

# **UNIVERSIDAD POLITÉCNICA SALESIANA**

**SEDE QUITO - CAMPUS SUR**

**CARRERA DE INGENIERIA EN SISTEMAS**

**MENCIÓN: TELEMÁTICA**

**TITULO DE LA TESIS DE GRADO**

**“ANÁLISIS, DISEÑO Y CONSTRUCCIÓN DE UN SISTEMA  
PROTOTIPO DE GESTIÓN DE VENTAS, CLIENTES, PERSONAL  
Y CONTROL DE STOCKS, BASADO EN TELEDETECCIÓN  
PARA “SOLVER PC” UTILIZANDO LA TECNOLOGÍA RFID”**

**TESIS PREVIA A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN  
SISTEMAS**

**ELABORADO POR:**

VINICIO ARTURO SALAS QUISPE

ALEX LENIN SALAZAR TONATO

**DIRECTOR DE TESIS:**

Ing. RAFAEL JAYA

**QUITO, Noviembre 2011**

## **DECLARACIÓN**

Nosotros, Vinicio Arturo Salas Quispe y Alex Lenin Salazar Tonato, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Universidad Politécnica Salesiana, según lo establecido por la Ley de Propiedad Intelectual, por su reglamento y por la normativa institucional vigente.

---

Vinicio Arturo Salas Quispe

---

Alex Lenin Salazar Tonato

## **CERTIFICACIÓN**

Certifico que el presente trabajo fue desarrollado por Vinicio Arturo Salas Quispe y Alex Lenin Salazar Tonato, bajo mi dirección.

---

Ing. Rafael Jaya  
Director de Tesis

## DEDICATORIA

*A mi Señor, Jesús, y mis virgencitas, quienes me dieron la fe, la fortaleza, la salud y la esperanza para terminar este trabajo.*

*A mis padres, Efraín y Gladys quienes me enseñaron desde pequeño a luchar para alcanzar mis metas, y nunca rendirme. Mi triunfo es el de ustedes, ¡los amo!*

*A mis queridas hermanas, a mis segundas madres Cristina y Shirley quienes me cuidaron y me apoyaron de mil formas, mientras realizaba mis estudios, ¡Gracias! Sin ustedes no hubiese podido hacer realidad este sueño.*

*A los que nunca dudaron que lograría este triunfo y siempre estuvieron pendiente de mi con su aliento y apoyo: a mi abuelita Elba y a mis 3 abuelitos que están en el cielo, a mis sobrinos, a mis tíos, a mis primos, a mi cuñado Santiago, a toda mi familia y a mis amigos de la universidad “Los de la salesiana Coneccion”, y del barrio “Los Elegidos”.*

*Vinicio “El Chula”*

*Quiero dedicar este trabajo a mi familia por acompañarme en cada una de mis éxitos que he emprendido y que siempre me alentaron.*

*A mis amados padres Ernesto y María quienes con mucho amor, esfuerzo, dedicación y enseñanzas han sido mi guía en todas las etapas de mi vida y con mucha comprensión y apoyo me convirtieron en el hombre que ahora soy para poder afrontar la vida y sus desafíos.*

*A mis hermanos Diego y Andrés que son mis mejores amigos, que junto a ellos cada día comparto mis sueños, alegrías, tristezas y éxitos en toda mi vida.*

*Alex “Dibu”*

## AGRADECIMIENTOS

*A Dios y mis virgencitas que me permitieron tener salud, fortaleza y estar con mis padres y mis hermanas en estos momentos de satisfacción para mí.*

*A mis dos familias los “Salas” y los “Quispe”, que siempre se preocuparon por mí y me apoyaron de diferentes maneras para lograr mis objetivos universitarios, estuvieron ahí cuando necesitaba ayuda y me dieron respaldo cuando las cosas iban mal.*

*A todos mis amigos en la universidad, que más que amigos fuimos como hermanos, siempre estuvimos juntos, en los estudios, y en cualquier parte fuera o dentro de la universidad nos dimos las mano, superamos problemas y siempre estaremos pendientes unos de otros por cualquier circunstancia que nos de la vida.*

*A los ingenieros que me impartieron cada una de las materias en mi vida universitaria, y estuvieron ahí para darme un consejo cuando lo necesite.*

*Por último agradezco de todo corazón al ingeniero “Rafael Jaya”, a quien le considero un amigo, que gracias a su ayuda pude cumplir mi objetivo universitario.*

*Vinicio Salas*

*A Dios por darme la salud, sabiduría y por llenarme de bendiciones para terminar una etapa más en mi vida.*

*A mis padres por ser mis amigos y compañeros que me han ayudado a crecer, gracias por estar siempre a mi lado en todo momento y por la paciencia que han tenido para enseñarme, por el amor que me brinda y por los regaños que me merecía y que nunca entendía.*

*En particular a mi Tutor quien me orientó durante toda la carrera y especialmente por sus consejos durante el tiempo que duró esta tesis.*

*Alex Salazar*

# CONTENIDO:

INTRODUCCIÓN.....	1
RESUMEN .....	3
PRESENTACIÓN .....	5
CAPÍTULO I.....	6
<b>1. ANTECEDENTES DEL PROYECTO .....</b>	<b>6</b>
1.1. PLANTEAMIENTO DEL PROBLEMA .....	7
1.2. OBJETIVOS DE LA INVESTIGACIÓN.....	8
1.2.1. Objetivo General .....	8
1.2.2. Objetivos Específicos .....	8
1.3. JUSTIFICACIÓN.....	9
1.4. ALCANCE DEL PROYECTO.....	11
1.4.1. Funcionalidades del sistema prototipo .....	11
1.4.2. Seguridad física .....	12
1.4.3. Seguridad lógica .....	12
1.4.4. Históricos del sistema prototipo .....	12
1.5. DESCRIPCIÓN GENERAL DEL SISTEMA.....	13
1.5.1. Gráficos de los módulos del sistema prototipo que trabaja con RFID .....	15
1.5.1.1. Módulo gestión de caja/ventas.....	15
1.5.1.2. Módulo gestión de empleados.....	16
1.5.1.3. Módulo monitoreo de robos .....	17
CAPÍTULO II .....	18
<b>2. SUSTENTO TEÓRICO DEL PROYECTO .....</b>	<b>18</b>
2.1. INGENIERÍA DE SOFTWARE .....	18
2.1.1. Definición .....	18
2.1.2. Etapas del proceso de la Ingeniería de Software.....	19
2.1.2.1. Análisis de requerimientos.....	19
2.1.2.2. Especificación .....	19
2.1.2.3. Arquitectura.....	19
2.1.2.4. Programación.....	20
2.1.2.5. Pruebas .....	20
2.1.2.6. Documentación .....	20
2.1.2.7. Mantenimiento.....	20
2.2. LENGUAJE DE MODELADO UNIFICADO UML.....	21
2.2.1. Fases del ciclo de desarrollo que soporta UML.....	21
2.3. TECNOLOGÍA RFID .....	22
2.3.1. Definición: .....	22
2.3.2. Arquitectura .....	24
2.3.3. Sistema RFID .....	24
2.3.3.1. Funcionamiento del sistema RFID.....	24
2.3.3.2. Componentes del sistema de funcionamiento RFID.....	25
2.3.3.3. Tags RFID .....	26
2.3.3.4. Lector RFID.....	28
2.3.3.5. Antena del Lector RFID .....	30
2.3.3.6. Controlador RFID .....	33
2.3.3.7. Conceptos Fundamentales RFID.....	33
2.3.4. El Espectro de Frecuencias RFID .....	34
2.3.5. Estandarización RFID.....	36
2.3.5.1. Norma ISO Para RFID .....	37
2.3.5.2. Norma EPC G2 (930 MHz) .....	39
2.3.6. Limitaciones de la tecnología RFID .....	40
2.4. TELEDETECCIÓN .....	40
2.5. PROTOTIPOS .....	41
2.5.1. Utilidad de los Prototipos.....	41
2.6. LENGUAJE DE PROGRAMACIÓN C# .....	42

2.6.1.	Características del lenguaje C#.....	43
2.6.1.1.	Ventajas y desventajas del lenguaje de programación C# .....	43
2.7.	EL SISTEMA OPERATIVO MICROSOFT WINDOWS .....	45
2.7.1.	Historia Cronológica de sistemas operativo Microsoft Windows .....	46
2.7.1.1.	Principales versiones de Windows .....	48
2.7.2.	Criticas y Polémicas.....	49
<b>CAPÍTULO III.....</b>		<b>50</b>
<b>3.</b>	<b>ANÁLISIS DEL PROYECTO .....</b>	<b>50</b>
3.1.	ESPECIFICACIÓN DE REQUISITOS .....	50
3.1.1.	Análisis de Factibilidad.....	50
3.2.	FACTIBILIDAD OPERACIONAL DEL PROYECTO .....	51
3.2.1.	Generalidades del Sistema .....	51
3.2.1.1.	Dificultad en la utilización del Sistema .....	51
3.2.1.2.	Operatividad del Sistema.....	52
3.2.1.3.	Formación en la utilización del Sistema .....	52
3.2.1.4.	Validez del Sistema.....	53
3.2.2.	Evaluación de la factibilidad .....	53
3.2.2.1.	Lenguajes de Programación .....	53
3.2.2.1.1.	Lenguaje de Programación PHP .....	53
3.2.2.1.1.1.	Descripción Lenguaje de Programación PHP.....	53
3.2.2.1.2.	Lenguaje de Programación Java .....	55
3.2.2.1.2.1.	Descripción del Lenguaje de Programación Java .....	55
3.2.2.1.3.	Lenguaje de Programación C# .....	56
3.2.2.1.3.1.	Descripción Del Lenguaje de Programación C# .....	56
3.2.2.1.4.	Calificación de la Factibilidad de los Lenguajes de Programación Analizados .....	58
3.2.2.1.4.1.	Resultado de la factibilidad sobre el lenguaje de programación a utilizar.....	59
3.2.2.2.	Bases de Datos.....	59
3.2.2.2.1.	Base de Datos SQL Server .....	59
3.2.2.2.1.1.	Descripción de la Base de Datos SQL Server.....	59
3.2.2.2.2.	Base de Datos Oracle .....	61
3.2.2.2.2.1.	Descripción de la Base de Datos Oracle .....	61
3.2.2.2.3.	Base de Datos MySQL Server .....	62
3.2.2.2.3.1.	Descripción de la Base de Datos MySQL Server .....	62
3.2.2.2.4.	Calificación de la Factibilidad de las Bases de Datos Analizadas .....	64
3.2.2.2.4.1.	Resultado de la factibilidad sobre la base de datos a utilizar .....	65
3.2.2.3.	Equipos RFID UHF de 860-960 MHz (Identificación por Radio Frecuencia).....	65
3.2.2.3.1.	Lector RFID Fijo de Red PoE A-GEN2 .....	65
3.2.2.3.1.1.	Descripción del Lector fijo de red PoE A-GEN2 .....	65
3.2.2.3.2.	Lector RFID Solutions Model U-RW-232 .....	67
3.2.2.3.2.1.	Descripción Del Lector RFID Solutions Model U-RW-232.....	67
3.2.2.3.3.	Lector RFID UHF Reader/Writer Versión Única Americana .....	69
3.2.2.3.3.1.	Descripción del Lector RFID UHF Reader/Writer versión única americana ..	69
3.2.2.3.4.	Calificación de la Factibilidad de los Lectores RFID Analizados .....	70
3.2.2.3.4.1.	Resultado de factibilidad sobre el lector RFID a utilizar .....	71
3.2.2.4.	Tags Pasivos RFID UHF de 860-960 MHz .....	72
3.2.2.4.1.	Tags Pasivos RFID Adhesivos .....	72
3.2.2.4.1.1.	Descripción de Tags Pasivos RFID adhesivos .....	72
3.2.2.4.2.	Tags Pasivos RFID de Plástico con Circuito .....	73
3.2.2.4.2.1.	Descripción de Tags de Plástico con circuito .....	73
3.2.2.4.3.	Tags Pasivos RFID de Mica Imprimibles .....	74
3.2.2.4.3.1.	Descripción de Tags Pasivos RFID de Mica Imprimibles.....	74
3.2.2.4.4.	Calificación de Criterios de los Tags RFID Analizados .....	75
3.2.2.4.4.1.	Resultado de factibilidad sobre el Tag RFID a utilizar .....	76
3.3.	FACTIBILIDAD TÉCNICA DEL PROYECTO.....	77
3.3.1.	Requisitos Técnicos de Equipos de Computación .....	77
3.3.2.	Requisitos Técnicos de Equipos RFID (Identificación por Radio Frecuencia).....	78
3.3.3.	Requisitos Técnicos de Tags RFID.....	79
3.3.4.	Argumentación Técnica .....	79
3.4.	FACTIBILIDAD ECONÓMICA DEL PROYECTO .....	80

3.4.1.	Análisis Costo / Beneficio.....	80
3.4.1.1.	Análisis de Resultados.....	82
3.4.1.2.	Punto de Equilibrio .....	82
3.4.1.3.	Periodo de Devolución .....	83
3.4.2.	Valor Actual Neto (VAN).....	84
3.4.3.	Tasa interna de Retorno (TIR).....	85
3.4.4.	Argumentación del Estudio Costo/Beneficio .....	86
<b>CAPÍTULO IV .....</b>		<b>88</b>
<b>4.</b>	<b>DISEÑO Y CONSTRUCCIÓN DEL SISTEMA PROTOTIPO.....</b>	<b>88</b>
4.1.	DISEÑO CONCEPTUAL .....	88
4.2.	DIAGRAMAS UML (LENGUAJE DE MODELADO UNIFICADO) .....	89
4.2.1.	Diagrama de clases .....	90
4.2.2.	Diagrama de caso de uso.....	91
4.2.3.	Diagrama de actividades .....	96
4.2.4.	Diagrama de secuencia.....	97
4.2.5.	Diagrama de componentes .....	98
4.3.	DISEÑO DE INTERFACES.....	99
4.4.	CREACIÓN DE MÓDULOS DEL SISTEMA RFIDBUSSINESSOFT .....	102
4.4.1.	Paquete Clases Usuario.....	103
4.4.2.	Paquete Clases Clientes .....	103
4.4.3.	Paquete Clases Empleados.....	103
4.4.4.	Paquete Clases Productos.....	104
4.4.5.	Paquete Clases Proveedores .....	104
4.4.6.	Paquete Clases RFID .....	105
4.4.7.	Paquete Clases Ventas .....	105
4.4.8.	Paquete Clases CMarcas Productos .....	105
4.4.9.	Paquete Clases Grupos Productos.....	106
4.4.10.	Paquete Casos Base de Datos.....	106
4.4.11.	Paquete Clases Roles.....	106
4.4.12.	Paquete Clases Seguridad Ventas.....	107
4.4.13.	Paquete Clases Validaciones.....	107
4.5.	CREACIÓN DE MÓDULOS DE GESTIÓN .....	108
4.6.	CREACIÓN DEL PAQUETE BASE DE DATOS .....	117
4.6.1.	Llamada a librerías en la clase BaseRFID.....	117
4.6.2.	Método Data set Consultar en la clase Base RFID .....	118
4.6.3.	Método Comandos en la clase BaseRFID .....	118
4.6.4.	Método varios en la Clase BaseRFID .....	119
4.6.4.1.	Método que crea la cadena de conexión a la Base de Datos .....	120
4.6.4.2.	Métodos de Extracción de la Clase BaseRFID .....	120
4.7.	CREACIÓN DEL PAQUETE CLASES RFID.....	121
4.7.1.	Clase Conversión de Datos.....	121
4.7.1.1.	Llamada a librerías en la Clase ConversionesDatos .....	121
4.7.1.2.	Creación de Métodos de Conversión en la Clase ConversionesDatos .....	122
4.7.2.	Clase Escritura RFID.....	123
4.7.2.1.	Llamadas a Librerías en la Clase EscrituraRFID .....	123
4.7.2.2.	Método Escribir Datos en la Clase EscrituraRFID.....	124
4.7.3.	Clase Lectura RFID .....	124
4.7.3.1.	Llamadas a Librería en la Clase LecturaRFID.....	125
4.7.3.2.	Método Lectura de Datos en la Clase LecturaRFID .....	125
4.7.4.	Clase Parámetros RFID.....	126
4.7.4.1.	Llamada a Librerías en la Clase ParametrosRFID .....	126
4.7.4.2.	Método Abrir Puerto en la Clase ParametrosRFID .....	126
4.7.4.3.	Método de Cerrar Puerto en la Clase ParametrosRFID .....	127
4.7.5.	Clase Verifica Estado.....	128
4.8.	CREACIÓN DE GESTIÓN DE EMPLEADOS .....	129
4.8.1.	Creación de la Clase Asignación de Tags .....	129
4.8.1.1.	Llamada a librerías en la Clase Asigna Tags Empleados.....	129
4.8.1.2.	Método de Asignación de Tags en la Clase Asignación Tags Empeados .....	130
4.8.2.	Creación de la Clase Ficha Control.....	131

4.8.2.1.	Llamadas a Librerías en la clase Ficha Control.....	131
4.8.2.2.	Método mostrar ingresos en la Clase Ficha Control.....	132
4.8.2.3.	Método Fecha del Sistema en la Clase Ficha Control .....	132
4.9.	CREACIÓN DE GESTIÓN DE VENTAS .....	133
4.9.1.	Clases Facturar .....	133
4.9.1.1.	Llamada a librerías en la Clase Facturar.....	133
4.9.1.2.	Método Cargar Factura en la Clase Facturar.....	134
4.9.1.3.	Método Cargar Factura Base en la Clase Facturar .....	134
4.9.2.	Clase Reporte Ventas.....	136
4.9.2.1.	Llamada a librerías de la Clase Reportes Ventas .....	136
4.9.2.2.	Métodos Impresión Factura en la Clase Reporte Ventas .....	137
4.9.3.	Clase Numero Letras .....	138
4.9.3.1.	Llamada a librerías de la Clase Numero Letras.....	138
4.9.3.2.	Método convertir de la Clase Números Letras .....	138
4.10.	CREACIÓN DE GESTIÓN DE USUARIOS .....	139
4.10.1.	Clase Asigna Permisos .....	139
4.10.1.1.	Llamadas a librerías de la Clase Asigna Permisos .....	139
4.10.1.2.	Método permisos en la Clase Asigna Permisos.....	139
4.10.2.	Clase Asigna Menú .....	140
4.10.2.1.	Llamadas a librerías de la Clase Asigna Menú .....	140
4.10.2.2.	Método Control Menú de la Clase Asigna Menú.....	141
4.10.3.	Clase Login Usuarios .....	141
4.10.3.1.	Llamadas a librerías de la Clase Login Usuario.....	141
4.10.3.2.	Método de validación de Usuario de la Clase Login Usuario .....	142
4.11.	CREACIÓN DE GESTIÓN DE PRODUCTOS .....	143
4.11.1.	Clase Tags .....	143
4.11.1.1.	Llamadas a librerías de la Clase Tags.....	143
4.11.1.2.	Método Genera tags de la Clase Tags.....	143
4.11.1.3.	Métodos escoger grilla Tags de la Clase Tags .....	144
4.11.1.4.	Método escoge código tag asignación de la Clase Tags .....	145
4.11.2.	Clase Reportes Productos .....	146
4.11.2.1.	Llamada a librerías de la Clase Reportes Productos.....	146
4.11.2.2.	Métodos Crear reporte de la Clase Reportes Productos.....	146
4.12.	CREACIÓN DE GESTIÓN DE ROBOS .....	147
4.12.1.	Clase Seguridad .....	147
4.12.2.	Llamada a Bibliotecas de la Clase Seguridad .....	147
4.12.3.	Método de Consulta tags asignado de la Clase Seguridad .....	148
4.12.4.	Método Comprueba Dato de la Clase Seguridad.....	148
<b>CAPÍTULO V</b> .....		<b>150</b>
<b>5. PRUEBAS Y RESULTADOS DEL SISTEMA PROTOTIPO “RFIDBUSSINESSOFT“</b> .....		<b>150</b>
5.1. PRUEBAS Y RESULTADOS DEL SISTEMA PROTOTIPO “RFIDBUSSINESOFT”.....		150
5.1.1. Conexión del Puerto serial del Lector RFID a la máquina donde trabaja el Sistema “RFIDBussinesSoft”.....		150
5.1.2. Ingreso al Sistema Prototipo “RFIDBussinesSoft” .....		152
5.1.3. Menú del Sistema Prototipo “RFIDBussinesSoft”.....		152
5.1.4. Gestión de Productos del Sistema Prototipo “RFIDBussinesSoft” .....		154
5.1.4.1. Control de Stock .....		154
5.1.4.2. Reportes y Estadísticas .....		160
5.1.5. Gestión de Clientes del Sistema Prototipo “RFIDBussinesSoft” .....		160
5.1.5.1. Control de Clientes.....		161
5.1.5.2. Reportes.....		162
5.1.6. Gestión de Empleados del Sistema Prototipo “RFIDBussinesSoft” .....		163
5.1.6.1. Gestión Empleados.....		163
5.1.6.2. Reportes.....		165
5.1.7. Gestión de Ventas del Sistema Prototipo “RFIDBussinesSoft” .....		166
5.1.7.1. Facturar .....		166
5.1.7.2. Detalles de Venta .....		168
5.1.8. Gestión de Usuarios del Sistema Prototipo “RFIDBussinesSoft” .....		168
5.1.8.1. Control de Usuarios.....		169

5.1.8.2.	Reportes.....	171
5.1.9.	Gestión de Proveedores del Sistema Prototipo “RFIDBussinesSoft” .....	171
5.1.9.1.	Control de Proveedores .....	172
5.1.9.2.	Reportes.....	173
5.1.10.	Gestión de Robos del Sistema Prototipo “RFIDBussinesSoft” .....	174
5.1.10.1.	Control Antirrobo .....	174
5.1.11.	Control de Empleados del Sistema Prototipo “RFIDBussinesSoft” .....	176
5.1.12.	Cambiar usuarios del Sistema Prototipo “RFIDBussinesSoft” .....	177
<b>CONCLUSIONES Y RECOMENDACIONES.....</b>		<b>179</b>
CONCLUSIONES: .....		179
RECOMENDACIONES .....		182
<b>BIBLIOGRAFÍA.....</b>		<b>185</b>
LIBROS:.....		185
FORMATOS PDF: .....		186
PÁGINAS WEB: .....		187
<b>ANEXOS.....</b>		<b>189</b>
ANEXO 1: DISEÑO CONCEPTUAL DE LA BASE DE DATOS “RFIDBUSSINESOFT” .....		190
ANEXO 2: DIAGRAMA DE CLASES DE “RFIDBUSSINESOFT” .....		191
ANEXO 3: DIAGRAMA DE CASO DE USO DE “RFIDBUSSINESOFT” .....		192
ANEXO 4: DIAGRAMA DE ACTIVIDADES DE “RFIDBUSSINESOFT” .....		193
ANEXO 5: DIAGRAMA DE SECUENCIA DE “RFIDBUSSINESOFT” .....		194
ANEXO 6: DIAGRAMA DE COMPONENTES DE “RFIDBUSSINESOFT” .....		195
ANEXO 7: MANUAL DE USUARIO DE “RFIDBUSSINESOFT” .....		196

