Pérez, *Introducción a AJAX*, www.librosweb.es
- [34] Matthias Hertel, *Aspects of AJAX*, Version 1.2 published 1. May 2007
- [35] *Oracle ADF 11g Primer, Introduction to building blocks of a Fusion Web application*, An Oracle White Paper, April 2007
- [36] Dirk Krafzig, Karl Banke, Dirk Slama, *Enterpsise SOA: Service-Oriented Architecture Best Practices*, Prentice Hall PTR, November, 2004
- [37] Erl Thomas, *Service-Oriented Architecture*, Prentice Hall , 2004
- [38] Harish Gaur, Markus Zirn, *BPEL Cookbook*, Ed. Packt, July 2006
- [39] Manual ,*Learn how BPEL Process Manager enables SOA*
- [40] Bruce Eckel, *Piensa en Java*, Ed. Prentice Hall, 4ta edición 2004
- [41] <http://es.wikipedia.org>
- [42] <http://www.sun.com>