# ÍNDICE DE FIGURAS:

Figura 1: “Módulo gestión caja/ventas”	15
Figura 2: “Módulo gestión de empleados”	16
Figura 3: “Módulo monitoreo de robos”	17
Figura 4: “Ingeniería de Software”	18
Figura 5: “Identificación por Radio Frecuencia”	23
Figura 6: “Arquitectura RFID”	24
Figura 7: “Componentes del sistema de funcionamiento RFID”	26
Figura 8: “Tags RFID”	27
Figura 9: “Lector RFID”	30
Figura 10: “Rango Antena Lector RFID”	31
Figura 11: “Antena Polarizada Linealmente”	32
Figura 12: “Antena Polarizada Circularmente”	32
Figura 13: “Ondas Electromagneticas”	34
Figura 14: “Espectro RFID”	34
Figura 15: “Teledetección con RFID”	41
Figura 16: “Lenguaje de Programación# “	42
Figura 17: “Logo de Windows “	45
Figura 18: “Cronología de Microsoft Windows “	48
Figura 19: “Logo del lenguaje de programación PHP “	54
Figura 20: Logo del lenguaje de la tecnología JAVA	55
Figura 21: “Logo del lenguaje de lenguaje de programación C#”	57
Figura 22: “Logo de Base de Datos SQL Server”	60
Figura 23: “Logo de Base de Datos ORACLE”	61
Figura 24: “Logo de Base de Datos MySQL“	63
Figura 25: “Lector RFID PoE A-GEN2”	66
Figura 26: “Lector RFID Modelo U-RW-232”	68
Figura 27: “Lector RFID UHF READER/WRITER “	69
Figura 28: “Tags RFID Pasivos adhesivo “	72
Figura 29: “Tags RFID Pasivos de Pastico con circuito”	73
Figura 31: “Estadística de “Sistema Prototipo” en tres años”	87
Figura 32: “Modelo Conceptual “RFIDBussinesSoft” desarrollado en DBDesigner4.0.5.6”	89
Figura 33: “Diagrama de Clases “RFIDBussinesSoft” desarrollado en Enterprise Architect 7.5”	90
Figura 34: “Diagrama de Caso de Uso “RFIDBussinesSoft” desarrollado en Enterprise Architect 7.5”	95
Figura 35: “Diagrama de Actividades “RFIDBussinesSoft” desarrollado en Enterprise Architect 7.5”	96
Figura 36: “Diagrama de Secuencia “RFIDBussinesSoft” desarrollado en Enterprise Architect 7.5”	97
Figura 37: “Diagrama de Componentes “RFIDBussinesSoft” desarrollado en Enterprise Architect 7.5”	98
Figura 38: “Presentación de “RFIDBussinesSoft” ”	99
Figura 39: “Ingreso a “RFIDBussinesSoft” ”	100
Figura 40: “Menú de “RFIDBussinesSoft” ”	100
Figura 41: “Interfaz Estándar de Trabajo y Operaciones de “RFIDBussinesSoft” ”	101
Figura 42: “Paquetes o Carpetas de “RFIDBussinesSoft” ”	102
Figura 43: “Paquetes o Carpetas Clases_Usuarios de “RFIDBussinesSoft” ”	103
Figura 44: “Paquetes o Carpetas Clases Clientes de “RFIDBussinesSoft” ”	103
Figura 45: “Paquetes o Carpetas Clases Empleados de “RFIDBussinesSoft” ”	104
Figura 46: “Paquetes o Carpetas Clases Productos de “RFIDBussinesSoft” ”	104
Figura 47: “Paquetes o Carpetas Clases Proveedores de “RFIDBussinesSoft” ”	104
Figura 48: “Paquetes o Carpetas Clases RFID de “RFIDBussinesSoft” ”	105
Figura 49: “Paquetes o Carpetas Clases Ventas de “RFIDBussinesSoft” ”	105
Figura 50: “Paquetes o Carpetas Clases CMarcas Productos de “RFIDBussinesSoft” ”	106
Figura 51: “Paquetes o Carpetas Clases Grupos Productos de “RFIDBussinesSoft” ”	106
Figura 52: “Paquetes o Carpetas Clases Base de Datos de “RFIDBussinesSoft” ”	106
Figura 53: “Paquetes o Carpetas Clases Roles de “RFIDBussinesSoft” ”	107
Figura 54: “Paquetes o Carpetas Clases Seguridad Ventas de “RFIDBussinesSoft” ”	107
Figura 55: “Paquetes o Carpetas Clases Seguridad Ventas de “RFIDBussinesSoft” ”	107
Figura 56: “Carpetas de Referencias para drivers de “RFIDBussinesSoft” ”	107
Figura 57: “Código C# de la clase “Producto_Datos” métodos de ingreso Set y Get	109
Figura 58: “Código C# de invocación a otras clases desde la clase “Producto_Datos”	110

Figura 59: “Código C# Instancia de la clase de Ingreso “Producto_Datos” .....	110
Figura 60: “Código C# Creación del método Ingreso_Productos en la clase “Producto_Datos” .....	111
Figura 61: “Código C# Creación del método Actualizar_Productos en la clase “Producto_Datos” .....	112
Figura 62: “Código C# Creación de la clase “ConsultaProductos” y el método de selección .....	113
Figura 63: “Código C# Creación del método Set y get Criterios de la clase “ConsultaProductos” .....	113
Figura 64: “Código C# Creación del método estable consulta de la clase “ConsultaProductos” .....	114
Figura 65: “Código C# Creación del método Estable consulta de la clase “ConsultaProductos” .....	114
Figura 66: “Código C# Instancia de la clase de consulta “ConsultaProductos” .....	115
Figura 67: “Código C# Llamada de métodos de ingreso de Productos en sus respectivo botón .....	115
Figura 68: “Código C# Llamada de métodos de actualización de Productos en su respectivo botón .....	116
Figura 69: “Código C# Llamada a librerías en la Clase BaseRFID .....	117
Figura 70: “Código C# Creación del método DataSet Consultar en la Clase BaseRFID .....	118
Figura 71: “Código C# Creación del método Consultar en la Clase BaseRFID .....	119
Figura 72: “Código C# Creación del método crea cadena en la Clase BaseRFID .....	120
Figura 73: “Código C# Creación del método de extracción en la Clase BaseRFID .....	120
Figura 74: “Código C# Librerías Utilizadas en la clase ConversionesDatos .....	121
Figura 75: “Código C# Creación del método de conversión cadena a hexadecimal .....	122
Figura 76: “Código C# Creación del método de conversión de bytes a hexadecimal .....	123
Figura 77: “Código C# Librerías utilizadas en la clase EscrituraRFID .....	123
Figura 78: “Código C# creación del método escribir datos en la clase EscrituraRFID .....	124
Figura 79: “Código C# Librerías utilizadas en la clase LecturaRFID .....	125
Figura 80: “Código C# creación del método lecturas de datos en la clase LecturaRFID .....	125
Figura 81: “Código C# llamada a librerías en la clase LecturaRFID .....	126
Figura 82: “Código C# creación del método abrir puerto en la clase ParametrosRFID .....	127
Figura 83: “Código C# creación del método abrir puerto de forma automática .....	127
Figura 84: “Código C# creación del método cerrar puerto en la clase ParametrosRFID .....	128
Figura 85: “Código C# Llamada de configuraciones RFID en el botón Activar RFID .....	128
Figura 86: “Código C# creación de la clase verificar estado y el método cargar parámetros .....	129
Figura 87: “Código C# llamada de librerías en la clase Asigna Tags empleados .....	130
Figura 88: “Código C# Creación de método Tags asignación en la clase Asigna Tags empleados .....	130
Figura 89: “Código C# Llamada de asignación de tags RFID en el botón Asignar Tags.....	131
Figura 90: “Código C# llamada de librerías en la clase Ficha Control .....	131
Figura 91: “Código C# Creación de mostrar ingresos en la clase Ficha control.....	132
Figura 92: “Código C# Creación del método Fecha del sistema en la clase Ficha control .....	133
Figura 93: “Código C# llamada de librerías en la clase Facturar.....	134
Figura 94: “Código C# Creación del método cargar factura de la clase facturar.....	134
Figura 95: “Código C# Creación del método cargar factura base de la clase facturar d .....	135
Figura 96: “Código C# Llamada a métodos de facturación desde el botón Facturar .....	136
Figura 97: “Código C# llamada de librerías en la clase Reportes Ventas .....	137
Figura 98: “Código C# método de imprimir facturas en la clase Reportes Ventas.....	137
Figura 99: “Código C# Llamada librerías de la clase Números Letras .....	138
Figura 100: “Código C# método convertir de la clase Números Letras.....	138
Figura 101: “Código C# Llamadas a librerías De la clase Asigna Permisos .....	139
Figura 102: “Código C# método permisos de la clase Asigna Permisos .....	140
Figura 103: “Código C# Llamada a librerías de la clase Asigna Menú .....	140
Figura 104: “Código C# Método control menú de la clase Asigna Menú.....	141
Figura 105: “Código C# Llamada a librerías de la clase Login Usuario.....	142
Figura 106: “Código C# Método valida usuario de la clase Login Usuario .....	142
Figura 107: “Código C# Llamada a librerías de la clase Tags de .....	143
Figura 108: “Código C# Método genera tag de la clase Tags de .....	144
Figura 109: “Código C# Método escoger grilla de la clase Tags de .....	145
Figura 110: “Código C# Método escoger cod tag y tags asignados de la clase Tags d.....	145
Figura 111: “Código C# Llamada a librerías de la clase Reporte Productos .....	146
Figura 112: “Código C# Método Crear reporte de la clase Reportes Productos.....	147
Figura 113: “Código C# Llamada a librerías de la clase Seguridad .....	148
Figura 114: “Código C# Método consulta tag asignados de la clase Seguridad .....	148
Figura 115: “Código C# Método comprueba dato de la clase Seguridad .....	149
Figura 116: “Conectores “Lector RFID”” .....	151
Figura 117: “Pantalla de ingreso de “RFIDBussinesSoft”” .....	152
Figura 118: “Pantalla del Menú de “RFIDBussinesSoft”” .....	153

Figura 119: “Pantalla del Menú de Usuario Guardia “RFIDBussinesSoft”” .....	153
Figura 120: “Pantalla del Menú del Sistema opciones Gestión de Productos “RFIDBussinesSoft”” .....	154
Figura 121: “Pantalla Ingreso Actualización de Gestión Productos “RFIDBussinesSoft”” .....	155
Figura 122: “Pantalla Ingreso/Actualización Gestión de Productos “RFIDBussinesSoft”” .....	155
Figura 123: “Pantalla Ingreso de Marca Gestión de Productos “RFIDBussinesSoft”” .....	156
Figura 124: “Pantalla Ingreso de Grupo Gestión de Productos “RFIDBussinesSoft”” .....	156
Figura 125: “Pantalla Ingreso mensaje de ingreso Gestión de Productos “RFIDBussinesSoft”” .....	157
Figura 126: “Pantalla consulta de productos Gestión de Productos “RFIDBussinesSoft”” .....	157
Figura 127: “Pantalla Asignación de Tags Gestión de Productos “RFIDBussinesSoft”” .....	158
Figura 128: “Pantalla Abrir puertos seriales Gestión de Productos “RFIDBussinesSoft”” .....	158
Figura 129: “Pantalla información y parámetros RFID Gestión de Productos “RFIDBussinesSoft”” .....	159
Figura 130: “Asignación Tags RFID a producto Gestión de Productos “RFIDBussinesSoft”” .....	159
Figura 131: “Reportes de producto Gestión de Productos “RFIDBussinesSoft”” .....	160
Figura 132: “Pantalla del Menú del Sistema opciones Gestión de Clientes “RFIDBussinesSoft”” .....	161
Figura 133: “Pantalla Ingreso mensaje de ingreso Gestión de Clientes “RFIDBussinesSoft”” .....	161
Figura 134: “Pantalla Consulta clientes Gestión de Clientes “RFIDBussinesSoft”” .....	162
Figura 135: “Reporte clientes Gestión de Clientes “RFIDBussinesSoft”” .....	162
Figura 136: “Pantalla del Menú del Sistema opciones Gestión de Empleados “RFIDBussinesSoft”” .....	163
Figura 137: “Pantalla Ingreso mensaje de ingreso Gestión de Empleados “RFIDBussinesSoft”” .....	164
Figura 138: “Pantalla Consulta empleados Gestión de Empleados “RFIDBussinesSoft”” .....	164
Figura 139: “Pantalla asigna tags a empleados Gestión de Empleados “RFIDBussinesSoft”” .....	165
Figura 140: “Reporte clientes Gestión de Clientes “RFIDBussinesSoft”” .....	165
Figura 141: “Pantalla del Menú del Sistema opciones Gestión de Ventas “RFIDBussinesSoft”” .....	166
Figura 142: “Pantalla de Facturación Gestión de Ventas “RFIDBussinesSoft”” .....	167
Figura 143: “Pantalla de Facturando por venta Gestión de Ventas “RFIDBussinesSoft”” .....	167
Figura 144: “Factura por venta Gestión de Ventas “RFIDBussinesSoft”” .....	168
Figura 145: “Pantalla del Menú del Sistema opciones Gestión de Usuarios “RFIDBussinesSoft”” .....	169
Figura 146: “Pantalla Ingreso de usuarios Gestión de Usuarios “RFIDBussinesSoft”” .....	169
Figura 147: “Pantalla asignación de permisos a usuarios Gestión de Usuarios “RFIDBussinesSoft”” .....	170
Figura 148: “Pantalla asignando permisos a usuarios Gestión de Usuarios “RFIDBussinesSoft”” .....	170
Figura 149: “Reporte usuarios Gestión de Usuarios “RFIDBussinesSoft”” .....	171
Figura 150: “Pantalla del Menú del Sistema opciones Gestión de Proveedores “RFIDBussinesSoft”” .....	172
Figura 151: “Pantalla Ingreso de proveedores Gestión de Proveedores “RFIDBussinesSoft”” .....	172
Figura 152: “Pantalla Consulta proveedores Gestión de Proveedores “RFIDBussinesSoft”” .....	173
Figura 153: “Reporte proveedores Gestión de Proveedores “RFIDBussinesSoft”” .....	173
Figura 154: “Pantalla del Menú del Sistema opciones Gestión de Robos “RFIDBussinesSoft”” .....	174
Figura 155: “Pantalla Anti robo Gestión de Robos “RFIDBussinesSoft”” .....	175
Figura 156: “Pantalla Anti robo velicación de productos Gestión de Robos “RFIDBussinesSoft”” .....	175
Figura 157: “Pantalla del Menú del Sistema opciones Control Empleados “RFIDBussinesSoft”” .....	176
Figura 158: “Pantalla Control de entrada empleados Control Empleados “RFIDBussinesSoft”” .....	176
Figura 159: “Pantalla Control de salida empleados Control Empleados “RFIDBussinesSoft”” .....	177
Figura 160: “Pantalla del Menú del Sistema opciones Control Empleados “RFIDBussinesSoft”” .....	177
Figura 161: “Pantalla de ingreso usuario en Cambiar Usuario “RFIDBussinesSoft”” .....	178

# ÍNDICE DE TABLAS:

Tabla 1: “Potencia de emisión RFID” .....	35
Tabla 2: “Versiones ISO 18000” .....	37
Tabla 3: “ventajas y desventajas de C#” .....	44
Tabla 4: “Principales ventajas y desventajas de PHP” .....	54
Tabla 5: “Principales ventajas y desventajas de JAVA” .....	56
Tabla 6: “Principales ventajas y desventajas de C#” .....	58
Tabla 7: “Tabla de resultados para la elección del lenguajes de Programación” .....	59
Tabla 8: “Principales ventajas y desventajas de SQL Server” .....	61
Tabla 9: “Principales ventajas y desventajas de SQL Server” .....	62
Tabla 10: “Principales ventajas y desventajas de MySql Server” .....	64
Tabla 11: “Tabla de resultados para la elección de la Base de Datos” .....	64
Tabla 12: “Principales ventajas y desventajas de Lector RFID fijo de red PoE A-GEN2” .....	67
Tabla 13: “Principales ventajas y desventajas de Lector RFID Modelo U-RW-232 .....	68
Tabla 14: “Principales ventajas y desventajas de Lector RFID UHF READER/WRITER” .....	70
Tabla 15: “Tabla de resultados para la elección del lector RFID” .....	71
Tabla 16: “Principales ventajas y desventajas de Tags Adhesivos RFID” .....	73
Tabla 17: “Principales ventajas y desventajas de Tags pasivos RFID de Plástico con circuito” .....	74
Tabla 18: “Principales ventajas y desventajas de Tags Pasivos RFID de Mica” .....	75
Tabla 19: “Tabla de resultados para la elección de tags pasivos RFID” .....	76
Tabla 20: “Costo/Beneficio del Sistema Prototipo” .....	81
Tabla 21: “Punto de equilibrio del Sistema Prototipo” .....	82
Tabla 22: “Periodo de Devolución del Sistema Prototipo” .....	83
Tabla 23: “Flujo de caja neto en 3 años Sistema Prototipo” .....	84
Tabla 24: “VAN para Sistema Prototipo” .....	85
Tabla 25: TIR para “Sistema Prototipo” .....	86
Tabla 26: Código de Colores para la “Configuración de Lector RFID” .....	151

# INTRODUCCIÓN

La tecnología RFID (identificación por radio frecuencia), ha logrado en los últimos años un importante espacio en la tecnología debido a la relativa reducción de precios en el mercado, al desarrollo de aplicaciones como también al crecimiento en sus capacidades y a las ventajas que ofrece con diversas tecnologías de auto identificación por medio de teledetección.

RFID plantea revolucionar el estilo de vida de la gente en varias aplicaciones de la vida cotidiana. En el caso de estar implicado en la cadena de producción y distribución de productos de las fábricas, generará grandes utilidades y beneficios tales como: verificación de los productos, otro caso es el cobro automático de peajes en las grandes ciudades, control de tiempo de parqueadero en hoteles o departamentos de transporte.

Finalmente, otras aplicaciones se encuentran en las lavadoras que identificarán el ciclo de lavado que le corresponde a determinado tipo de ropa o 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 la actualidad las aplicaciones que trabajan a la par con esta tecnología son prometedoras, pero todavía, en Sur América no son de uso masivo por lo que se encuentra aún en proceso de evolución.

En este trabajo de tesis, se analizan las ventajas que ofrece la tecnología RFID frente a otras tecnologías. Se da a conocer los componentes que participan en esta tesis pese a que en esta ocasión se orientó a la gestión de ventas, clientes, personal y control de stock, la mayoría de los elementos estudiados se adaptan para múltiples casos.

Se analizó el período completo de este tipo de sistemas, desde cómo se

genera información a partir de los lectores de RFID; hasta el procesamiento de los datos, a partir de diversas aplicaciones. Finalmente, se muestra el funcionamiento en un esquema de este tipo, de esta manera esta tesis pretende ser una guía de los elementos a estudiar en un proyecto de teledetección utilizando tag de identificación con la tecnología RFID (identificación por radio frecuencia) con frecuencias UHF de 860-960 MHz aproximadamente, que soporte la norma EPC G2 UHF o ISO18000-6 tag para locales comerciales u otras aplicaciones.

## RESUMEN

Para la elaboración del sistema prototipo, se encontró varios problemas, uno de estos fue al momento de la elección de los equipos RFID, ya que en nuestro país los lectores RFID de largo alcance son escasos, por lo que Importamos los equipos RFID desde China.

La elaboración y construcción de un sistema que faculte disponer de una nueva, rápida y eficaz facturación automática, es en la actualidad el propósito que desean lograr todas las empresas, tiendas, locales comerciales, etc. que realizan ventas, aprovechando la tecnología avanzada de los últimos tiempos, es así que al comprender como es el modo que funciona los sistema de facturación con sistemas tecnológicos, nos permitirá identificar los principales inconvenientes y limitaciones encontradas, de manera que se pueda obtener por medio de otra tecnología un sistema que sea cual sea la cantidad de productos facture de forma automática y rápida.

A través de este estudio, se pretende elaborar el análisis, diseño y construcción de un software prototipo que ayude a comprender el funcionamiento de un sistema donde por medio de la tecnología RFID realice diversas gestiones como: Gestión de Ventas, Clientes, Personal y Control de Stocks.

El desarrollo de este trabajo se compone de cinco capítulos, en donde catalogamos paso a paso todo acerca de la tecnología RFID y la ingeniería de software para la fabricación del sistema formulado.

De esta forma en el capítulo I, nombrado “Antecedentes del Proyecto”, se describe la problemática encontrada, la misma que será el principio para dar a conocer la construcción de este proyecto, así como los objetivos que se procura conseguir, también mencionamos la tendencia, las dirección y los rasgo característicos que se pretende entender, además de un detalle específico del software a construir.

En el capítulo II, nombrado “Sustento Teórico del Proyecto”, se reunirán las principales teorías que serán la base primordial de la construcción de este proyecto, ya que se tendrá las fuentes necesarias que permitirán notar que es lo que se desea alcanzar.

Capítulo III, nombrado “Análisis del Proyecto”, se elaborará el análisis de factibilidad para verificar que tan viable es la construcción del sistema, a nivel operativo, técnico y económico, así como de las herramientas necesarias para la creación del mismo, como los lenguajes de programación, bases de datos y equipos RFID.

En el Capítulo IV, nombrado “Diseño y Construcción del Sistema Prototipo”, se crean diversos diagramas UML, que facultarán la estructura que poseerá el Sistema Prototipo, como también las principales clases y métodos elaborados en el lenguaje de programación C# , para la respectiva construcción del sistema.

El capítulo V, nombrado “Pruebas y Resultados del Sistema Prototipo “RFIDBussinesSoft”, permitirá conocer como es el funcionamiento del sistema, y las diferentes gestiones que trabaja, para verificar el funcionamiento esperado.

De esta manera, se alcanzarán las pertinentes conclusiones y recomendaciones pertinentes según la construcción del sistema en cada etapa de su elaboración.

## **PRESENTACIÓN**

La elaboración del presente proyecto tiene como propósito, el análisis, Diseño y Construcción de un Sistema prototipo de gestión de ventas, clientes, personal y control de stocks, utilizando la tecnología RFID (Identificación por Radio Frecuencia), a través del cual se entenderá como es la estructura lógica de funcionamiento en lo que se refiere a facturación, control de horarios y tentativas de robos.

La construcción del sistema formulado, facultará que los usuarios puedan manejar facturaciones automáticas y rápidas, registros de horarios como también de tentativas de robo, por medio de secciones y permisos de usuario.

La creación de este sistema prototipo, permitirá otorgar permisos de uso a usuarios según su rol, de esta manera podrán comprender cuáles son sus funciones dentro del sistema, a través de una interfaz gráfica accesible a los usuarios.

Es así como este proyecto, es la consecuencia de la investigación desarrollada y de los conocimientos adquiridos en toda nuestra carrera universitaria que en complemento, permitirán obtener los resultados deseados en nuestro sistema final.

# CAPÍTULO I

## 1. ANTECEDENTES DEL PROYECTO

En este capítulo, se hará alusión al problema identificado, el mismo que será el fundamento principal para la construcción del actual proyecto, igualmente se pretende dar solución por medio de objetivos claros, concretos y alcanzables, que permitan justificar el proyecto y la investigación que se efectuará, como también tener un clara prospectiva del sistema a elaborarse en toda la investigación antes citada.

En el transcurso de los años han surgido distintas tecnologías de teledetección por medio de auto identificación. Entre sus diversas aplicaciones, podemos mencionar la gestión de locales comerciales en donde abarca gestión de ventas, clientes, personal y control de stock.

La automatización en ventas el control de stock y empleados, gestionada en la actualidad por sus respectivas entidades administrativas, genera problemas en el adecuado pago rápido, ágil y automático en caja como la elaboración de inventarios automáticos para controlar el stock, también la gestión en los horarios de trabajo que los empleados tienen establecidos y un monitoreo constante de robos en sus instalaciones.

En muchos 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 o en otros lugares no tienen ningún tipo de infraestructura. Debido a esto, surge la necesidad de desarrollar alternativas que permitan resolver el problema de gestión de ventas, clientes, personal y control de stock, haciendo uso de la tecnología actual, para así aprovechar al máximo sus capacidades.

## **1.1. PLANTEAMIENTO DEL PROBLEMA**

La creciente demanda de ventas de computadoras y sus accesorios para empresas, negocios, el hogar, y la necesidad de un rápido como también un ágil pago de sus compras por parte de los clientes para evitar grandes colas, pérdida de tiempo en caja, además la necesidad de un control de stock de productos, registro de empleados y clientes afiliados, obligan a buscar nuevos e innovadores métodos para la gestión de planta mediante el desarrollo de software, equipos tecnológicos, etc., existe una preocupación de no contar con una efectiva y rápida atención al cliente, un control diario de stock de productos, registro de empleados como también clientes afiliados, teniendo presente que pueda ocurrir cualquier eventualidad de robo o sustracción de productos o clientes insatisfechos por el servicio.

Si bien es cierto en los Locales Comerciales el sistema de facturación de productos no es ágil, ni automatizado, tampoco tienen un control exacto de stock de productos, ni gestión de empleados y de clientes, debido a causas como una escasa apertura administrativa y de infraestructura tecnológica por parte de los directivos, en la actualidad se buscan varias formas de tener altos niveles de agilidad de facturación, control de stock de productos, gestión de empleados y clientes.

Por estas razones existen consecuencias que se presentan como un problema para una gestión completa y automática en ventas, stock de productos, gestión de empleados y clientes, por no contar con un sistema y un dispositivo tecnológico adecuado.

En este sentido, la formalidad de contar con un sistema prototipo para la gestión de locales comerciales con sus respectivos módulos para Facturación, Control de stock de productos, Gestión de empleados y gestión de clientes mediante teledetección usando la señal de la tecnología RFID con frecuencia UHF que su radio mida cerca de 3 a 5 metros, así se tendrá una mayor optimización en facturación, control de stock de productos, mayor rendimiento

de los empleados y satisfacción en los clientes, como también menos vulnerabilidad al momento de una tentativa de robo, para así mantener altos niveles de seguridad en el control de sus productos y clientes en general.

## **1.2. OBJETIVOS DE LA INVESTIGACIÓN**

### **1.2.1. Objetivo General**

Investigar, desarrollar y construir un sistema prototipo orientado a la gestión de Locales Comerciales por medio de módulos que optimizarán los pagos por ventas, control de stock de productos, gestión de empleados y clientes, basado en teledetección utilizando tag de identificación con la tecnología RFID (identificación por radio frecuencia) con frecuencias UHF de 860-960 MHz, que soporte la norma EPC G2 o ISO18000-6B y así brindar un cómodo y excelente servicio a los clientes.

### **1.2.2. Objetivos Específicos**

- Estudiar y analizar los métodos, técnicas y herramientas para desarrollar un sistema prototipo con Tecnología RFID de frecuencia UHF.
- Escoger el Equipo RFID mas óptimo que cumpla con el rango de cobertura requerido, la frecuencia UHF adecuada y el estándar EPC G2 o ISO18000-6B para trabajar en el sistema a desarrollar.
- Escoger los tags de identificación más apropiados, que cumplan con el estándar EPC G2 o ISO18000-6B y trabajen con frecuencia UHF para utilizar en el sistema a desarrollar.
- Diseñar y crear una interfaz gráfica según el usuario del sistema.
- Diseñar y crear una base de datos para el sistema, que permita

manipular la información según las necesidades de los usuarios.

- Agilitar y perfeccionar el tiempo de facturación en caja por ventas.
- Tener un control completo de stock de productos para generar reportes.
- Controlar y detectar el hurto de los productos por medio del sistema a la salida del Local Comercial.
- Gestionar horarios de entrada y salida, de los empleados en sus lugares de trabajo del Local Comercial.

### **1.3. JUSTIFICACIÓN**

Los Locales Comerciales no poseen ningún sistema prototipo de gestión de optimización de ventas, gestión de empleados (horas de entrada y salida) en sus puestos de trabajo, registro de clientes afiliados y no afiliados, control de stock de productos, por lo que han tenido serios inconvenientes, como pérdidas económicas, malestar en su clientela, etc.

Por lo antes mencionado existe la necesidad de construir un sistema prototipo, que permita optimizar, controlar y registrar los productos, gestionar a los empleados y clientes basándonos en normas para desarrollar un sistema de alta calidad mediante teledetección, con la ayuda de la señal de la tecnología RFID UHF de mediano alcance basado en las normas EPC G2 o ISO18000-6B.

Si bien es cierto el presente análisis, diseño y construcción de un sistema prototipo de gestión de planta, no pretende dar una solución definitiva al problema planteado, porque siempre van haber nuevas e innovadoras formas de optimizar tiempos en ventas, registrar a los empleados y clientes en general y evitar el robo de productos de locales comerciales, pero se tratara de lograr

optimizar tiempos en ventas, reduciendo y agilizando las facturaciones por medio de los módulos antes mencionados.

Consideremos que la justificación del sistema prototipo se basa en:

- La prevención de tentativa de robo o hurto, al usar el sistema prototipo de control se podría hacer desistir a la persona que pretendía sustraerse un producto del local comercial.
- Agilizar la facturación por ventas, ahorro de tiempo, recursos tanto al cliente como a la empresa.
- Alta calidad de funcionamiento del sistema prototipo y alta calidad tecnológica para el control y registro de la información a manejar.
- Optar por el uso de software libre o licenciado según lo requiera las necesidades del equipo RFID, pero se tratará en la mayoría de nuestras herramientas para la construcción del sistema prototipo, de usar herramientas libres para reducir costos y problemas referentes al uso de licencias.

Y finalmente se puede mencionar que la recopilación de información para el análisis, diseño y construcción a realizar respecto a la gestión de locales comerciales, se circunscriba verificar los procedimientos metodológicos que se tienen que seguir hasta alcanzar una efectividad en este sistema prototipo, que se construirá para el local comercial de "SOLVER PC" para ofrecer un mejor servicio y una gran seguridad.

## **1.4. ALCANCE DEL PROYECTO**

El alcance del proyecto es analizar, diseñar y construir un sistema prototipo que trabaje sobre la plataforma Windows, basado en gestión de planta de un local comercial con módulos para optimizar ventas, control de stock de productos y la gestión de empleados (horas de entrada y salida) como también la gestión de clientes, para así brindar un rápido y ágil servicio.

Éste sistema prototipo se puede construir para locales comerciales, almacenes y supermercados medianos o pequeños.

El sistema prototipo, se realizará bajo un lenguaje de programación orientado a objetos con herramientas de software libre o software licenciado dependiendo de los requerimientos del equipo RFID a trabajar, también por su versatilidad, facilidad de manipulación, y soporte en la elaboración y ejecución del sistema, además contara con puertos seriales RJ-45 o RJ-232, USB versión 2, TCP/IP para la comunicación con RFID con la Norma EPC G2 o ISO18000-6B.

### **1.4.1. Funcionalidades del sistema prototipo**

- Gestión de planta de un local comercial por medio de módulos.
- Optimización de ventas al momento de pago de compras de productos por parte de los clientes.
- Facturación rápida y ágil.
- Control de stock de productos del local comercial y robos de productos.
- Gestión de empleados de horarios de entrada y salida en sus lugares de trabajo.
- Gestión de clientes.
- Mostrar y registrar reportes estadísticos de stock de productos.
- Mostrar reportes de empleados y clientes según las necesidades.

El sistema prototipo de gestión está comprendido por los siguientes servicios:

#### **1.4.2. Seguridad física**

- Colocación de tags en los productos y credenciales de los empleados del local comercial.
- Prevención de fallo de suministro eléctrico mediante el uso de UPS, conectados al servidor principal donde trabajará la aplicación, para evitar la deshabilitación total del equipo RFID mientras se soluciona el problema eléctrico.
- Notificación de eventualidades mediante el sistema, en tiempo real o por el servicio de correo electrónico, en el acto o cada cierto tiempo determinado por el usuario encargado.
- Control de sistema de gestión de planta de un local comercial.

#### **1.4.3. Seguridad lógica**

##### ✓ **En el Sistema Prototipo**

- Control de usuarios mediante el uso de sesiones (Autenticación, Autorización) para garantizar la identidad del usuario.

##### ✓ **En el Control**

- Clave de ingreso para el ingreso y uso del sistema dependiendo del usuario.

#### **1.4.4. Históricos del sistema prototipo**

- Almacenar en una base de datos los datos de los productos, empleados y clientes del local comercial.
- Consultar todos los eventos pasados cuando se necesite.

Para demostrar que el sistema alcanza las metas propuestas deberá cumplir con lo siguiente:

- Contar con un sistema prototipo que trabaje mediante teledetección con la señal de la tecnología RFID.
- ✓ **El sistema será capaz de monitorear por medio de los tags:**
  - El precio de cada producto para su ágil facturación y pago.
  - La hora de entrada y salida de los empleados del local comercial.
  - Registrar en la base de datos cualquier producto nuevo, empleados y clientes.
  - El robo de productos al salir del local comercial.
- ✓ **El sistema permitirá ver:**
  - Datos en tiempo real de eventos ocurridos.
  - Datos estadísticos de eventos ocurridos.
  - El historial de eventos ocurridos en un determinado tiempo, realizando la consulta en la base de datos.
- ✓ **El sistema no realizará:**
  - Contabilidad.
  - Producción.
  - Inventarios.
  - Roll de Pagos.
  - Cardex.
  - Localización de los productos en locales comerciales.

## 1.5. DESCRIPCIÓN GENERAL DEL SISTEMA

El sistema prototipo a desarrollar, trata de analizar, diseñar y construir diferentes módulos que trabajen por medio de teledetección a través de la tecnología RFID (identificación por radio frecuencia) de mediano alcance que trabaje con frecuencia UHF de 902-928 MHz, estos módulos permitirán tener

una facturación ágil y automatizada por cada venta, también se tendrá un control completo de stocks de productos, como la oportunidad de gestionar tanto a clientes, a empleados en sus horarios de entrada y salida de sus turnos de trabajos y se tendrá un monitoreo constante de tentativa de robo detectando cual producto es el que se están sustrayendo.

Se debe recalcar que este software es un prototipo lo cual es una “demostración limitada pero valida, como también funcional del diseño del software planificado para experimentar, estudiar en situaciones reales su uso y comportamiento por lo que no lleva a cabo todas las funciones necesarias del software final” .<sup>1</sup>

El prototipo al ser construido no debe tomar demasiado tiempo, usando los programas adecuados y con un presupuesto económico, la idea es comunicar, discutir, definir ideas entre los diseñadores y las partes responsables.

Los prototipos apoyan la evaluación, clarifican requisitos de usuario y definen alternativas para mejorar el producto final.

## **Ventajas<sup>2</sup>**

- Este modelo es útil cuando el cliente conoce los objetivos generales para el software, pero no identifica los requisitos detallados de entrada, procesamiento o salida.
- También ofrece un mejor enfoque cuando el responsable del desarrollo del software está inseguro de la eficacia de un algoritmo, de la adaptabilidad de un sistema operativo o de la forma que debería tomar la interacción humano-máquina.

---

<sup>1</sup> Definición de Prototipo Tomado de: Lacalle Alberto

<sup>2</sup> Los prototipos Tomado de: <http://www.eumed.net/libros/2008a/358/LOS%20PROTOTIPOS.htm>

## 1.5.1. Gráficos de los módulos del sistema prototipo que trabaja con RFID

### 1.5.1.1. Módulo gestión de caja/ventas

En este modulo se ingresará, actualizará y se buscará a todos los productos del local comercial, para identificar a cada uno de ellos se colocarán los tag RFID de identificación, y de esta manera realizar una facturación automática, rápida, ágil y segura por medio de teledetección con lector y la antena RFID, como también tener un control de stock de productos para ofrecer reportes de los productos, según sea la necesidad<sup>3</sup>. (Ver Figura 1).



Figura 1: "Módulo gestión caja/ventas"

Fuente: Los Autores

<sup>3</sup> Los Tag RFID Tomado de: [http://www.adventech-logistica.com/productos/rfid\\_tags.php](http://www.adventech-logistica.com/productos/rfid_tags.php)

### 1.5.1.2. Módulo gestión de empleados

En este módulo se ingresará, actualizará y se buscará a todos los empleados del local comercial. Para identificar a cada uno de ellos colocaremos los tag RFID de identificación en su credencial o carnet de identificación, y de esta manera tener un registro de las horas de entradas y salidas de su lugar de trabajo, por medio de teledetección con lector y la antena RFID, como también tener un listado de empleados que han cometido una falta por medio de reportes, según sea la necesidad<sup>4</sup>. (Ver Figura 2).



Figura 2: “Módulo gestión de empleados”

Fuente: Los Autores

<sup>4</sup> Identificación Tomado de: <http://es.thefreedictionary.com/identificaci%C3%B3n>

### 1.5.1.3. Módulo monitoreo de robos

En este módulo se monitoreará las tentativas de robo, hurto o subtracciones ilegales de productos a la salida del local comercial, por medio de los tag RFID de identificación colocados a cada producto, y de esta manera tener un monitoreo constante de los productos que pudieran ser robados, por medio de teledetección con lector y la antena RFID, como también verificar exactamente que producto es el que se están sustrayendo por más pequeño o grande que este sea, y así dar informes o reportes de tentativas de robos según sea la necesidad<sup>5</sup>. (Ver Figura 3).

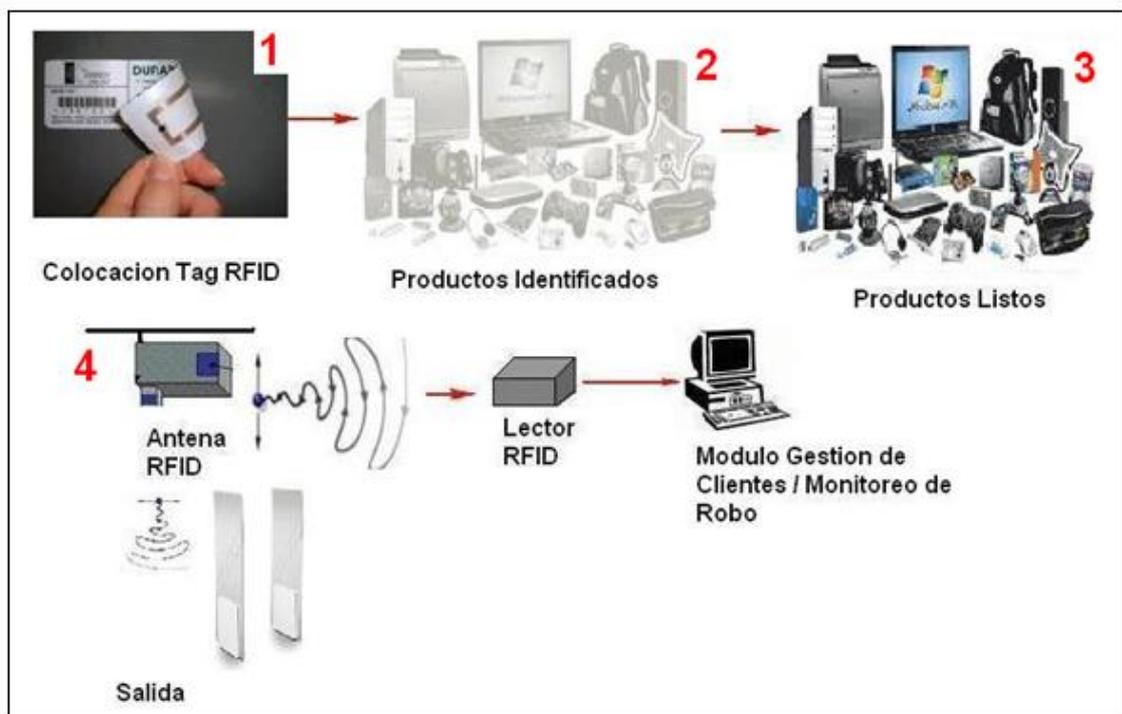


Figura 3: “Módulo monitoreo de robos”

Fuente: Los Autores

<sup>5</sup> Monitoreo de robos Tomado de: <http://comercioweb.jfinternational.com/proteccion-listas-email.html>

# CAPÍTULO II

## 2. SUSTENTO TEÓRICO DEL PROYECTO

En este capítulo, se notificará la bases teóricas que argumentan la investigación la misma en donde se expondrá de forma general una prospectiva del proyecto que se pretende elaborar, a través de los principales conceptos vinculados con ventas, pagos automáticos e instantáneos, gestión de horarios de entrada, salida de empleados y control de robos utilizando tag de identificación con la tecnología RFID (identificación por radio frecuencia) con frecuencias UHF de 860-960 MHz aproximadamente, así como las primordiales plataformas que serán parte fundamental para el éxito del proyecto y su correcta ejecución.

### 2.1. INGENIERÍA DE SOFTWARE<sup>6</sup>

#### 2.1.1. Definición

La ingeniería de Software (SE del inglés Software Engineering) es la rama de la ingeniería que aplica los principios de la ciencia de la computación y las matemáticas para lograr soluciones costo-efectivas (eficaces en costo o económicas) a los problemas de desarrollo de software<sup>7</sup>. (Ver Figura 4).



Figura 4: “Ingeniería de Software”

Fuente: Ciclo de vida del Software [www.kioskea.net](http://www.kioskea.net)

<sup>6</sup> Ingeniería de software Tomado de: libro ingeniería de software de Roger S. Pressman

<sup>7</sup> Ingeniería de software Tomado de: <http://www.angelfire.com/scifi/jzavalar/apuntes/IngSoftware.html>

## **2.1.2. Etapas del proceso de la Ingeniería de Software<sup>8</sup>**

La ingeniería de software requiere llevar a cabo numerosas tareas, dentro de etapas como las siguientes:

### **2.1.2.1. Análisis de requerimientos<sup>9</sup>.**

Extraer los requisitos y requerimientos de un producto de software es la primera etapa para crearlo, se requiere de habilidad y experiencia en la ingeniería de software para reconocer requerimientos incompletos, ambiguos o contradictorios.

### **2.1.2.2. Especificación**

La Especificación de Requisitos describe el comportamiento esperado en el software una vez desarrollado, gran parte del éxito de un proyecto de software radicará en la identificación de las necesidades del negocio, e puede utilizar estas técnicas:

- Casos de Uso,
- Historias de usuario.

### **2.1.2.3. Arquitectura**

La integración de infraestructura, desarrollo de aplicaciones, bases de datos y herramientas gerenciales, requieren de capacidad y liderazgo para poder ser conceptualizados y proyectados a futuro, solucionando los problemas de hoy.

Un diseño arquitectónico describe en general el cómo se construirá una

---

<sup>8</sup> Etapas de ingeniería de software Tomado de: <http://www.monografias.com>

<sup>9</sup> Sommerville, 2004 Tomado de: <http://www.rodolfoquispe.org>

aplicación de software. Para ello, se documenta utilizando diagramas, por ejemplo:

- Diagramas de clases
- Diagramas de base de datos
- Diagramas de Actividades
- Diagramas de Componentes

#### **2.1.2.4. Programación**

Reducir un diseño a código puede ser la parte más obvia del trabajo de ingeniería de software, pero no necesariamente es la que demanda mayor trabajo y ni la más complicada.

#### **2.1.2.5. Pruebas**

Consiste en comprobar que el software realice correctamente las tareas indicadas en la especificación del problema.

#### **2.1.2.6. Documentación**

Todo lo concerniente a la documentación del propio desarrollo del software y de la gestión del proyecto, pasando por modelaciones (UML), diagramas, pruebas, manuales de usuario, manuales técnicos, etc.

#### **2.1.2.7. Mantenimiento**

Mantener y mejorar el software para enfrentar errores descubiertos y nuevos requisitos.

## **2.2. LENGUAJE DE MODELADO UNIFICADO UML**

Lenguaje Unificado de Modelado UML. El Lenguaje Unificado de Modelado o UML es una técnica para la especificación de sistemas en todas sus fases, Se ha desarrollado por un sin número de programadores en materia de análisis y diseño de sistemas y ha sido usada con éxito en sistemas hechos para toda clase de industrias alrededor del mundo: salud, bancos, comunicaciones, aeronáutica, finanzas, etc.<sup>10</sup>.

UML no es un lenguaje de programación, existen herramientas que pueden ofrecer generadores de código de UML para una gran variedad de lenguaje de programación, así como construir modelos por ingeniería inversa a partir de programas existentes, este es pues un lenguaje de propósito general para el modelado orientado a objetos, UML es también un lenguaje de modelamiento visual que permite una abstracción del sistema y sus componentes.

### **2.2.1. Fases del ciclo de desarrollo que soporta UML**

Cada diagrama puede ser usado con énfasis distinto en las fase de desarrollo: análisis, diseño e implementación, un diagrama cualquiera en una fase tendrá un estudio lógico, cabe aclarar que aunque UML es orientado a objetos preferentemente, esto es útil en cualquier modelo tecnológico ya que es independiente de lenguajes de programación o tecnología determinada.

---

<sup>10</sup> Sistemas webs modernos Tomado de: Colegio de ingenieros de España “Juan Salvador Castejón”

## 2.3. TECNOLOGÍA RFID

### 2.3.1. Definición:

El RFID en inglés (Radio Frequency Identification), en español (Identificación por Radio-Frecuencia) es una tecnología que sirve para el almacenamiento y recuperación de información (normalmente poca) de manera remota.

El sistema esencial RFID está formado por dos componentes:

- El tag o etiqueta.
- El lector.

Pero para la utilización verídica de la tecnología RFID se necesitan una gama de componentes adicionales como:

- Antenas
- Soportes
- Escritores
- Programadores
- Hosts con software de control<sup>11</sup>.

Una definición más seria sería: RFID es una tecnología y dispositivos que se valen de las ondas electromagnéticas para intercambiar datos de identificación, normalmente esto implica la utilización de un pequeño tag o etiqueta que identifica un objeto específico.

La serie sigue los siguientes pasos: se obtiene una señal de radio, se interpreta y se devuelve un número u otro tipo de información identificativa, complementariamente este proceso puede ser tan complejo como una comunicación bidireccional y ser capaz de llegar a ser encriptado e interpretado a través de una base de datos y transportado a través de varios sistemas de

---

<sup>11</sup> RFID investigación UFAC Tomado de: Diego Contreras Jiménez y Antonio Lacasa Corral

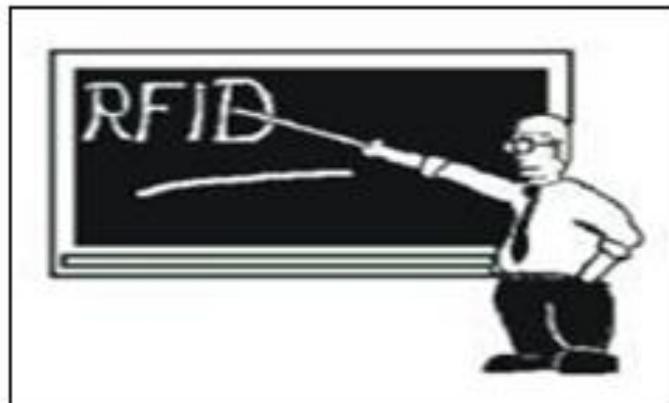
comunicaciones.<sup>12</sup> (Ver figura 5).

El rango de lectura de un sistema RFID está definido como la distancia de lectura entre la etiqueta y el lector, por lo que un sistema RFID puede estar dividido entre los siguientes tres tipos:

- Enganche cercano.- El rango de lectura es menor que 1cm. Operan a frecuencias de LF y HF.
- Enganche remoto.- El rango de lectura es de 1cm a 100cm. Operan a frecuencias de LF y HF.
- Rango largo.- El rango de lectura es mayor a 100cm. Operan a frecuencias de UHF y microondas.

Los diferentes tipos de frecuencia usados en RFID son:

- Baja frecuencia (LF). [30,300] MHz.
- Alta frecuencia (HF). [3,30] MHz.
- Ultra-alta frecuencia (UHF) [300,1000] MHz.
- Microondas (Microwave) [1,5.8] GHz.



**Figura 5: “Identificación por Radio Frecuencia”**

**Fuente:** RFID Básico [www.forodeseguridad.com](http://www.forodeseguridad.com)

---

<sup>12</sup> RFID Sourcebook Tomado de: Sandip Lahiri, Editorial IBM Press, Agosto 31, 2009 “versión traducida”

### 2.3.2. Arquitectura

El modo de funcionamiento de los sistemas RFID es simple. La etiqueta RFID, que contiene los datos de identificación del objeto al que se encuentra adherido, genera una señal de radiofrecuencia con dichos datos.

Esta señal puede ser captada por un lector RFID, el cual se encarga de leer la información y pasarla en formato digital a una aplicación específica que utiliza RFID. (Ver Figura 6).

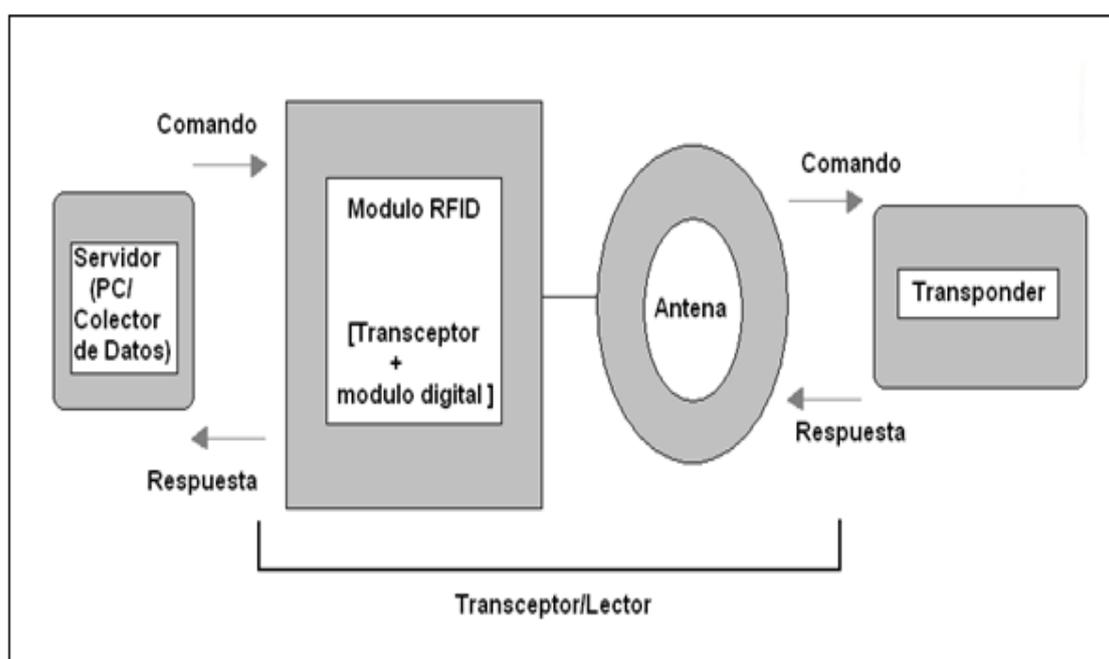


Figura 6: "Arquitectura RFID"

Fuente: Arquitectura básica RFID [www.todoarquitectura.com](http://www.todoarquitectura.com)

### 2.3.3. Sistema RFID

#### 2.3.3.1. Funcionamiento del sistema RFID

Su funcionamiento es fácil de entender por qué trabaja con un dispositivo de radio llamado tag y este está añadido al objeto que se quiere identificar. Este tag contiene unos datos de identificación que hacen único al objeto, este tag

está expuesto al alcance del lector RFID, el tag le transmite la información a través de la antena del lector.

El lector por lo tanto, obtiene la información y la transfiere a través de una vía de comunicación (conexión serie, red local, etc.) a una aplicación software que maneja los datos para identificar el objeto y realizar las acciones pertinentes según sean los requerimientos. Como por ejemplo, la localización de un objeto en una base de datos.

Por lo que el sistema RFID es una herramienta completa que abarca tanto el hardware como el software necesario para el correcto funcionamiento del sistema.<sup>13</sup>.

### **2.3.3.2. Componentes del sistema de funcionamiento RFID**

El sistema RFID está compuesto por los siguientes componentes:

- Tag: Es un componente fundamental del sistema RFID.
- Lector: Es igualmente un componente fundamental del sistema RFID.
- Antena del lector: Es otro componente básico del sistema, a veces incorporado en el mismo lector.
- Controlador: Es un componente necesario. pero, en muchos lectores de última generación lo tienen incorporado.
- Sensor, actuador, alarma: Son componentes opcionales que son necesarios para interactuar con el sistema según su uso.
- Equipo y sistema software: hipotéticamente, un sistema RFID puede trabajar independientemente de este componente. Pero un RFID sin estos carece de utilidad.

---

<sup>13</sup> RFID Essentials Tomado de: Himanshu Bhatt, Bill Glover, Editorial O'Reilly, Enero 2010  
"versión traducida"

- Infraestructura de comunicación: Es una parte importante del sistema que conecta los componentes previamente listados, ya sean alambicados o no, para hacer una comunicación efectiva entre ellos. (Ver Figura 7).

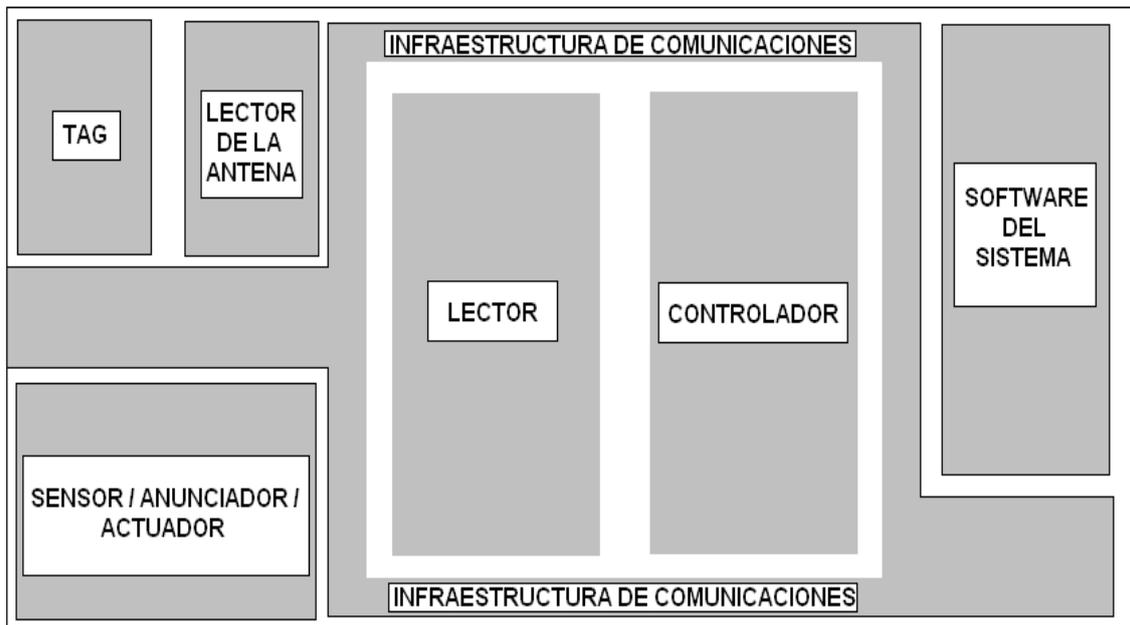


Figura 7: “Componentes del sistema de funcionamiento RFID”

Fuente: Componentes RFID [www.kimaldi.com](http://www.kimaldi.com)

### 2.3.3.3. Tags RFID

Las etiquetas RFID o tags en inglés son unos dispositivos pequeños, que pueden ser adheridos o incorporados a un producto, un animal o una persona, los tag contienen antenas para permitirles recibir y responder a peticiones por radiofrecuencia desde un emisor-receptor RFID.

Los tag o etiquetas electrónicas se pueden clasificar de formas diferentes, considerando su alimentación, su rango de lectura o su posibilidad de lectura/escritura. (Ver Figura 8).

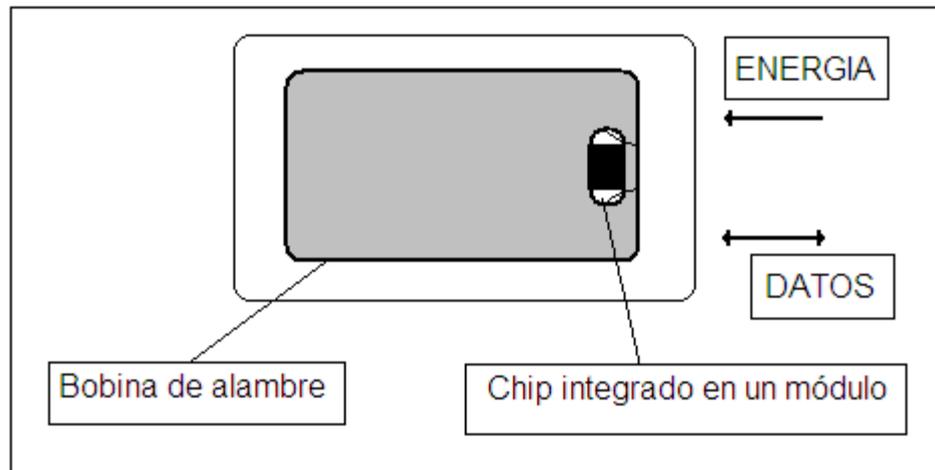


Figura 8: "Tags RFID"

Fuente: Tag RFID [www.informatica-hoy.com.ar](http://www.informatica-hoy.com.ar)

Considerando su forma de alimentación, se pueden clasificar en tag<sup>14</sup>:

- **Pasivos.**- Estos tag no necesitan alimentación eléctrica interna.
- **Activos.**- Estos tag si necesitan alimentación eléctrica interna.
- **Semi-activos / Semi-pasivos.**- Estos tag poseen una fuente de energía interna, una batería, y unos circuitos para realizar tareas específicas.

Considerando su forma de escritura y lectura, se pueden clasificar en tag<sup>15</sup>:

- **Solo lectura (RO).**- Estos tag solo pueden ser escritos una sola vez en toda su existencia y se hace en la cadena de fabricación de estos a modo de ROM.
- **Una Escritura / Varias lecturas (WORM).**- Los tags WORM solo se pueden escribir una vez pero no por el fabricante, sino por el usuario. En realidad pueden reescribirse (menos de 100 veces) por fallos en el bosquejo, pero solo algunos modelos, aunque si superan un cierto número de veces pueden quedar inutilizables.

<sup>14</sup> Tipos de Tag RFID Tomado de: <http://www.rfidpoint.com>

<sup>15</sup> Tag RFID Tomado de: [http:// http://www.rfidtags.com/](http://http://www.rfidtags.com/)

- **Lectura / Escritura (RW).**- Estos tag RW permiten ser reescritos un gran número de veces (de entre 10.000 hasta 100.000), y tienen el gran atributo de poder ser escritos por los lectores o incluso por si mismos (en el caso de los tags activos) a estos tag se los llaman también reprogramables.

#### 2.3.3.4. Lector RFID

El lector es un dispositivo que puede leer o incluso escribir datos en tags RFID compatibles, la acción de escribir en un tag se le llama “crear” un tag, fase en la cual se le asocia un identificador que lo vinculará a un objeto, al vincullarlo lo que se está haciendo es apoderar el tag. Lo contrario es desapoderar es desasociarlo del objeto y destruyendo el tag opcionalmente.

El período que el lector produce energía RF<sup>16</sup> para leer el tag es el tiempo de espera del lector y el lector es la parte más importante del hardware de un sistema RFID y está constituido por:

- **Transmisor.**- Se emplea para transmitir la corriente alterna y el período de reloj por las antenas hacia los tags a leer esta en el módulo emisor/receptor donde envía la señal del lector a la distribución y recibe la respuesta por la antena/s.
- **Receptor.**- Complementa al módulo emisor/receptor y es el que manda a recibir la señal analógica del tag por la antena y la transfiere al microprocesador del lector el cual se transformara a señal digital.
- **Microprocesador.**- se encarga de implementar los protocolos de comunicación con los tag, es decir decodifica y comprueba errores en la señal analógica recibida para después filtrarla y tratar la información del tag.

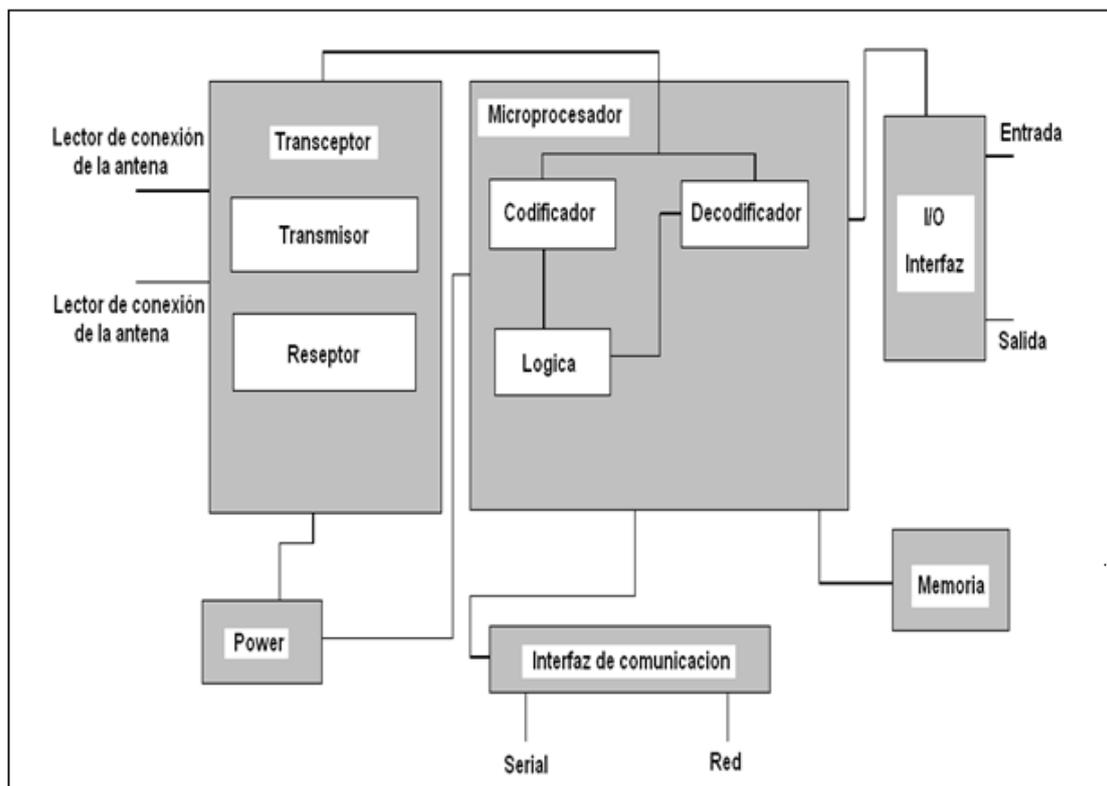
---

<sup>16</sup> RF Siglas de: Radio Frecuencia

- **Memoria.-** Se emplea para almacenar diversos datos por decir uno los parámetros de configuración del lector, y si existe un fallo de comunicación entre el lector y el controlador no se perdería toda la información leída de los tags.
- **Entradas y salidas para los sensores externos / actuadores / alarmas.-** El lector por lo general no siempre manda señales de lectura ya que se hace un gasto innecesario de energía, las entradas y salidas sirven para localizar la presencia de objetos y activar el lector o para activar algunas alarmas o actuadores según su función.
- **Controlador (suele ser externo).-** Es un dispositivo externo que trabaja con los datos identificativos del tag para realizar diversas operaciones.
- **Interfaz de comunicación.-** Es el mecanismo que provee las instrucciones para la comunicación entre el lector y los dispositivos externos para enviarles la información obtenida del tag y recibir comandos que se traducen para escribir en el tag<sup>17</sup>.
- **Energía.-** Es el que suministra la corriente eléctrica a todos los componentes del lector. (Ver Figura 9).

---

<sup>17</sup> Guía sobre tecnologías RFID Tomado de <http://www.mastermagazine.info/articulo/9105.php>



**Figura 9: “Lector RFID”**

**Fuente:** Lectores RFID [www.slideshare.net](http://www.slideshare.net)

### 2.3.3.5. Antena del Lector RFID

El lector se comunica con los tags por medio de la antena del lector, que suele estar separada físicamente del lector y conectada con esta a través de un cable.

La antena es la que crea el campo electromagnético que induce la corriente a la antena del tag, de esta manera para conseguir que se lea un tag deberá estar próxima a la antena del lector, en la actualidad hay muchos lectores que pueden llevar integrada la antena.

El rango de la antena es la zona en la que esta es más efectiva, es decir, donde mejor asigna su señal y es más sencillo que se lea el tag<sup>18</sup>.

<sup>18</sup> Antenas RFID Tomado de: Editorial IBM Press, Agosto 31, 2010 “Versión Traducida”

En realidad, el rango de la antena no es uniforme y presenta algunas deformaciones con zonas muertas en las que no se envía la señal y no se podría leer el tag, por esta razón es muy importante encontrar el rango de acción de la antena para determinar cuáles son las zonas se quedarán ciegas. (Ver Figura 10).

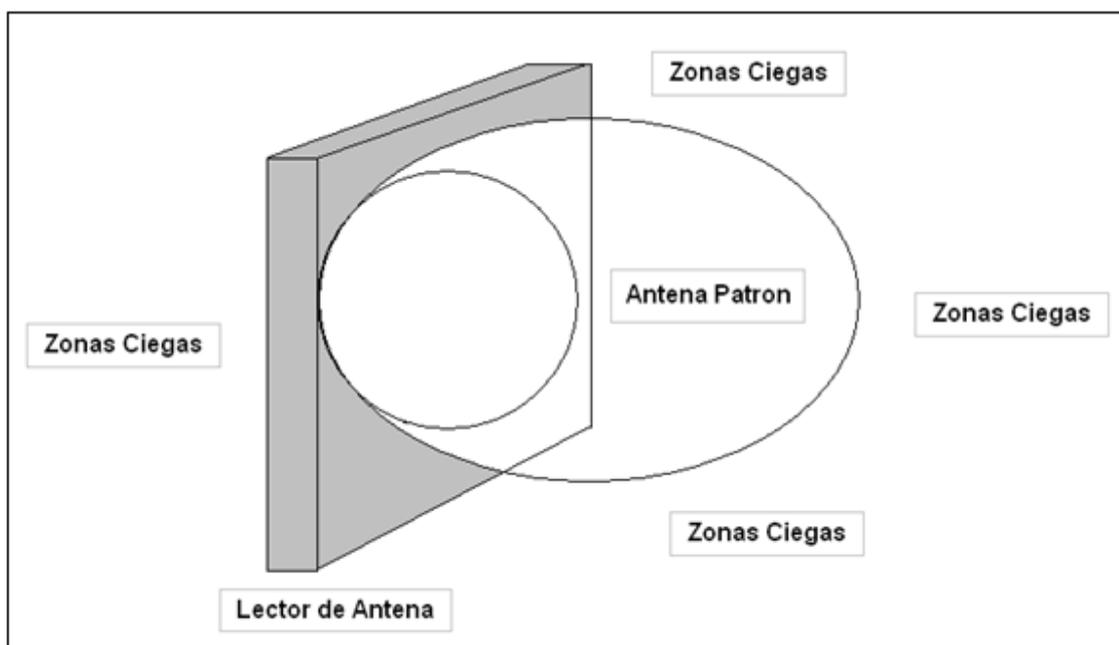
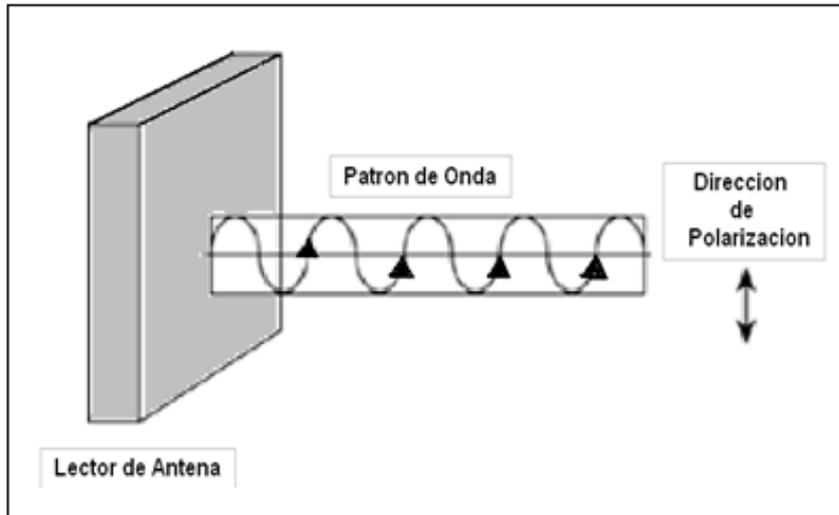


Figura 10: “Rango Antena Lector RFID”

Fuente: Antena RFID [www.rfidpoint.com/](http://www.rfidpoint.com/)

La polarización de la antena es la dirección de oscilación en la que son emitidas las ondas, en si esta figura es importante en robustez de lectura como también en el rango o alcance de lectura y se divide en dos tipos que son:

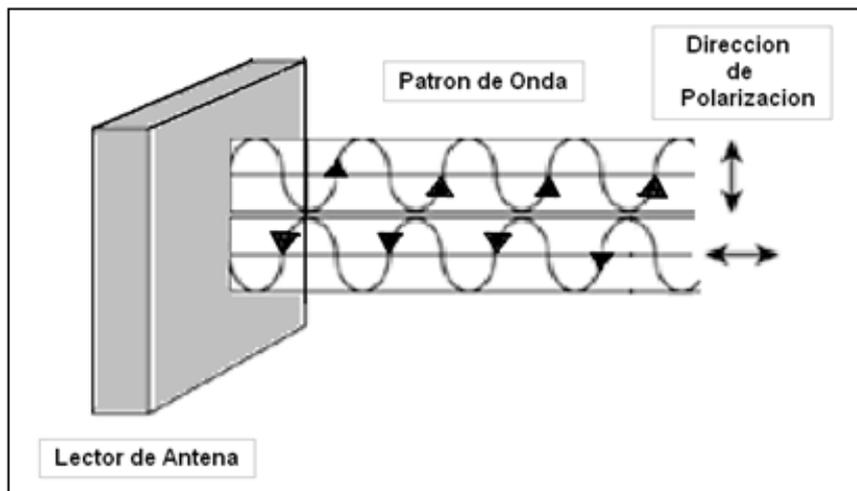
- **Polarizada Linealmente.-** Tienen mucho más alcance que las antenas polarizadas circularmente, sin embargo es sensible a la orientación del tag con relación a la dirección de polarización por lo que únicamente trabaja cuando ya esta predefinido y predicho la orientación del tag. (Ver Figura 11).



**Figura 11: “Antena Polarizada Linealmente”**

**Fuente:** Polaridad de antena RFID [www.rfidpoint.com](http://www.rfidpoint.com)

- Polarizada Circularmente.-** Utilizan un patrón de irradiación circular y consta de dos ondas de igual amplitud y magnitud, con una diferencia de fase de 90, por lo que al poseer esta polarización la antena no se ve afectada por la orientación del tag y permite cubrir mucha más área.<sup>19</sup> (Ver Figura 12).



**Figura 12: “Antena Polarizada Circularmente”**

**Fuente:** Polaridad de antena RFID [www.rfidpoint.com](http://www.rfidpoint.com)

<sup>19</sup> Antenas RFID Polarizadas Circularmente Tomado de: ODIN Technologies, Dulles V. “Versión Traducida”

La potencia que difunde una antena se mide en ERP (effective radiated power) por lo que la potencia está limitada a los estándares que adopte cada país y para aumentar esta potencia es necesario pedir permisos legales a las autoridades gubernamentales.

#### **2.3.3.6. Controlador RFID**

Es la parte intermedia la cual sirve para la comunicación con una entidad externa y controla el comportamiento del lector junto con los indicadores y los actuadores vinculados con este lector, este es el único dispositivo de un sistema RFID (o un lector, dependiendo del punto de vista) por medio de él las comunicaciones son posibles.

#### **2.3.3.7. Conceptos Fundamentales RFID**

- Las ondas electromagnéticas están creadas por electrones en movimiento y consiste en la oscilación del campo eléctrico y magnético, estas ondas pueden pasar a través de muchos tipos de materiales.
- Al punto más alto de una onda se le llama cresta y al más bajo valle, el alejamiento por medio de dos crestas sucesivas es la longitud de onda.
- Una oscilación terminada de la longitud de onda es denominada ciclo.
- El tiempo que gasta en completarse un ciclo es la de oscilación, su inversa es la frecuencia la cual se mide en hertzios.
- La radiofrecuencia son ondas electromagnéticas con frecuencias comprendidas entre 30 Hz y 300 GHz. En el caso del RFID se utilizan frecuentemente las comprendidas entre 30 KHz y 5.8 GHz.<sup>20</sup>

---

<sup>20</sup> Ondas Electromagnéticas Tomado de: [www-istp.gsfc.nasa.gov](http://www-istp.gsfc.nasa.gov)

- Una onda continua es una onda de radio cuya frecuencia y amplitud son constantes.
- La modulación es el proceso que modifica las generalidades de la onda de radio para codificar alguna señal.
- La amplitud de la onda es la altura de la cresta o la profundidad del valle desde el punto de reposo. (Ver Figura 13).

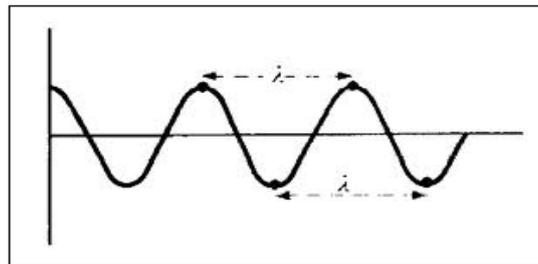


Figura 13: “Ondas Electromagneticas”

Fuente: Longitud de onda [www.buenastareas.com](http://www.buenastareas.com)

#### 2.3.4. El Espectro de Frecuencias RFID

Todas la frecuencias mostradas en el siguiente espectro son RFID, en si se notan que todas ellas tienen un comportamiento físico muy diferente o diferenciado. (Ver Figura 14).

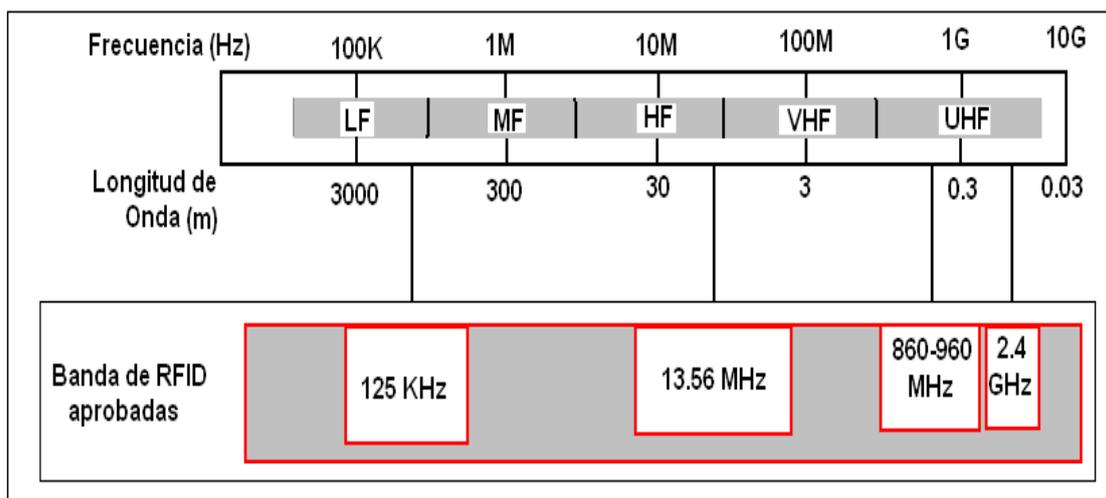


Figura 14: “Espectro RFID”

Fuente: Banda RFID [www.achto.com](http://www.achto.com).

Entonces no es lo mismo trabajar con frecuencias RFID HF a 13.56 MHz que con frecuencias RFID UHF a 868 MHz, las diferencias en tecnologías son grandes, también la distancia de lectura varían entre 1 a 15 metros y su costo es casi el 50% uno del otro.<sup>21</sup>

Las frecuencias RFID son inofensivas para la naturaleza no causa ninguna modificación sobre los productos donde se aplica debido a su baja potencia de emisión con que trabaja. (Observar Tabla 1).

<b>Fuente Electromagnética</b>	<b>Potencia de Emisión</b>
<b>Tag Pasivo RFID</b>	10 a 200 $\mu$ W
<b>Teléfono Móvil</b>	Inferior a 2W
<b>Emisor RFID</b>	2 W
<b>Estación Base GSM</b>	10 a 15 W
<b>Radio Fm</b>	300.000 W
<b>Televisión UHF</b>	5.000.000 W

**Tabla 1: "Potencia de emisión RFID"**

**Fuente:** Frecuencias RFID [www.rfidpoint.com/](http://www.rfidpoint.com/)

El gran avance tecnológico de RFID actual es que puede leer esas potencias dentro de un ambiente de multifrecuencia con más potencia.

---

<sup>21</sup> Tecnología RFID Tomado de: [http://www.dipolerfid.es/tecnologia\\_RFID](http://www.dipolerfid.es/tecnologia_RFID)

### 2.3.5. Estandarización RFID

La estandarización de RFID es importante para conocer y estudiar sus diferentes versiones, ISO y EPC global son dos de las organizaciones más importantes que definen las normas y los estándares de radio frecuencia y su identificación.

- **EPCglobal.-** Es una organización que lidera el desarrollo de estándares de la industria, impulsados por las normas del Código Electrónico de Producto o EPC, esta organización fue creada para establecer y dar soporte a la red de EPC (Electronic Product Code) como una especificación mundial que llevará a un estándar internacional (ISO) a fin de lograr que la identificación de cualquier artículo dentro de la cadena de suministro sea inmediata, automática y exacta, por lo que se ha transformado en la principal organización para el desarrollo de especificaciones en RFID.
- **ISO.-** (Organización internacional de normalización) es una organización no gubernamental, que define y estudia normas, estas normas permiten llegar a un consenso sobre las soluciones que satisfagan tanto los requisitos de las empresas y las necesidades más amplias de la sociedad.

Los estándares de RFID abarcan cuatro áreas principales dependiendo de su funcionamiento:

- **Protocolo en el interfaz aéreo.-** Especifica el modo en el que etiquetas RFID y lectores se comunican mediante radiofrecuencia.
- **Contenido de los datos.-** Especifica el formato y semántica de los datos que se comunican entre etiquetas y lectores.
- **Certificación.-** Pruebas que los productos deben cumplir para garantizar que cumplen los estándares y pueden inter operar con otros dispositivos de distintos fabricantes.

- **Aplicaciones.-** Usos de los sistemas RFID, para diversos proyectos o actividades de gobierno, milicia y personas en general.

### 2.3.5.1. Norma ISO Para RFID

Hay una variedad de normas ISO para la regulación de radio frecuencias, pero para normalizar las diferentes frecuencias utilizadas por RFID (identificación por radio frecuencia), ISO a incluido todas estas normas en un sola especificación que es ISO-18000 con diferentes versiones. (Observar Tabla 2)

Versión	Frecuencias
ISO 18000-2	<135 KHz
ISO 18000-3	HF a 13.56 MHz
ISO 18000-4	UHF a 2.45 GHz
ISO 18000-6 (A, B o C)	UHF entre 860-960 MHz
ISO 18000-6C	RFID UHF EPC Gen 2
ISO 18000-7	UHF 433 (activa)

Tabla 2: “Versiones ISO 18000”

Fuente: Estándares ISO 18000 [www.hightechaid.com](http://www.hightechaid.com)

ISO 18000 es un conjunto de normas internacionales establecidas por la Organización Internacional de Normalización (ISO) para apoyar el desarrollo y la integración de la identificación por radiofrecuencia (RFID).

La razón para desarrollar normativas ISO RFID diferentes según el tipo de frecuencias, se da por lo que entre ellas existen diferencias básicas en las leyes físicas que las conducen.<sup>22</sup>

- Parte 1 – 18000-1: Parámetros genéricos para Interface por aire para frecuencias de aceptación global.

<sup>22</sup> Servicio de educación RFID Tomado de: RFID Tag Attachments

- Parte 2 -18000-2: Parámetros para comunicaciones de interface por aire por debajo de una banda de 135 MHz.
- Parte 3 -18000-3: Parámetros para comunicaciones de interface por aire en una banda de 13.56 MHz.
- Parte 4 -18000-4: Parámetros para comunicaciones de interface por aire en una banda de 2.45 GHz.
- Parte 5 -18000-5: Parámetros para comunicaciones de interface por aire en una banda de 5.8 GHz (retirados).
- Parte 6 -18000-6: Parámetros para comunicaciones de interface por aire en una banda de 860 a 960 MHz (usualmente pasivos).
- Parte 7 -18000-7: Parámetros para comunicaciones de interface por aire en una banda de 433 MHz (usualmente activos).<sup>23</sup>

Las partes 6A y B de ISO 18000 especifican las pautas de interface por aire para RFID pasiva en UHF (900 MHz), también se describe esta norma en estas tres formas cuando hablamos según el ámbito de aplicación:

- Las interacciones físicas entre el interrogador y la etiqueta.
- Los protocolos y los comandos.
- Los sistemas de arbitraje de colinos.

El estándar ISO 18000-6B UHF utiliza el enlace directo una modulación bifásica y la conocida codificación Manchester a los 10 o 40 Kbit/s, la ISO 18000-6B UHF se diferencia de las otras versiones como la versión ISO 18000-6A UHF por su modulación que utiliza impulsos en el intervalo de 33 Kbit/s y su mecanismo de adaptación.<sup>24</sup>

<sup>23</sup> ISO 18000-6B UHF RFID Tomado de: <http://www.ehow.com/>

<sup>24</sup> RFID SourceBook Tomado de: Publicación IBM Fress Book LAHIRINI Sandip 2010

### 2.3.5.2. Norma EPC G2 (930 MHz)

Para algunas aplicaciones EPC Global Inc. define algunos protocolos basados en el EPC<sup>25</sup> que pueden funcionar en elementos RFID de HF o UHF, para los elementos de clase 1 se realizó una revisión mejorando de diversos parámetros tales como:

- Funcionamiento en entornos densos de lectores y tags.
- Orden de destrucción con contraseña.
- Mayor velocidad de transmisión.
- Mejor interoperabilidad basada en RFID a frecuencias UHF

Además esta versión mejorada es la conocida como Generación 2 (C1G2, Class1 Gen2) la misma que está incluida como tipo C en la norma ISO18000-6.

La clase 2 tags pasivos de EPG (de lectura-escritura) tienen la memoria específica para el código EPC, los chips usados pueden tener memoria no volátil adicional para datos del usuario.

Entonces la versión ISO 18000-6C UHF que es identificada por EPC Global como EPC Generación 2 (C1G2, Class1 Gen2) UHF tiene 2 especificaciones de enlace directo que son:

- Intervalo de pulsos codificado en 26.7 a 128 Kbit/s.
- Un reverso que utiliza la banda base FM0 o enlace de retorno Miller.

Este proyecto de tesis se basará en el estándar EPC G2 o la ISO 18000-6 donde se trabajará específicamente con el estándar ISO 18000-6C, los cuales ayudan en el mejor funcionamiento, desempeño del proyecto de tesis por su largo alcance, su escritura y su multi-lectura del tag RFID.

---

<sup>25</sup> EPC Siglas en ingles de : El código electrónico de producto“

### 2.3.6. Limitaciones de la tecnología RFID

La tecnología RFID como toda tecnología en evolución tiene varias limitaciones, pero cada limitación que existe en la actualidad tiene una buena probabilidad de ser solucionada según el paso del tiempo y el avance tecnológico<sup>26</sup>, entre estas limitaciones se encontró:

- **Bajo rendimiento con materiales RF opacos y objetos absorbentes.-** Esta es una conducta auxiliar de los materiales, por lo que la tecnología vacante no trabaja bien con estos materiales y en muchos de los casos falla completamente.
- **El impacto de factores ambientales.-** Las limitaciones del medio pueden impactar considerablemente en las soluciones RFID.
- **Lectura limitada de etiquetas.-** Un margen práctico se aplica al hecho de la cantidad de etiquetas que pueden ser leídas en un ciclo determinado.
- **Impacto por interferencia de Hardware.-** Una solución RFID puede tener un impacto perjudicial con el hardware instalado (por ejemplo, la colocación de la antena con su respectiva orientación).
- **Tecnología en desarrollo.-** Pese a que la tecnología RFID está rápidamente evolucionando, estos cambios pueden representar un inconveniente para quien no sepa del tema.

## 2.4. TELEDETECCIÓN<sup>27</sup>

La teledetección remota es la adquisición de información a pequeña o gran escala de un objeto, ya sea usando instrumentos de identificación o instrumentos de escaneo en tiempo real inalámbricos o que no están en

---

<sup>26</sup> RFID Essentials Tomado de: publicación GLOVER Bill pág. 57 2009

<sup>27</sup> Chuiwieco E. Tomado de: Fundamentos de Teledetección Espacial. Rialp. Madrid 2010

contacto directo con el objeto. (Ver Figura 15).



**Figura 15: “Teledetección con RFID”**

**Fuente:** Los Autores

## **2.5. PROTOTIPOS**

Los prototipos son una representación limitada de un producto, permite a las partes probarlo en situaciones reales o explorar su uso, creando así un proceso de diseño de iteración que genera calidad.

Un prototipo puede ser cualquier cosa, desde un trozo de papel con sencillos dibujos a un complejo software<sup>28</sup>.

### **2.5.1. Utilidad de los Prototipos**

Los prototipos son útiles porque a través de ellos podemos comunicar, discutir y definir ideas entre los diseñadores y las partes responsables es decir los prototipos apoyan la evaluación de productos, clarifican requisitos de usuario y definen alternativas buenas y fáciles.

A continuación se detallará el lenguaje de programación que se utilizará para la construcción del sistema prototipo de gestión de planta utilizando la tecnología

<sup>28</sup> Prototipos Tomado de : <http://albertolacalle.com>

RFID, este lenguaje de programación es versátil y tiene dos versiones la versión licenciada y la versión libre. Al escoger el equipo RFID, dependiendo de sus especificaciones, se analizará si se trabajará con el lenguaje de programación C# licenciado o libre.

## 2.6. LENGUAJE DE PROGRAMACIÓN C#<sup>29</sup>

El lenguaje de programación C# fue creado por el danés Anders Hejlsberg que diseño también los lenguajes Turbo Pascal y Delphi, en si C# (pronunciado en inglés “C sharp” o en español “C sostenido”) es un lenguaje de programación orientado a objetos, con este nuevo lenguaje se quiso mejorar con respecto de los dos lenguajes anteriores de los que deriva el C, y el C++.

Con C# se pretendió incorporarse las ventajas o mejoras que tiene el lenguaje JAVA, así que se consiguió que tuviese las ventajas del C, del C++, pero además la productividad que posee el lenguaje JAVA y se le denominó C#, a pesar que el lenguaje C# forma parte de la plataforma .NET, que es una interfaz de programación de aplicaciones. C# es un lenguaje independiente que originariamente se creó para producir programas sobre esta plataforma .NET, por lo que algunos libros, suelen decir que C# es el lenguaje nativo de .NET. (Ver Figura 16).



Figura 16: “Lenguaje de Programación# “

Fuente: C Sharp [www.microsoft.com](http://www.microsoft.com)

<sup>29</sup> C# Tomado de : <http://elvex.ugr.es/decsai/csharp/>

### **2.6.1. Características del lenguaje C#**

Algunas de las características del lenguaje de programación C# son:

- Su código se puede tratar íntegramente como un objeto.
- Su sintaxis es muy similar a la de JAVA.
- Es un lenguaje orientado a objetos y a componentes.
  - Encapsulación: además de los modificadores de acceso convencionales: public, private y protected, C# añade el modificador internal, que limita el acceso al proyecto actual.
  - C# sólo admite herencia simple.
  - Todos los métodos son, por defecto, sellados, y los métodos redefinibles han de marcarse, obligatoriamente, con el modificador virtual.
- Armoniza la productividad del Visual Basic con el poder y la flexibilidad del C++. Ahorramos tiempo en la programación ya que tiene una librería de clases muy completa y bien diseñada.

#### **2.6.1.1. Ventajas y desventajas del lenguaje de programación C#**

A continuación se presentan las ventajas y las desventajas más importantes que tiene el lenguaje de programación C#, algunas de estas no son propias del lenguaje, si no de la plataforma .NET ya que esta plataforma tiene una implicación directa en el lenguaje. (Observar Tabla 3).

<b>Lenguaje de Programación C#</b>	
<b>Ventajas</b>	<b>Desventajas</b>
<b>Declaraciones en el espacio de nombres.</b>	C # es menos flexible que los otros lenguajes.
<b>Completo soporte para programación orientada a objetos (incluyendo crear operadores).</b>	Se tiene que tener algunos requerimientos mínimos del sistema para poder trabajar adecuadamente.
<b>Mejoras en la calidad del código que puedes escribir. C# atrapa muchos errores que otros lenguajes (VB.NET, JAVA) pasan por alto.</b>	Carencia de soporte incorporado para números complejos.
<b>Existen compiladores de C# gratis que puedes utilizar y muchos de estos existen en otras plataformas de computación como (UNIX).</b>	Quien no está familiarizado con ningún lenguaje de programación, le costará más trabajo iniciarse con C#
<b>Todas las nuevas funciones son adaptadas primero en C# y algunas ni siquiera llegan a ser parte de VB.NET.</b>	
<b>Ideal para desarrollo rápido de aplicaciones, por su costo de mantenimiento es definitivamente bajo.</b>	

Tabla 3: “ventajas y desventajas de C#”

Fuente: Los Autores

## 2.7. EL SISTEMA OPERATIVO MICROSOFT WINDOWS<sup>30</sup>

Microsoft Windows es el nombre de una serie de sistemas operativos desarrollados por Microsoft desde 1981, año en que el proyecto se denominaba «Interface Manager».

Anunciado en 1983, Microsoft comercializó por primera vez el entorno operativo denominado Windows en noviembre de 1985 como complemento para MS-DOS, en respuesta al creciente interés del mercado en una interfaz gráfica de usuario (GUI).

Microsoft Windows llegó a dominar el mercado de computadores personales del mundo, superando a Mac OS, el cual había sido introducido previamente a Windows.

En octubre de 2009, Windows tenía aproximadamente el 91% de la cuota de mercado de sistemas operativos en equipos cliente que acceden a Internet, sus versiones más recientes de Windows son Windows 7 para equipos de escritorio, Windows Server 2008 R2 para servidores y Windows Phone 7 para dispositivos móviles. (Ver Figura 17).



Figura 17: “Logo de Windows “

Fuente: Marco del Windows [www.microsoft.com](http://www.microsoft.com)

<sup>30</sup> Windows Especificaciones Tomado de : <http://www.alegsa.com.ar/Dic/windows.php>

### **2.7.1. Historia Cronológica de sistemas operativo Microsoft Windows <sup>31</sup>**

La primera versión de Windows fue la 1.0, lanzada en noviembre de 1985, carecía de funcionalidades y consiguió un poco de popularidad. No era un sistema operativo en sí mismo, sino que era un programa ejecutándose en el sistema MS-DOS pero con grandes logros tecnológicos en esos tiempos.

En noviembre de 1987 fue lanzada la versión 2.0 de Windows, y fue un poco más popular que su predecesora. La versión 2.03, lanzada en enero de 1988, cambió su interfaz de ventanas, haciéndose muy parecido al sistema operativo de las Apple (le trajo a Microsoft problemas legales), pero se solucionaron.

Microsoft Windows 3.0 fue lanzada en 1990, fue la primera versión en alcanzar éxito comercial; vendió dos millones de copias en seis meses. Tuvo mejoras en la interfaz de usuario y las capacidades de multitarea. Luego salió el 1 de marzo de 1992 la versión 3.1 con pequeños cambios no muy generales.

En julio de 1993, Microsoft lanzó Windows NT basado en un nuevo kernel. NT es considerado la línea profesional de los sistemas operativos Windows. La línea de hogar y la línea profesional fueron fusionadas años más tarde con la llegada de Windows XP, el más grande sistema operativo que fue lanzado.

Para agosto de 1995, Microsoft lanza Windows 95, que es considerado realmente un sistema operativo. De todas maneras seguía dependiendo del MS-DOS, por lo que muchos expertos no lo consideran todavía un sistema operativo. Windows 95 cambió completamente su interfaz y se hizo más fácil de usar, para todos los usuarios en general como adultos adolescentes, etc.

Luego, en junio de 1998, se lanzó Microsoft Windows 98, también con gran popularidad, pero con varios problemas de seguridad que necesitaron una Second Edition en 1999 para resolverlos, y no fue tan innovador para la gente.

En tanto en la línea de hogar a Windows 98 le siguió Windows ME (Millennium

---

<sup>31</sup> Microsoft Windows Tomado de : <http://www.microsoft.com/spain/versionwindows/default.msp>

Edition), lanzado en septiembre de 2000. Fue una de las versiones más criticadas del Windows por sus debilidades en la estabilidad y la compatibilidad.

En octubre de 2001, Microsoft lanza Windows XP, la cual está basada en el kernel de Windows NT, pero que también incorpora características de la línea de hogar. Esta versión fue sumamente elogiada en revistas de computación, aunque de todas maneras necesitó dos Service Pack de actualización para lograr una seguridad robusta, mejoras y novedades para los usuarios. En abril de 2003, fue lanzado Windows Server 2003 en reemplazo de la línea de productos para servidores Windows 2000. Esta versión poseía muchas mejoras y una fuerte seguridad. Fue seguido de Windows 2003 R2 en diciembre de 2005, que trajo nuevas actividades de monitoreo reales.

La siguiente versión del sistema operativo fue Windows Vista, lanzada el 30 de noviembre de 2006 para clientes de negocios, casi al mismo tiempo del 3 pack para Windows XP. En cambio la versión para consumidores fue lanzada el 30 de enero de 2007. Windows Vista trajo mejoras en la seguridad, características más avanzadas en sus gráficos con una interfaz opcional llamada Windows Aero, además de múltiples nuevas aplicaciones.

En el año de 2008 lanza la última versión de Windows Server, una versión estable robusta para dar servicio con innovadoras aplicaciones y una interfaz gráfica agradable y fácil de utilizarla.

En la actualidad 2011 se trabaja mucho con la nueva versión de Microsoft Windows que es la versión Windows 7 que es una versión mejorada y estable. (Ver Figura 18).

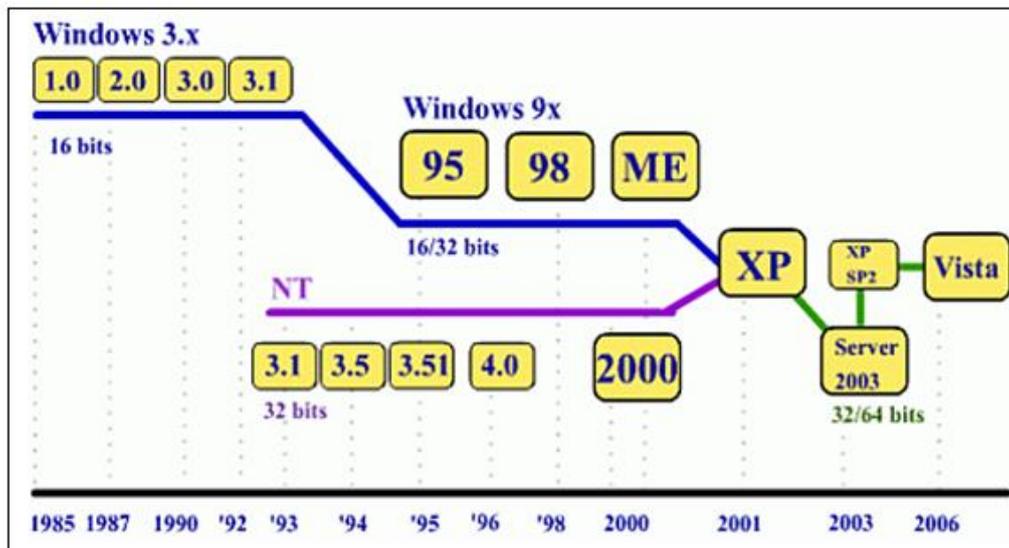


Figura 18: “Cronología de Microsoft Windows “

Fuente: Historia de Windows [www.alegsa.com.ar](http://www.alegsa.com.ar)

### 2.7.1.1. Principales versiones de Windows<sup>32</sup>

- 1985 (junio): Windows 1.01
- 1986 (agosto): Windows 1.03
- 1987 (diciembre): Windows 2.03
- 1988 (junio): Windows 2.1
- 1990 (mayo): Windows 3.0
- 1992 (abril): Windows 3.1
- 1992 (octubre): Windows For Workgroups 3.1
- 1993 (febrero): Microsoft Bob
- 1993 (agosto): Windows NT 3.1
- 1993 (noviembre): Windows For Workgroups 3.11
- 1994 (septiembre): Windows NT 3.5
- 1994 (noviembre): Windows NT 3.51
- 1995 (agosto): Windows 95
- 1996 (septiembre): Windows NT 4.0
- 1998 (junio): Windows 98
- 2000 (febrero): Windows 2000

<sup>32</sup> Microsoft Windows Tomado de : <http://www.alegsa.com.ar/Dic/windows.php>

- 2000 (julio): Windows ME
- 2001 (octubre): Windows XP
- 2002 (enero): Microsoft Windows Server 2003
- 2006 (noviembre): Windows Vista
- 2009 (enero): Microsoft Windows Server 2008
- 2009 (agosto): Windows 7

### 2.7.2. Críticas y Polémicas<sup>33</sup>

Windows, desde sus inicios ha estado envuelto en la polémica. Al principio se decía que Windows era una copia del sistema operativo de Apple; más adelante se hablaba de si existía competencia desleal con algunos programas que se incluían dentro del sistema. Con la aparición del software libre las polémicas se orientan a la política de código cerrado de Microsoft.

Las mayores críticas que recibió Windows hasta la versión Windows XP Service Pack 2 eran la estabilidad del sistema. El sistema operativo presentaba varios fallos de distinta índole y gravedad, los cuales fueron disminuyendo con el correr de las versiones. Desde Microsoft siempre expresaron que estos fallos se debían a aplicaciones externas a Windows, pero algunos fallos se producían apenas instalado el sistema, sin siquiera haber agregado programa alguno.

Otra crítica que se le hace al sistema, específicamente a Windows Vista, es la gran cantidad de recursos del sistema que ocupa, estando «sobrecargado» de objetos, los cuales hacen que los computadores de hoy no soporten adecuadamente el sistema y no ofrezcan al usuario una experiencia fluida de uso.

---

<sup>33</sup> Críticas de Microsoft Windows Tomado de : <http://www.softwarelibre.com.ch/>

# CAPÍTULO III

## 3. ANÁLISIS DEL PROYECTO

En esta parte del proyecto, se tendrá en cuenta todas las necesidades que se tienen que cubrir al momento de desplegar nuestra investigación, por lo que se realizará un análisis detallado sobre las viabilidades del software prototipo a desarrollarse en diversas funciones que son:

- Operacionales.
- Técnicas.
- Económicas.

Estas funciones ayudarán a realizar un buen sistema con una muy buena factibilidad y buen desempeño en su funcionamiento.

### 3.1. ESPECIFICACIÓN DE REQUISITOS

#### 3.1.1. Análisis de Factibilidad

Para ser mucho más específicos, el análisis de factibilidad se basa en evaluar varios aspectos mediante procesos, del probable acierto de nuestro proyecto a construir, traerá consigo un software estable, y los resultados alcanzados se verán reflejados al momento de implementar el sistema.

Este proyecto por ser un prototipo se trabaja ampliamente en la función económica que abarca el análisis de factibilidad, para ser muchos más claros en el estudio de costo beneficio, por lo que, se ocupará equipos económicos y de última tecnología.

Como se puede observar estos factores económicos del proyecto pueden influir sobre un software prototipo exitoso, por lo que se requiere un estudio previo que permita determinar la factibilidad de este proyecto antes que sean tomadas decisiones incorrectas que generen costos adicionales.

## **3.2. FACTIBILIDAD OPERACIONAL DEL PROYECTO**

En esta parte del análisis de factibilidad operacional, se pondrá mucho énfasis en la complejidad del sistema, como también su funcionamiento y el tiempo de vida que puede tener el mismo, para ello, se abalará diversos temas que aclaran la factibilidad operacional del proyecto.

### **3.2.1. Generalidades del Sistema**

#### **3.2.1.1. Dificultad en la utilización del Sistema**

El software que se elaborará, es un mecanismo automatizado por medio de teledetección, el cual estará diseñado fundamentalmente con el objetivo de facilitar:

- Los pagos por ventas, haciendo una facturación inmediata para satisfacer a los clientes dependiendo del tipo, también se controlarán el stock de productos según las ventas que se realicen por día.
- Tener un registro exacto de horarios de entrada y salida de los empleados.
- Tener un control e identificación automático de robos de productos sin pagar.

El sistema prototipo de Gestión de Ventas, Clientes, Personal y Control de Stocks, basado en Teledetección utilizando la tecnología RFID trabajará específicamente dentro de locales comerciales en ambientes cerrados donde cada modulo realizara trabajos independientes, utilizando la misma tecnología.

Hay tecnologías como los ultrasonidos que son muy dependientes sin posibilidad de verificación visual. En nuestro caso sucede lo contrario porque se tendrá visualización de todos los acontecimientos detallados en módulos gracias a los Tags de identificación RFID que se especificarán más a delante, los cuales trabajarán por medio de teledetección.

### **3.2.1.2. Operatividad del Sistema**

En este aspecto el sistema propuesto, es una idea de automatización en pagos, el cual no va a competir con grandes software de pagos por ventas, es decir conocer como es el funcionamiento del software propuesto, de esta forma se construirá módulos adicionales con el propósito de brindar mejoramientos en:

- Ventas.- Detección de productos en forma instantánea para una facturación inmediata.
- Clientes.- Verificación del cliente.
- Personal.- Registro de entrada y salida de los turnos de trabajo de los empleados del local comercial.
- Control de Stocks.- Será según las ventas de los productos.
- Control de Robos.- Cuando un producto es extraído sin pagar.

De esta manera podemos verificar la funcionalidad del software prototipo a realizar con una tecnología nueva de auto detección e identificación que es RFID (Identificación por Radio Frecuencia).

### **3.2.1.3. Formación en la utilización del Sistema**

Para la utilización del software a desarrollar, en caso de ser comercializado, se definirá un CD que incluirá:

- Los drivers del software.
- El manual de usuario.
- El manual de instalación.
- El manual de configuración.
- Tutoriales en texto y video de uso del software.<sup>34</sup>

---

<sup>34</sup> Tutoriales en video Tomado de : <http://www.tutorialesenlared.com/>

Esta información se entregará a los probables usuarios de este software para que de esta manera no tengan ningún inadecuado o mal uso del mismo.

#### **3.2.1.4. Validez del Sistema**

La validez del software es muy importante para la factibilidad del proyecto ya que el desarrollo del software prototipo va a ser elaborado en un lenguaje de programación actual, que va de la mano y trabaja sin ningún problema con equipos tecnológicos nuevos, mas especio con equipos de RFID, este lenguaje de programación es muy común, y conocido por todos los que abarcan esta área de sistemas y telecomunicaciones, por lo que en un futuro se podrá adaptarlo a otras necesidades en diversas industrias.

#### **3.2.2. Evaluación de la factibilidad**

En la evaluación de factibilidad se analizará y evaluará sus características, riesgos, problemas, etc. De las mejores soluciones que sean factibles para realizar el sistema prototipo de Gestión de Ventas, Clientes, Personal y Control de Stocks, basado en Teledetección utilizando la tecnología RFID, de esta forma tener un software optimo, seguro, fiable y factible.

##### **3.2.2.1. Lenguajes de programación**

###### **3.2.2.1.1. Lenguaje de Programación PHP**

###### **3.2.2.1.1.1. Descripción Lenguaje de Programación PHP**

El Lenguaje de programación PHP (Hypertext Preprocessor), es un lenguaje interpretado de alto nivel centrado en páginas HTML, PHP está diseñado para la creación de web dinámicas y de escritorio, se usa principalmente para la interpretación del lado del servidor (server-side scripting), PHP actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz

gráfica usando las bibliotecas Qt o GTK+ (Ver Figura 19) y (Observar Tabla 4).

35



Figura 19: “Logo del lenguaje de programación PHP “

Fuente: PHP [www.alegsa.com.ar](http://www.alegsa.com.ar)

Lenguaje de Programación PHP	
Ventajas	Desventajas
<b>Permite aplicar técnicas de programación orientada a objetos.</b>	PHP no soporta directamente punteros, como C.
<b>Es un lenguaje multiplataforma.</b>	Es necesario instalar un servidor web.
<b>PHP es Open Source.</b>	Promueve creación de código desordenado y con un mantenimiento complejo.
<b>Tiene manejo de excepciones.</b>	Dificulta la organización y programación por capas.

Tabla 4: “Principales ventajas y desventajas de PHP”

Fuente: Los Autores

<sup>35</sup> Lenguaje PHP Tomado de : <http://www.vaslibre.org.ve/publicaciones/phpflisol2006.pdf>

### 3.2.2.1.2. Lenguaje de Programación JAVA

#### 3.2.2.1.2.1. Descripción del Lenguaje de Programación JAVA

Es un lenguaje de programación creado por Sun Microsystem que permite crear pequeñas aplicaciones en diferentes tipos de ordenadores y sistemas operativos.

EL lenguaje de programación para las aplicaciones Java está típicamente compilado en un bytecode, aunque la compilación es en código de máquina nativo.

La plataforma Java consta de las siguientes partes:

- La máquina virtual de Java o JRE, que permite la portabilidad en ejecución.
- El API Java, una biblioteca estándar y única para el lenguaje. (Ver Figura 20) y (Observar Tabla 5).<sup>36</sup>



**Figura 20: Logo del lenguaje de la tecnología JAVA**

**Fuente:** JAVA [www.ayuda-en-pcs.blogspot.com](http://www.ayuda-en-pcs.blogspot.com)

<sup>36</sup> Plataforma JAVA Tomado de : <http://www.programacion.com/java>

<b>Tecnología JAVA</b>	
<b>Ventajas</b>	<b>Desventajas</b>
<b>Capacidad para moverse fácilmente de un sistema informático a otro.</b>	No es multiplataforma, hay problemas al ejecutar en varios SO.
<b>Es un lenguaje de programación orientado a objetos.</b>	Es considerablemente lento respecto de lenguajes como C# o C++.
<b>Java es robusto.</b>	Requiere el plug-in de Java, que no está disponible por defecto en todos los navegadores web.
<b>Permite un control de la memoria y una capacidad de programación de bajo nivel.</b>	La tecnología de la máquina virtual, si se hace referencia a la velocidad, pero no a la portabilidad.

**Tabla 5: “Principales ventajas y desventajas de JAVA”**

Fuente: Los Autores

### **3.2.2.1.3. Lenguaje de Programación C#**

#### **3.2.2.1.3.1. Descripción Del Lenguaje de Programación C#**

Es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET, pero este lenguaje es de código abierto por lo que también se lo considera como software libre.

La sintaxis y estructuración de C# es muy parecida a la de C++ o Java, pero mucho ms rápido en ejecutarse, puesto que la intención de Microsoft es facilitar la migración de códigos escritos en estos lenguajes a C# y facilitar su aprendizaje a los desarrolladores habituados a ellos.

C# es un lenguaje de propósito general que puede ser utilizado para la construcción de aplicaciones web, aplicaciones de escritorio, servicios web,

aplicaciones para celulares y otros componentes. (Ver Figura 21) y (Observar Tabla 6).<sup>37</sup>



Figura 21: “Logo del lenguaje de lenguaje de programación C#”

Fuente: C Sharp [www.elvex.com](http://www.elvex.com)

Lenguaje de Programación C#	
Ventajas	Desventajas
<b>Declaraciones en el espacio de nombres.</b>	C # es menos flexible que los otros lenguajes.
<b>Completo soporte para programación orientada a objetos (incluyendo crear operadores).</b>	Se tiene que tener algunos requerimientos mínimos del sistema para poder trabajar adecuadamente.
<b>Mejoras en la calidad del código que puedes escribir. C# atrapa muchos errores que otros lenguajes (VB.NET, JAVA) pasan por alto.</b>	Carencia de soporte incorporado para números complejos.
<b>Existen compiladores de C# gratis que puedes utilizar y muchos de estos existen en otras plataformas de</b>	Quien no está familiarizado con ningún lenguaje de programación, le costará más

<sup>37</sup> Lenguaje C# Tomado de: <http://urriellu.net/es/articles-software/csharp-advantages.html>

computación como (UNIX).	trabajo iniciarse con C#
Todas las nuevas funciones son adaptadas primero en C# y algunas ni siquiera llegan a ser parte de VB.NET.	
Ideal para desarrollo rápido de aplicaciones, por su costo de mantenimiento es definitivamente bajo.	

Tabla 6: “Principales ventajas y desventajas de C#”

Fuente: Los Autores

#### 3.2.2.1.4. Calificación de la Factibilidad de los Lenguajes de Programación analizados

Después del análisis de factibilidad para los lenguajes de programación, se evaluaron todos los criterios tomados en cuenta para calificar la factibilidad de cada lenguaje. (Observar Tabla 7).

LENGUAJES DE PROGRAMACIÓN	PHP	JAVA	C#
<b>Sistema de evaluación (1/10)</b>			
Costos	8	9	9
Características de diseño, interfaz y escalabilidad.	9	9	10
Seguridad, garantía y conectividad	6	7	8
Capacidad de conexión a base de datos	8	8	9
<b>Riesgos (1/10)</b>			
Estabilidad del lenguaje	6	8	8
Seguridad del código	8	8	8
Adaptación de controladores RFID	6	6	8
Multiplataforma	7	8	7
<b>Problemas (1/10)</b>			
Incompatibilidad con lenguajes de programación antiguos	4	4	8
Disponibilidad de Manejos de métodos	7	8	8

Conexión con equipos RFID	7	6	9
<b>Puntaje Total /(110)</b>	<b>76</b>	<b>81</b>	<b>92</b>

Tabla 7: “Tabla de resultados para la elección del lenguajes de Programación”

Fuente: Los Autores

#### 3.2.2.1.4.1. Resultado de la factibilidad sobre el lenguaje de programación a utilizar

Justificados en la tabla 7, se establece como lenguaje de programación ideal, y ganador, en base al análisis de las calificaciones, al **LENGUAJE DE PROGRAMACIÓN C#** considerando sus características, ventajas, desventajas, y por los resultados que arrojaron los indicadores de la evaluación de la factibilidad.

Lo que lleva a la conclusión que el lenguaje de programación C# es el más apto, en base al cual se construirá el sistema prototipo de Gestión de Ventas, Clientes, Personal y Control de Stocks, basado en Teledetección utilizando la tecnología RFID.

#### 3.2.2.2. Bases de Datos

##### 3.2.2.2.1. Base de Datos SQL Server

##### 3.2.2.2.1.1. Descripción de la Base de Datos SQL Server

Es un sistema de gestión de bases de datos relacionales, basado en el lenguaje T-SQL y ANSI SQL y especialmente en el lenguaje Sybase, IQ proporcionando a muchos usuarios gran cantidad de datos de forma simultánea.<sup>38</sup>

SQL Server constituye una alternativa para la gestión de bases de datos. Entre

<sup>38</sup> BDD SQL Server Tomado de: <http://www.shica19.tripod.com/sql.html>

sus características principales están:

- Facilidad de instalación, distribución y utilización.
- Soporte de Transacciones.
- SQL Server se integra con el correo electrónico, internet y Windows, permitiendo una comunicación local. (Ver Figura 22) y (Observar Tabla 8).



Figura 22: “Logo de Base de Datos SQL Server”

Fuente: Microsoft SQL Server [www.microsoft.com](http://www.microsoft.com)

Base de Datos SQL Server	
Ventajas	Desventajas
<b>Tiene sentencias estándar, las que sirven para cualquier lenguaje de base de datos.</b>	Tiene poca capacidad de almacenamiento de datos limitado.
<b>Consta de escalabilidad, estabilidad y seguridad.</b>	Enorme cantidad de memoria RAM que utiliza para la instalación y utilización del software.
<b>Soporta procedimientos almacenados.</b>	No sirve para prácticas por que prohíben muchas cosas tiene y consta de restricciones.
<b>Permite la administración desde otros</b>	La relación calidad-precio está muy

servidores.	debajo comparado con otras bases de datos
-------------	---

**Tabla 8: “Principales ventajas y desventajas de SQL Server”**

**Fuente:** Los Autores

### 3.2.2.2. Base de Datos Oracle

#### 3.2.2.2.1. Descripción de la Base de Datos Oracle

Oracle es una potente herramienta cliente/servidor para la gestión de Bases de Datos, para su utilización primero sería necesario la instalación de la herramienta servidor (Oracle 8i) y posteriormente podríamos atacar a la base de datos desde otros equipos con herramientas de desarrollo como Oracle Designer y Oracle Developer, que son las herramientas básicas de programación sobre Oracle.<sup>39</sup>

Entre sus características principales están:

- Soporte de transacción
- Escalabilidad.
- Soporte multiplataforma.
- Facilidad de Administración. (Figura 23) y (Observar Tabla 9).



**Figura 23: “Logo de Base de Datos ORACLE”**

**Fuente:** Oracle [www.desarrolloweb.com](http://www.desarrolloweb.com)

<sup>39</sup> BDD Oracle Tomado de: <http://www.zonaoracle.com/>

<b>Base de Datos Oracle</b>	
<b>Ventajas</b>	<b>Desventajas</b>
<b>Posee gran capacidad ilimitada de datos para cualquier información.</b>	Elevado el coste de la formación, y sólo últimamente han comenzado a aparecer buenos libros sobre asuntos técnicos.
<b>Puede ejecutarse en todas las plataformas, desde una Pc. hasta un supercomputador.</b>	El precio. Incluso las licencias de Personal Oracle son excesivamente caras.
<b>Permite el uso de particiones para la mejora de la eficiencia, de replicación.</b>	La versión última no es fiable y hay parches o versiones que arreglan estos desperfectos.
<b>Oracle es la base de datos con mas orientación hacia INTERNET</b>	Oracle mal configurado puede ser lento y su rendimiento bajo.

Tabla 9: “Principales ventajas y desventajas de SQL Server”

Fuente: Los Autores

### 3.2.2.2.3. Base de Datos MySql Server

#### 3.2.2.2.3.1. Descripción de la Base de Datos MySql Server

MySql cuya sigla en inglés es Lenguaje de Consulta Estructurado. Es un sistema de gestión de bases de datos multiusuario, multiplataforma y de código abierto, MySql es muy popular en aplicaciones de escritorio y webs, por lo que es componente de las plataformas LAMP, MAMP, WAMP, entre otras.

Este gestor de bases de datos es, probablemente, el gestor más usado en el mundo del software libre, debido a su gran rapidez y facilidad de uso, su gran aceptación es debida, en parte, a que existen infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de

programación, además de su fácil instalación y configuración (Ver Figura 24) y (Observar Tabla 10).<sup>40</sup>



Figura 24: “Logo de Base de Datos MySql“

Fuente: MySql [www.desarrolloweb.com](http://www.desarrolloweb.com)

Base de Datos MySql Server	
Ventajas	Desventajas
<b>MySql está escrito en C y C++.</b>	No sincroniza los datos con otras bases de datos réplicas.
<b>Completo soporte para operadores y funciones en cláusulas select y where.</b>	Un gran porcentaje de las utilidades de MySql no están documentadas.
<b>Uso de multihilos mediante el uso de kernel.</b>	No tiene integridad referencial.
<b>Soporta gran cantidad de datos. MySql Server tiene bases de datos hasta 50 millones de registros.</b>	
<b>Los clientes se conectan al servidor MySql usando sockets TCP/IP en</b>	

<sup>40</sup> BDD My SQL Tomado de: [http://www.ciao.es/MySQL\\_\\_Opinion\\_1075588](http://www.ciao.es/MySQL__Opinion_1075588)

cualquier plataforma.

Tabla 10: “Principales ventajas y desventajas de MySQL Server”

Fuente: Los Autores

#### 3.2.2.2.4. Calificación de la Factibilidad de las Bases de Datos analizadas

Después del análisis de factibilidad para las bases de datos, se evaluaron todos los criterios tomados en cuenta para calificar la factibilidad de cada base de datos. (Observar Tabla 11).

BASES DE DATOS	SQL SERVER	ORACLE	MYSQL
<b>Sistema de evaluación (1/10)</b>			
Costos	6	5	9
Capacidad de almacenamiento	8	9	8
Escalabilidad, seguridad y conectividad	8	9	9
Capacidad de almacenamiento	8	9	9
<b>Riesgos (1/10)</b>			
Estabilidad de la información	8	8	8
Seguridad de la información	8	8	8
Entornos gráficos	9	8	9
Multiplataforma	7	7	9
<b>Problemas (1/10)</b>			
Incompatibilidad con lenguajes de programación antiguos	8	7	8
Estabilidad en ejecuciones y búsquedas	8	9	9
Seguridad en el almacenamiento de información	8	8	8
<b>Puntaje Total /(110)</b>	<b>86</b>	<b>87</b>	<b>94</b>

Tabla 11: “Tabla de resultados para la elección de la Base de Datos”

Fuente: Los Autores

#### **3.2.2.2.4.1. Resultado de la factibilidad sobre la base de datos a utilizar**

Justificados en la tabla 11, se establece como base de datos ideal y ganadora, en base al análisis de las calificaciones, a la **BASE DE DATOS MYSQL** considerando sus características, ventajas, desventajas, y por los resultados que arrojaron los indicadores de la evaluación de la factibilidad.

Lo que lleva a la conclusión que la base de datos MySql es la más apta con todos sus fundamentos para la construcción del sistema prototipo de Gestión de Ventas, Clientes, Personal y Control de Stocks, basado en Teledetección utilizando la tecnología RFID.

#### **3.2.2.3. Equipos RFID UHF DE 860-960 MHZ (identificación por radio frecuencia)**

En la actualidad existe una gran variedad de equipos de radio frecuencia, los cuales trabajan en usos específicos para diversas aplicaciones, estos equipos RFID han ido surgiendo debido a las tendencias y a las necesidades de las industrias en trabajos de identificación y sus derivados, por lo que existen equipos RFID de todo precio, de toda marca para cualquier aplicación a realizar.

##### **3.2.2.3.1. Lector RFID fijo de red PoE A-GEN2**

###### **3.2.2.3.1.1. Descripción del Lector fijo de red PoE A-GEN2**

Este es un poderoso lector RFID con soporte anti colisión y con posibilidades de conexión vía Ethernet (Inclusive con soporte para PoE) y WiFi, la capacidad de poder trabajar vía Ethernet (PoE), evita que se tenga que usar un extenso cableado.

La opción WiFi permite el obtener una conexión sin cables con seguridad TKIP, AES y EAPTLS, puede leer microchips RFID entre 865-956 MHz. (Observar Tabla 12).

El sistema incluye un servidor web embebido en el lector, lo cual facilita su operación y configuración, también posee opciones de 16MB y 32MB de flash

RAM, lo cual permite incluir aplicaciones tipo “hosted”, por lo que su rango de lectura es de aproximadamente 10 metros y puede leer hasta 190 etiquetas por segundo. (Ver Figura 25) <sup>41</sup>



Figura 25: “Lector RFID PoE A-GEN2”

Fuente: A-GEN 2 Lectores <http://www.rfidecuador.com>

Lector RFID fijo de red PoE A-GEN2	
Ventajas	Desventajas
<b>Su rango de lectura es superior a los 7 metros.</b>	Su precio es considerablemente elevado.
<b>Capacidad de leer más de 190 etiquetas o tags a la vez.</b>	Su configuración es compleja.
<b>Tiene WIFI para trabajar sin cables.</b>	No Incluye Driver de algunos lenguajes de programación como C#, java.
<b>Viene con un servidor web para facilitar su operatividad.</b>	No consta con antena integrada.
<b>Tiene interfaces de comunicación Ethernet.</b>	Es más para aplicaciones de teledetección en peajes o lugares

<sup>41</sup> Catalogo Lectores RFID 2011 Tomado de : <http://www.rfidecuador.com/>

	grandes y abiertos.
--	---------------------

**Tabla 12: “Principales ventajas y desventajas de Lector RFID fijo de red PoE A-GEN2”**

**Fuente:** Los Autores

### **3.2.2.3.2. Lector RFID Solutions Model U-RW-232**

#### **3.2.2.3.2.1. Descripción Del Lector RFID Solutions Model U-RW-232**

Este es un compacto lector RFID con soporte de multiprotocolo, con un estándar de frecuencia Europeo posibilidades de conexión vía RS 232 - 383 y consta con una antena integrada, la capacidad de trabajar con una antena integrada, evita tener que usar una antena independiente y la opción UHF RW permite escribir y leer en los tags, también puede leer microchips RFID entre 902 - 928 MHz. (Ver Figura 26).

Este Lector RFID incluye un paquete completo de software libre (VB, VC programa de demostración con código fuente), lo cual facilita su operación y configuración, también posee opciones de software compatible con la versión UHF-RW-MP-232-IP-4A, lo cual permite trabajar con estándares EPC G2 o ISO18000-6B, por lo que su rango de lectura es de aproximadamente 8 metros y puede leer hasta 80 etiquetas por segundo. (Observar Tabla 13).<sup>42</sup>

<sup>42</sup> Equipos RFID Lectores Tomado de : <http://www.rfidpoint.com/>



Figura 26: “Lector RFID Modelo U-RW-232”

Fuente: Lector 232 <http://www.rfidpoint.com>

Lector RFID Modelo U-RW-232	
Ventajas	Desventajas
Su rango de lectura es superior a los 4 metros.	Su precio es elevado.
Capacidad de leer más de 50 etiquetas o tags a la vez.	Sus drivers de conexión están en lenguajes obsoletos.
Soporte el estándar EPC G2 o ISO18000-6B.	No Incluye Driver de algunos lenguajes de programación actuales como C#, java.
Consta con antena integrina polarizada linealmente.	Su poder de transmisión es relativamente bajo < 20dbm...
Tiene interfaces de comunicación RS 232 y 383.	Es más para aplicaciones de teledetección corta, como acceso de puertas.

Tabla 13: “Principales ventajas y desventajas de Lector RFID Modelo U-RW-232

Fuente: Los Autores

### 3.2.2.3.3. Lector RFID UHF READER/WRITER versión única americana

#### 3.2.2.3.3.1. Descripción del Lector RFID UHF READER/WRITER versión única americana

Este es un lector RFID compacto y liviano con soporte de multiprotocolo, con un estándar de frecuencia Americano posibilidades de conexión vía RS 232 – RS 383 adicionalmente ID de memoria y consta con una antena integrada, La capacidad de poder trabajar con una antena integrada, evita tener que usar una antena independiente, esta antena es potente, la opción UHF Read/Write UHF tag de identificación y de memoria permite escribir y leer en los tags, también puede leer microchips RFID entre 902 - 928 MHz. (Ver Figura 27).

Este Lector RFID incluye un paquete completo de software libre (drivers de conectividad Delphi y C#), lo cual facilita su operación y configuración, también posee opciones de software compatible con la versión UHF-RW-MP-232-D2, lo cual permite incluir y trabajar con estándares EPC G2 o ISO18000-6B., por lo que su rango de lectura es de aproximadamente 7 metros y puede leer hasta 80 etiquetas por segundo. (Observar Tabla 14)<sup>43</sup>

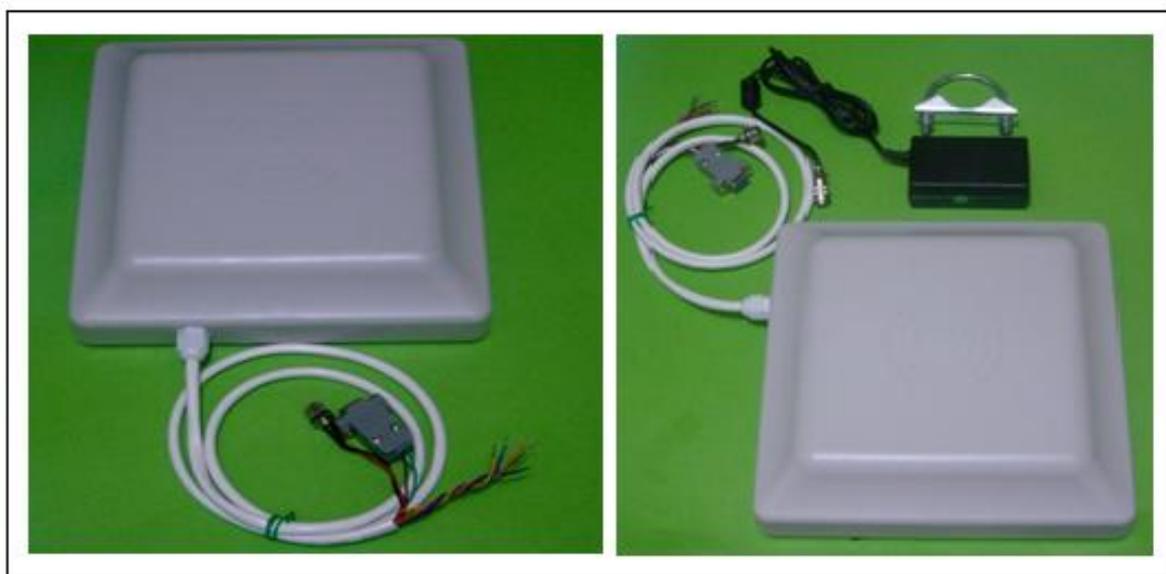


Figura 27: “Lector RFID UHF READER/WRITER “

Fuente: Lector RFID R/W [www.idconsultants.us](http://www.idconsultants.us)

<sup>43</sup> Equipos RFID UHF Tomado de : ID Consultants en <http://www.idconsultants.us/>

<b>Lector RFID UHF READER/WRITER</b>	
<b>Ventajas</b>	<b>Desventajas</b>
<b>Su rango de lectura es superior a los 5 metros.</b>	Su antena es de 7 dbi.
<b>Capacidad de leer más de 70 etiquetas o tags a la vez.</b>	No Incluye Driver para algunos lenguajes de programación.
<b>Consta con antena integrina polarizada circularmente.</b>	Su poder de transmisión no es muy alto < 30dbm.
<b>Su Modo de operación es lectura/escritura UHF etiqueta de identificación y de memoria gracias a su escritor con antena integrada.</b>	
<b>Es para trabajar en ambientes y lugares cerrados.</b>	
<b>Su precio es accesible.</b>	

**Tabla 14: “Principales ventajas y desventajas de Lector RFID UHF READER/WRITER”**

**Fuente:** Los Autores

#### **3.2.2.3.4. Calificación de la Factibilidad de los Lectores RFID analizados**

Después del análisis de factibilidad para los lectores RFID, se evaluaron todos los criterios tomados en cuenta para calificar la factibilidad de cada lector RFID. (Observar Tabla 15).

LECTORES RFID	RFID GEN 2	RFID U-RW-232	RFID UHF R/W
<b>Sistema de evaluación (1/10)</b>			
Costos	6	7	9
Conectividad con drivers	9	7	9
Rango de cobertura y de lectura	8	6	8
Características técnicas	9	7	8
<b>Riesgos (1/10)</b>			
Precisión en la captura de datos	8	8	9
Seguridad de la señales	9	9	9
Lectura rango de alcance	9	8	9
Velocidad de detección de tags	8	8	9
<b>Problemas (1/10)</b>			
Interferencia entre lectores 2 o mas	8	7	7
Estabilidad en la señal	9	9	9
Conectividad con otros equipos	8	7	8
<b>Puntaje Total /(110)</b>	<b>91</b>	<b>83</b>	<b>94</b>

Tabla 15: “Tabla de resultados para la elección del lector RFID”

Fuente: Los Autores

#### 3.2.2.3.4.1. Resultado de factibilidad sobre el lector RFID a utilizar

Justificados en la tabla 15, se establece como equipo o lector RFID ideal y ganador, en base al análisis de las calificaciones, al lector **RFID UHF READER/WRITER** considerando sus características, ventajas, desventajas, y por los resultados que arrojaron los indicadores de la evaluación de la factibilidad.

Lo que lleva a la conclusión que el lector RFID UHF READER/WRITER es el más apto con todos sus fundamentos para la construcción del sistema prototipo de Gestión de Ventas, Clientes, Personal y Control de Stocks, basado en Teledetección utilizando la tecnología RFID ya que trabajará con frecuencias de 902 a 928 MHz. y su costo es relativamente bajo y accesible.

#### **3.2.2.4. Tags pasivos RFID UHF DE 860-960 MHZ**

Para alcanzar la efectividad del software prototipo a realizar en el presente trabajo, se toma muy en cuenta, cual es el dispositivo mas optimo para la identificación de los productos y la comunicación del equipo RFID, los Tags RFID de identificación han ido surgiendo debido a las tendencias y necesidades de las industrias para que trabajaren en diversas aplicaciones, en si existen tags de todo precio, de toda marca, de varios tipos para cualquier aplicación según sea su utilidad.

Este tipo de tags RFID no tienen fuente de alimentación (batería) integrada, utilizan la energía emitida por el lector para autoalimentarse y transmitir su información almacenada al lector, por lo que se puede decir que las etiquetas pasivas son más simples de construir y no tienen partes móviles, por lo que, esto hace que tengan mayor vida y soporten condiciones ambientales extremas como corrosivas, o temperaturas de más de 200 °C.<sup>44</sup>

##### **3.2.2.4.1. Tags Pasivos RFID Adhesivos**

###### **3.2.2.4.1.1. Descripción de Tags Pasivos RFID adhesivos**

Los tags adhesivos, se los imprimen con una impresora especial adhesivos, su precio varía de acuerdo a la cantidad y al tipo de papel, adhesivo, según sea el rollo por unidades. (Ver Figura 28) y (Observar Tabla 16).



**Figura 28: “Tags RFID Pasivos adhesivo “**

<sup>44</sup> Tags RFID Tomado de : [http://www.dipolerfid.es/productos/RFID\\_tag/Clasificacion\\_RFID\\_tags.aspx](http://www.dipolerfid.es/productos/RFID_tag/Clasificacion_RFID_tags.aspx)

Tags RFID Adhesivos	
Ventaja	Desventaja
No muy resistente a climas tropicales, altas temperaturas	Su precio es considerablemente elevado.

Tabla 16: “Principales ventajas y desventajas de Tags Adhesivos RFID”

Fuente: Los Autores

### 3.2.2.4.2. Tags Pasivos RFID de Plástico con circuito

#### 3.2.2.4.2.1. Descripción de Tags de Plástico con circuito

Los tags de plástico tiene un circuito para que su alcance de detención sea mucho más largo, estos tags no tiene cubierta, su precio es relativamente alto según varia el circuito, su tamaño general es diverso según la de construcción. (Ver Figura 29) y (Observar Tabla 17)



Figura 29: “Tags RFID Pasivos de Pastico con circuito”

Fuente: Tags con circuitos [www.idconsultants.us](http://www.idconsultants.us)

Tags RFID Plástico con circuito	
Ventaja	Desventaja
Demasiado pesado y muy frágil	Su precio es elevado.

Tabla 17: “Principales ventajas y desventajas de Tags pasivos RFID de Plástico con circuito”

Fuente: Los Autores

### 3.2.2.4.3. Tags Pasivos RFID de Mica Imprimibles

#### 3.2.2.4.3.1. Descripción de Tags Pasivos RFID de Mica Imprimibles

Los tag pasivos RFID de mica, son los más comunes y más utilizados en diferentes aplicaciones RFID, su costo es económico, estos tag soportan todo tipo de trato y vienen en diferentes tamaños con la opción de imprimirlos según sea la aplicación o su uso. (Ver Figura 30) y (Observar Tabla 18).



Figura 17: “Tags RFID Pasivos de Mica”

Fuente: TAGS RFID [www.idconsultants.us](http://www.idconsultants.us)

Tags Adhesivos RFID	
Ventaja	Desventaja
Son resistente a todo factor extremo natural	Su precio es económico.
Es liviano delgado y imprimible	No son blandos

Tabla 18: “Principales ventajas y desventajas de Tags Pasivos RFID de Mica”

Fuente: Los Autores

#### 3.2.2.4.4. Calificación de criterios de los Tags RFID analizados

Lo único que varía de estos tags pasivos RFID es su costo, su diseño, su resistencia y su tamaño ya que la funcionalidad es la misma, en si estos tags son muy utilizados en diversas aplicaciones industriales, comerciales, etc. Para identificación de objetos específicos en varias situaciones<sup>45</sup>. (Observar Tabla 19).

TAGS RFID	TAGS ADHESIVO	TGAS CIRCUITO	TGAS MICA
<b>Sistema de evaluación (1/10)</b>			
Costos	8	7	9
Soporte condiciones ambientales	8	7	8
Rango de detención	9	9	9
Peso	10	6	9
<b>Riesgos (1/10)</b>			
Nivel de detención	8	7	9
Seguridad de las señales	9	9	9
Factores climáticos	8	7	9
Duración de tags	9	7	8
<b>Problemas (1/10)</b>			
Resistencia a ambientes externos	9	7	9

<sup>45</sup> Tags RFID UHF Tomado de : ID Consultants en <http://www.idconsultants.us/>

Conectividad con lectores RFID	9	8	9
Capacidad de lectura y escritura	9	7	9
<b>Puntaje Total /(110)</b>	<b>96</b>	<b>81</b>	<b>97</b>

Tabla 19: “Tabla de resultados para la elección de tags pasivos RFID”

Fuente: Los Autores

#### 3.2.2.4.4.1. Resultado de factibilidad sobre el Tag RFID a utilizar

Justificados en la tabla 19, se establece como tag pasivo RFID ideal y ganador, en base al análisis de las calificaciones, a **LOS TAGS PASIVOS RFID DE MICA**, considerando sus características, ventajas, desventajas, y por los resultados que arrojaron los indicadores de la evaluación de la factibilidad.

La elección de estos tags fue difícil ya que todos cumplen con las especificaciones que se necesitan para elaborar el software prototipo, por estas razones:

- Su bajo costo y peso.
- Su buen rendimiento y su rango de detección.

Se decidió que los tags pasivos RFID de Mica son los más aptos con todos sus fundamentos para la construcción del sistema prototipo de Gestión de Ventas, Clientes, Personal y Control de Stocks, basado en Teledetección utilizando la tecnología RFID.

### **3.3. FACTIBILIDAD TÉCNICA DEL PROYECTO**

La factibilidad técnica es la más importante cuando se trabaja con equipos tecnológicos nuevos y actuales, que no son comúnmente utilizados en la elaboración de software de este tipo, por lo que tomamos a consideración los siguientes requisitos:

#### **3.3.1. Requisitos Técnicos de Equipos de Computación**

Para un funcionamiento normal y sin mucha carga en el sistema, los requisitos mínimos con los que deberían contar los equipos de computación en donde trabajará la base de datos, otros competes y el software prototipo de Gestión de Ventas, Clientes, Personal y Control de Stocks, basado en Teledetección utilizando la tecnología RFID serán los siguientes:

- Procesador Pentium III 600 MHz o superior.
- Disco Duro mínimo 40 Gb. disponible o superior.
- Mínimo 512 Mb en RAM, se recomienda 1 Gb. o superior.
- Sistema Operativo Windows XP profesional Service Pack 2 o Superior instalado<sup>46</sup>.
- Monitor y Adaptador de video.
- Resolución gráfica mínimo 800\*600.
- Unidad de CD-ROM o DVD ROM.
- Teclado multimedia, Mouse óptico, Altavoces o Parlantes.
- Navegador de Internet (recomendable Internet Explorer Versión 6.0 o superior).
- Acrobat Reader 5.0 o superior.
- Conexión a Internet.

---

<sup>46</sup> Requisitos para sistema operativo Windows XP Tomado de : [www.microsoft.com](http://www.microsoft.com)

- Conexión puerto USB versión 2 o superior.
- Conexión de interfaz RJ45 / RS232 / RS485.
- Impresora a color, se recomienda láser.

### **3.3.2. Requisitos Técnicos de Equipos RFID (Identificación por Radio Frecuencia)**

Para que el equipo RFID trabaje de forma óptima, de esta manera pueda funcionar a la par con el sistema prototipo, y tenga una buena conectividad con los equipos de computación para automatizar los módulos de Ventas, Clientes, Personal y Control de Stocks, basado en Teledetección, los requisitos mínimos serán:

- Antena de 7 dbi. polarizada circularmente o superior, se recomienda integrada en el lector<sup>47</sup>.
- Lector que soporte lectura de tag múltiple hasta 10 tags, o superior se recomienda 20 tags.
- Lector que soporte frecuencia 860Mhz o superior, se recomienda 902 MHz.
- Lector que lea y Escriba tags UHF.
- Rango o Distancia de lectura 1 metro o superior, se recomienda 3 Metros.
- Interfaz de comunicación RS232 o superior.
- Fuente de alimentación +DC 9 Voltios.
- Cumpla con el estándar EPC G2 o ISO18000-6B.

---

<sup>47</sup> Equipos RFID UHF Tomado de: Catalogo 2011 Id. Consultants [www.idconsultants.us](http://www.idconsultants.us)

### 3.3.3. Requisitos Técnicos de Tags RFID

La identificación, lectura y escritura de los tags es importante en el sistema prototipo, para identificar productos, empleados y robos por medio de radio frecuencia en teledetección con ayuda del Lector RFID de esta forma el sistema prototipo funcionara correctamente con estos requisitos mínimos:

- Tag o Mifare pasivo que cumpla estándar EPC G2 o ISO18000-6B.
- Dimensiones del tag o Mifare 2\*4 cm. o superior, se recomienda 5\*9 cm. o adhesiva de 2.5\*2.5 cm.
- Energía y datos integrados en un mismo módulo.
- Chip 1ICS50 o superior para la energía y los datos<sup>48</sup>.
- Rango de alcance UHF 860MHz o superior.
- Mifare RW o tags RW.
- Tag o Mifare de 2 bits o superior, se recomienda de 4 bits.

### 3.3.4. Argumentación Técnica

Es factible técnicamente la construcción del Sistema Prototipo, porque con los equipos que se realizará la construcción y sus pruebas respectivas tanto de computación como de radio frecuencia, son actos y reúnen todos los requerimientos básicos para el funcionamiento de los programas a utilizar, y la conectividad del equipo RFID, tags y la PC para que en la ejecución del sistema prototipo nos den los efectos, los rendimientos requeridos y deseados.

---

<sup>48</sup> Tags RFID UHF Tomado de: Catalogo 2011 Id. Consultants [www.idconsultants.us](http://www.idconsultants.us)

### **3.4. FACTIBILIDAD ECONÓMICA DEL PROYECTO**

#### **3.4.1. Análisis Costo / Beneficio**

Para la elaboración del software prototipo de Gestión de Ventas, Clientes, Personal y Control de Stocks, basado en la tecnología RFID por medio de teledetección, cuyo objetivo principal que es el pago ágil, inmediatos y automático de los productos que el cliente desea adquirir, como también registrar la hora y salida de los empleados y por ultimo monitorear robos de los productos.

Para comprobar y probar que el software a desarrollar es viable, se realizará un estudio minucioso del costo beneficio, en donde se tomarán en cuenta temas fundamentales en los que se basa la implementación del software prototipo a construir, de esta forma nos permitirá reconocer la factibilidad económica del sistema, en el mercado de las industrias tecnológicas que trabajen con tecnologías RFID.

Ahora se realizará el estudio o análisis costo/beneficio de la construcción del sistema prototipo que trabajará con la tecnología RFID, en un el período de un año aproximadamente<sup>49</sup>.

Estos valores que se analizarán a continuación en el estudio costo/beneficio son estimados de los posibles valores que el software tendría si fuese comercializado, en locales comerciales. (Observar Tabla 20).

---

<sup>49</sup> Análisis costo/beneficio Tomado de: [http://boards5.melodysoft.com/Costo\\_Beneficio/analisis-costobeneficio-6.html](http://boards5.melodysoft.com/Costo_Beneficio/analisis-costobeneficio-6.html)

<b>ANALISIS COSTO/BENEFICIO</b>			
<b>Costos</b>		<b>Beneficios</b>	
<b>Descripción</b>	<b>Valor (Dólares)</b>	<b>Descripción</b>	<b>Valor (Costo)</b>
<b>Nuevos Equipos</b>		<b>Mejor Negocio</b>	
Computadora de Escritorio (Especificaciones Altas y Rendimiento estable)	\$ 400.00	Mayor agilidad en trámites, reportes, etc.	\$ 150.00
Equipo RFID (Antena integrada, capacidad de lectura y escritura, alcance de señal de 3 a 7 metros, frecuencia UHF de 860 a 960 MHz.)	\$ 600.00	Mejorada atención al cliente	\$ 150.00
Tag RFID (pasivos, de largo alcance UHF, resistentes a condiciones climatológicas variadas, 8 de diversos tipos y normas)	\$ 1.20.00 c/u \$ 33.60 En 28 Tags	Nuevos clientes afiliados	\$ 400.00
Impresora (de última generación, velocidad en impresión)	\$ 120.00	Nuevos clientes no afiliados	\$ 200.00
Sistema Operativo Microsoft Windows XP (Service pack 2 o superior, licencia exprés para estudiantes e investigadores )	\$150.00	Mayor rapidez en facturación y atención más ventas	\$ 600.00
<b>Otros</b>			
Publicidad del nuevo Servicio a ofrecer	\$ 30.00	Mejor control a los empleados aumento de productividad	\$ 300.00
Capacitación todo el personal	\$ 50.00	Menor pérdida de productos por robos de adentro o de afuera	\$ 60.00
Pérdida de clientes que no les agrada el servicio (Forma de facturación, ventas, etc.)	\$200.00		
<b>Costo Totales</b>	<b>\$ 1583.60</b>	<b>Beneficios Totales</b>	<b>\$ 1860.00</b>

Tabla 20: “Costo/Beneficio del “Sistema Prototipo de Gestión de Ventas, Clientes, Personal y Control de Stocks, basado en Teledetección utilizando la tecnología RFID”“

Fuente: Los Autores

### 3.4.1.1. Análisis de Resultados

En este análisis los resultados son muy alentadores según la tabla 12 con la idea de tener una nueva forma de pago, control de productos empleados, incluida una nueva tecnología, y los cálculos demostraron un beneficio básico para el primer año (\$276,40), sin embargo, la relación de beneficios a costos es de \$1.17 de retorno por cada dólar gastado (\$1860.00/\$1583.60). Este sería un retorno positivo, ya que se trata de un prototipo.

### 3.4.1.2. Punto de Equilibrio

El punto de equilibrio es una de las formas más sencillas de hacer el análisis de Costo / Beneficio. El punto de equilibrio es el tiempo que tomaría para que el total de los beneficios totales sea igual al costo total, en este caso de 1 año (12 meses). De esta forma: (Observar Tabla 21).

Punto de Equilibrio	
<b>Datos</b>	
<b>Costo total: \$ 1583.60</b>	<b>Beneficios Totales: \$ 1860.00</b>
<b>Formula</b>	
$PE = [(Costo Total / Beneficio Total) * (12 meses)]$ $PE = [(1583.60.00 / 1860.00) * (12)]$ $PE = 10.21$	
<b>El Punto de Equilibrio es: 10.21</b>	

Tabla 21: “Punto de equilibrio del “Sistema Prototipo de Gestión de Ventas, Clientes, Personal y Control de Stocks, basado en Teledetección utilizando la tecnología RFID””

Fuente: Los Autores

Entonces según la tabla 21 los resultados son alentadores porque a partir del 1 primer año, se empezará a obtener utilidades, pese a ser un prototipo pero esta utilidad no sería muy alta, ni notoria.

### 3.4.1.3. Periodo de Devolución

El periodo de devolución es el tiempo requerido para recuperar el monto inicial de una inversión, donde se calcula la cantidad de tiempo que se tomaría para lograr un flujo positivo igual a la inversión total requerida para la construcción del Sistema Prototipo de Gestión de Ventas, Clientes, Personal y Control de Stocks, basado en Teledetección utilizando la tecnología RFID, al Igual que el Análisis del Punto de Equilibrio, el análisis del período de devolución no tiene en cuenta el valor del dinero en el tiempo. (Observar Tabla 22).

Periodo de Devolución		
<b>Datos</b>		
<b>Costo total:</b> \$ 1583.60	<b>Beneficios Totales:</b> \$ 1860.00 tiempo estimado de un año (12 meses)	<b>Valor asegurado:</b> \$ 600.00
<b>Formula</b>		
$PE = [(Costo\ Total - Valor\ asegurado) / (Beneficio\ Total) * (12\ meses)]$ $PE = [(1583.60.00 - 600) / (1860.00) * (12)]$ $PE = 6.34$		
<b>El Periodo de Devolución es: 6.34</b>		

Tabla 22: “Periodo de Devolución del “Sistema Prototipo de Gestión de Ventas, Clientes, Personal y Control de Stocks, basado en Teledetección utilizando la tecnología RFID””

Fuente: Los Autores

Entonces según la tabla 22 la variación del valor del dinero no es tomada en cuenta, este análisis solo verifica el tiempo en el que la inversión inicial sea igual a los beneficios. Los costos de operación para la construcción del sistema prototipo son equilibrados con los beneficios, por lo que el proyecto es provechoso y no se lo rechazarán si fuese comercializado.

### 3.4.2. Valor Actual Neto (VAN)

El VAN es un indicador financiero que mide los flujos de los futuros ingresos y egresos que tendrá el Sistema Prototipo de Gestión de Ventas, Clientes, Personal y Control de Stocks, basado en Teledetección utilizando la tecnología RFID, para determinar, si luego de descontar la inversión inicial, quedaría alguna ganancia. Si el resultado es positivo, la construcción del sistema es viable.

Según los resultados del VAN se verifican de la siguiente forma:

- VAN > 0 → el proyecto es rentable.
- VAN = 0 → el proyecto es rentable también, porque ya está incorporado ganancia de la tasa de descuento.
- VAN < 0 → el proyecto no es rentable.

Si el proyecto tiene una inversión de 1800 y una tasa de descuento (TD) de 12%. El flujo de caja neto estimado sería. (Observar Tabla 23)

Años	Año 1	Año 2	Año 3
Flujo de caja neto	\$ 500.00	\$ 600.00	\$ 800.00

Tabla 23: “Flujo de caja neto en 3 años “Sistema Prototipo de Gestión de Ventas, Clientes, Personal y Control de Stocks, basado en Teledetección utilizando la tecnología RFID””

Fuente: Los Autores

El beneficio neto nominal sería de:

**\$ 1900 (\$ 500 + \$ 600 + \$ 800)**

Por lo que la utilidad lógica sería \$ 100 (\$ 1900 – \$ 1800), pero este beneficio o ganancia no sería real en la construcción del sistema prototipo porque no se

estaría considerando el valor del dinero en el tiempo, por lo que cada periodo se deberá actualizarlo a través de una tasa de descuento (tasa de rentabilidad mínima que se espera ganar). (Observar Tabla 24).

<b>Valor Actual Neto (VAN)</b>		
<b>Datos</b>		
<b>Inversión: \$ 1583.60</b> <b>Recomendada: \$</b> <b>1800</b>	Beneficio neto actualizado BNA proyectado: \$ 500.00 a \$ 800.00 tiempo estimado de 1 a 3 años	Tasa de descuento: 12 % según el Banco Central del Ecuador
<b>Formula</b>		
<b><math>VAN = [(BNA - Inversión)]</math></b> Donde el BNA es el valor actual del flujo de caja o beneficio neto proyectado, el cual ha sido actualizado a través de una tasa de descuento.  <b><math>VAN = [(500/(1+0,12^1)+600/(1+0,12^2)+800/(1+0,12^3)) - (1800.00)]</math></b> <b><math>VAN = 36.531</math></b>		
<b>El Valor Actual Neto es: 36.531</b>		

Tabla 24: “VAN para “Sistema Prototipo de Gestión de Ventas, Clientes, Personal y Control de Stocks, basado en Teledetección utilizando la tecnología RFID””

Fuente: Los Autores

En la tabla 24, se demuestra que el proyecto brindaría una mayor ganancia adicional, pero cabe recalcar que es un prototipo con costos mínimos y no con todas sus funcionalidades.

### 3.4.3. Tasa interna de Retorno (TIR)

La TIR es la tasa de descuento que tendrá el Sistema Prototipo de inversión que permite que el beneficio neto actualizado sea igual o superior a la inversión para la construcción del sistema, es la máxima tasa de descuento que puede tener el proyecto para que sea rentable, una mayor tasa ocasionaría que el beneficio neto actualizado sea menor que la inversión. (Observar Tabla 25).

<b>Tasa Interna de Retorno (TIR)</b>	
<b>Datos</b>	
<b>Tamaño de la Inversión: \$ 1583.60</b> <b>Recomendada: \$ 1800</b>	Beneficio neto actualizado BNA proyectado: \$ 500.00 a \$ 800.00 tiempo estimado de 1 a 3 años
<b>Formula</b>	
<p>Para hallar la TIR hacemos uso de la fórmula del VAN, sólo que en vez de hallar el VAN (el cual reemplazamos por 0), estaríamos hallando la tasa de descuento:</p> <p><math>i = 12 \%</math></p> <p><math>VAN = [(BNA - Inversion)]</math></p> <p><math>0 = [(500/(1+i^1)+600/(1+i^2)+800/(1+i^3)) - (1800.00)]</math></p> <p><math>TIR = 3 \%</math></p>	
<b>La Tasa Interna de Retorno es: 3 %</b>	

**Tabla 25: TIR para “Sistema Prototipo de Gestión de Ventas, Clientes, Personal y Control de Stocks, basado en Teledetección utilizando la tecnología RFID”**

Fuente: Los Autores

La tabla 25, da como resultado una tasa del 3 % por lo que a menor tasa, el sistema prototipo sería cada vez más rentable, obviamente con costos mínimos y no con todas sus funcionalidades, pues el beneficio neto actualizado sería cada vez mayor que la inversión.

#### **3.4.4. Argumentación del Estudio Costo/Beneficio**

Después del análisis costo beneficio y evaluar los resultados que arrojó este análisis, se puede verificar por medio del estudio realizado que se ha producido un beneficio dentro de un ciclo determinado, es decir que la construcción del proyecto prototipo de Gestión de Ventas, Clientes, Personal y Control de Stocks, basado utilizando la tecnología RFID por medio de teledetección es admisible y posible, ya que son muchos los beneficios, rebasando así a los valores indispensables para la elaboración del software prototipo antes

mencionado. (Ver Figura 31).

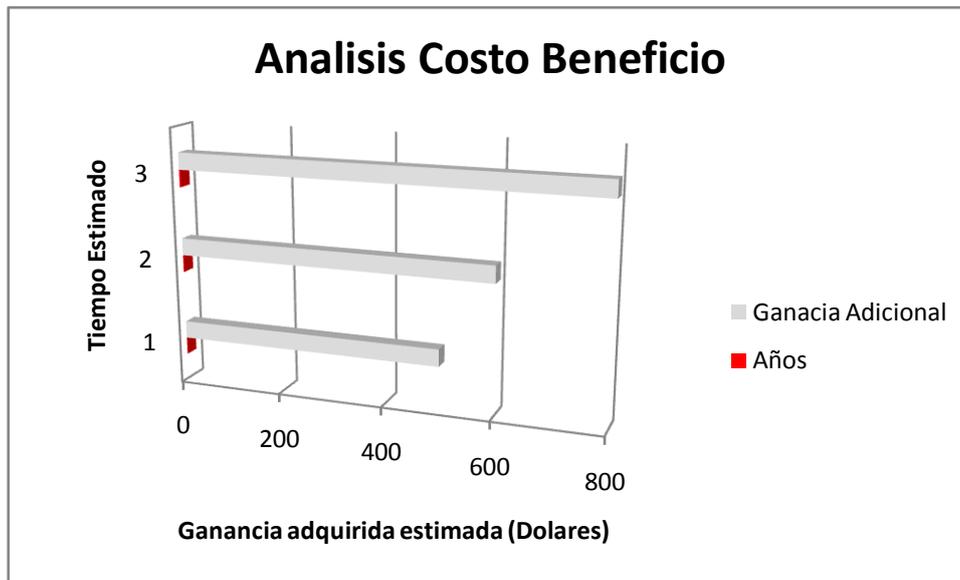


Figura 30: “Estadística de “Sistema Prototipo de Gestión de Ventas, Clientes, Personal y Control de Stocks, basado en Teledetección utilizando la tecnología RFID” en tres años”

Fuente: Los Autores

# CAPÍTULO IV

## 4. DISEÑO Y CONSTRUCCIÓN DEL SISTEMA PROTOTIPO

En este capítulo, se presentará el diseño en sustento a todo lo estudiado y analizado en la elaboración del sistema prototipo de Gestión de Ventas, Clientes, Personal y Control de Stocks basado en Teledetección utilizando la tecnología RFID (Identificación por Radio Frecuencia), donde mediante la utilización de UML (lenguaje de modulado Unificado), se expondrán los diversos diagramas para tener una interpretación bastante clara de la funcionalidad del sistema, el mismo que se lo denominara “RFIDBussinesSoft” desde este punto en adelante.

También en esta misma etapa se introducirá el modelo de la interfaz, sus colores, estilos, etc. Donde los usuarios de “RFIDBussinesSoft” interactuaran y trabajarán según sea su perfil específico.

De la misma forma se mostrarán los módulos de uno en uno que se construyeron para la elaboración de “RFIDBussinesSoft”.

### 4.1. DISEÑO CONCEPTUAL

Mediante el uso y la construcción del modelo conceptual de la base de datos para la construcción de “RFIDBussinesSoft”, se entenderá los datos y todo el flujo de información que se manipulará y operara en el sistema prototipo, por lo que es importante resaltar que la base de datos fue elaborada, diseñada y probada en el gestor de base de datos relacional DBDesigner4.0.5.6. (Ver Figura 32)





#### 4.2.2. DIAGRAMA DE CASO DE USO

En este diagrama se mostrarán las diferentes acciones que efectuarán los usuarios de “RFIDBussinesSoft” al ejercer una interacción con el mismo las cuales son:

**Usuario Administrador:** Este usuario tendrá el control completo del sistema “RFIDBussinesSoft”, donde él controlara las diferentes gestiones que brinda el sistema, creará los usuarios secundarios de acuerdo a su rol asignará permisos según el trabajo que desempeñe en el sistema.

A continuación, se muestran todas las acciones del diagrama de caso de uso especificadas, para una mayor comprensión de las operaciones que se realizan dentro del sistema “RFIDBussinesSoft”.

- **Presentación de “RFIDBussinesSoft”.-** en esta acción el usuario visualizara el logo del sistema, una barra de tiempo que indica cómo cargar todos sus componentes antes de iniciar el mismo.
- **Ingresar Usuario y Contraseña.-** en esta acción cada usuario se identificará antes de ingresar al menú principal de “RFIDBussinesSoft”. Esta acción es obligatoria para cualquier usuario del sistema sea cual sea su cargo o su rol dentro del mismo.
- **Gestión de productos.-**, en esta etapa el usuario relazará diversas acciones como:
  - Operación de control de stock donde se pueden realizar ingresos, consultas, actualizaciones y reportes de productos.
  - Operación de Activación de RFID donde se puede configurar el lector RFID, para abrir o cerrar un puerto de manera automática que o que previamente elija.
  - Operación de Activar Tags es cuando se activan los tags de identificación RFID para trabajar con los mismos.
  - Operación de generar tag de identificación para asignar un tag a cada producto ingresado.

- **Gestión de Clientes.-** en la etapa referente a los clientes el usuario realizara diversas acciones como:
  - Operación de Control de clientes, se pueden realizar ingresos, consultas, actualizaciones y reportes de Clientes.
- **Gestión de Ventas.-** esta etapa, es una de las más importantes del sistema ya que el usuario realizará acciones importantes como:
  - Operación de Facturación donde se facturara las compras que realicen los clientes, ingresando datos de los clientes, de los productos y del vendedor.
  - Operación de Activación de RFID donde se puede configurar el lector RFID, para abrir o cerrar un puerto de manera automática que o que previamente elija.
  - Operación de Activar Tags es cuando se activan los tags de identificación RFID para trabajar con los mismos.
- **Gestión de Empleados.-** es la etapa donde se gestionaran a los empleados se trabajará con varias actividades como:
  - Operación de Gestión de Empleados en el que se puede realizar ingresos, consultas, actualizaciones y reportes de todos los empleados.
  - Operación de Activación de RFID donde se puede configurar el lector RFID, para abrir o cerrar un puerto de manera automática o que previamente se elija.
  - Operación de Activar Tags es cuando se activan los tags de identificación RFID para trabajar con los mismos.
  - Operación de seleccionar empleado para la asignación de un tag de identificación para su gestión.
- **Gestión de Usuarios.-** en esta etapa se trabajará con varias operaciones de asignación y gestión como:

- Operación de Control de usuarios: se puede realizar ingresos, consultas, actualizaciones y reportes de todos los usuarios del sistema “RFIDBussinesSoft”.
  - Operación permisos: el usuario administrador tendrá la facultad de asignar permisos a los usuarios secundarios de todas las operaciones de “RFIDBussinesSoft” según sea el rol que ocupe dentro del sistema.
- **Control de Empleados.-** esta etapa es un control de horarios de entrada y salida de los empleados al lugar de trabajo que le corresponde con estas acciones:
  - Operación Control automático donde el sistema “RFIDBussinesSoft” detectará automáticamente el ingreso y la salida del empleado de acuerdo a políticas de la empresa en cuestión de horas.
  - Operación de Activación de RFID donde se puede configurar el lector RFID, para abrir o cerrar un puerto de manera automática o que previamente se elija.
  - Operación de Activar Tags es cuando se activan los tags de identificación RFID para trabajar con los mismos.
- **Gestión de Robos.-** esta etapa es un control antirrobo o de sustracciones de productos, esta gestión trabaja con estas acciones:
  - Operación control antirrobo donde se verificar en tiempo real que producto está siendo sustraído ilegalmente del local comercial.
  - Operación de Activación de RFID donde se puede configurar el lector RFID, para abrir o cerrar un puerto de manera automática o que previamente se elija.
  - Operación de Activar Tags es cuando se activan los tags de identificación RFID para trabajar con los mismos.

- Operación Reportes estos reportes serán para verificar que productos son los que más tentativas de robo tienen.
- **Gestión de Proveedores.-** en la etapa referente a los proveedores se realizara diversas acciones como:
  - Operación de Control de clientes se puede realizar ingresos, consultas, actualizaciones y reportes de Clientes de proveedores.
- **Cambiar Usuario.-** en esta etapa se puede cambiar de usuario, para diversas actividades de trabajo.

Es importante resaltar que los usuarios secundarios son creados únicamente por el administrador del sistema “RFIDBusinessSoft” el mismo que es el encargado de dar los permisos y habilitar todas las acciones que relazará en el sistema según su roll o trabajo, estas son las diferentes acciones que pueden ser habilitadas para cada usuario secundario:

- Gestiona de Productos
- Gestiona de Clientes
- Gestiona de Empleados
- Gestión de Ventas
- Gestión de Usuarios
- Gestión de Proveedores
- Gestión de Robos
- Control de Empleados

Por lo que los casos de uso de los usuarios secundarios son según su roll o su respectivo permiso que le asignen, ya que los mismos no tendrán todas las operaciones, si no solo las que necesite para su trabajo dentro del sistema “RFIDBusinessSoft”, dependiendo de su cargo en la empresa. (Ver Figura 34).

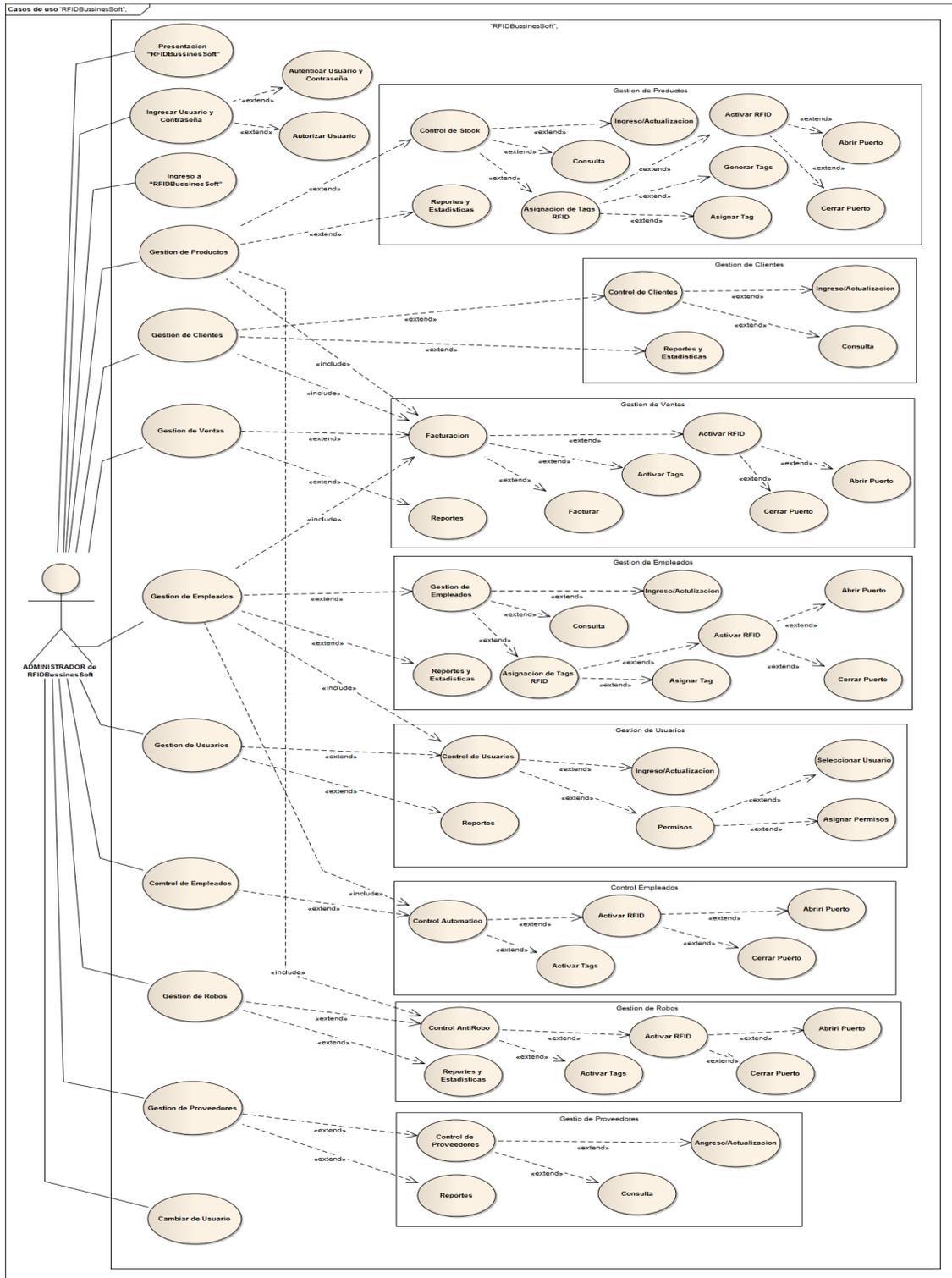


Figura 33: “Diagrama de Caso de Uso “RFIDBusinessSoft” desarrollado en Enterprise Architect 7.5”

Fuente: Los Autores

Para visualizar, analizar y estudiar de manera más amplia el diagrama de caso de uso del sistema “RFIDBusinessSoft” se recomienda (Revisar anexo 3).

### 4.2.3. DIAGRAMA DE ACTIVIDADES

A continuación se podrá analizar mediante este diagrama los flujos de trabajo paso a paso de negocio y operacionales de los componentes de “RFIDBussinesSoft”, también se observará el flujo de control general y la jerarquía dentro del sistema. (Ver Figura 35).

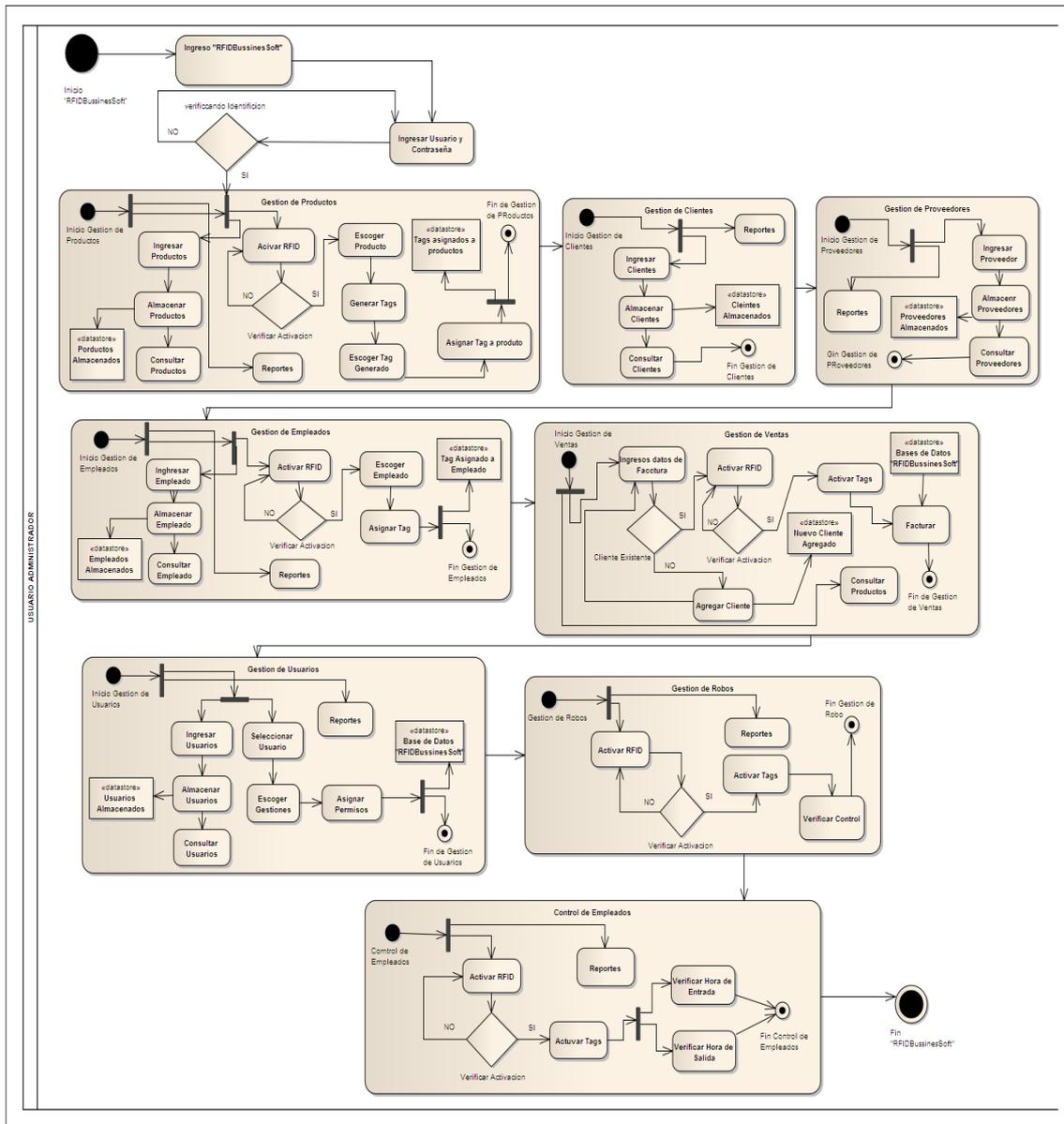


Figura 34: “Diagrama de Actividades “RFIDBussinesSoft” desarrollado en Enterprise Architect 7.5”

Fuente: Los Autores

Para visualizar, analizar y estudiar de manera más amplia el diagrama de Actividades del sistema “RFIDBussinesSoft” se recomienda (Revisar anexo 4).



#### 4.2.5. DIAGRAMA DE COMPONENTES

Mediante la elaboración de este diagrama se puede verificar los componentes principales y la estructura del sistema “RFIDBussinesSoft”, ya que un componente es la materialización de una o más clases y se pueden agrupar en paquetes así como los objetos en clases. (Ver Figura 37).

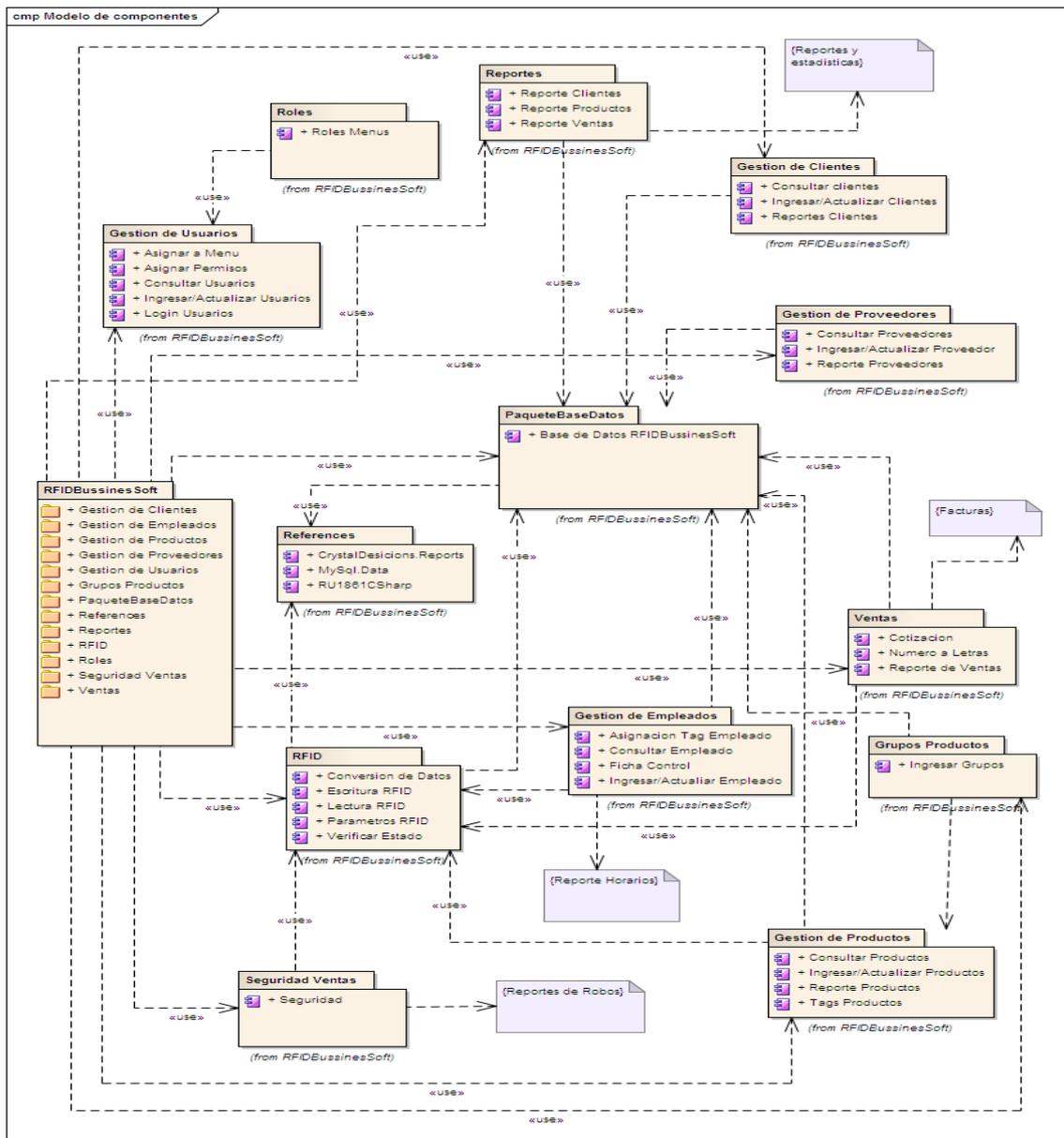


Figura 36: “Diagrama de Componentes “RFIDBussinesSoft” desarrollado en Enterprise Architect 7.5”

Fuente: Los Autores

Para visualizar, analizar y estudiar de manera más amplia el diagrama de Secuencia del sistema “RFIDBussinesSoft” se recomienda (Revisar anexo 6).

### 4.3. DISEÑO DE INTERFACES

En el diseño de las interfaces se sostendrá como base fundamental la usabilidad y el manejo apropiado del sistema “RFIDBussinesSoft”, para ello se basará en tres puntos fundamentales que son:

- Dar el control a los usuarios, es decir, que reaccione el sistema a las necesidades de estos.
- Reducir la carga en la memoria del usuario, es importante que la interfaz de usuario este bien diseñada para ayudar al usuario con un escenario de interacción que le facilite el uso de la memoria.
- Lograr que la interfaz sea consistente, para obtener un estándar de diseño, donde los mecanismo para ir de una tarea a otra sean definidos y consistentes.

A continuación se muestra el modelo de las interfaces diseñadas para cada acción en el sistema:

**Presentación del Sistema.-** En esta interfaz muestra una presentación rápida del sistema mientras se cargan sus componentes. (Ver Figura 38).



Figura 37: “Presentación de “RFIDBussinesSoft” ”

Fuente: Los Autores

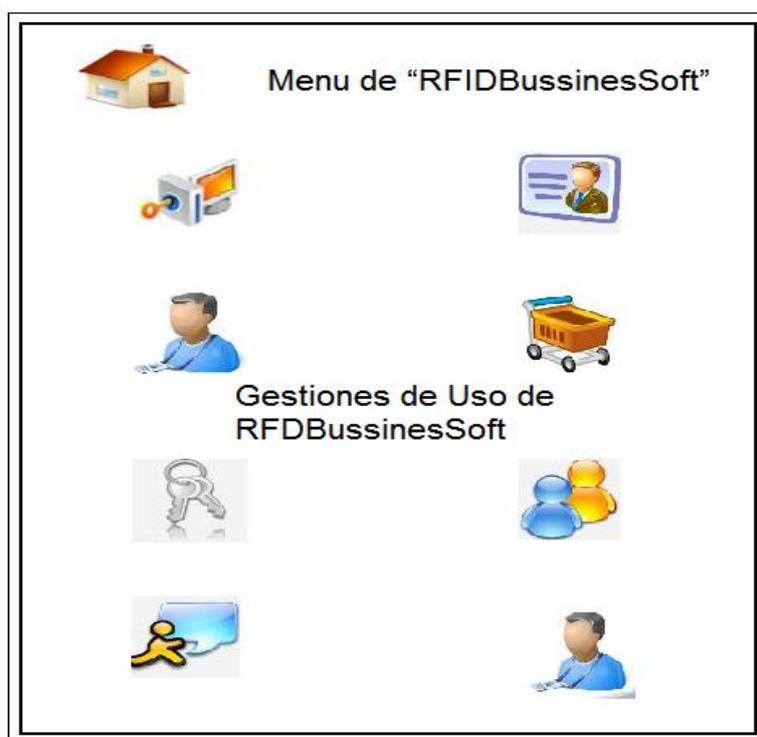
**Ingreso al Sistema “RFIDBussinesSoft”.-** Es donde los usuarios se identifican para ingresar al sistema, dependiendo de sus permisos. (Ver Figura 39).



**Figura 38: “Ingreso a “RFIDBussinesSoft” ”**

**Fuente:** Los Autores

**Menú Principal.-** es donde, se encuentran las diferentes gestiones y operaciones que ofrece el sistema. (Ver Figura 40).



**Figura 39: “Menú de “RFIDBussinesSoft” ”**

**Fuente:** Los Autores

**Interfaz Estándar de trabajo y operaciones.-** aquí se trabajará todo lo referente a las actividades, esta interfaz tiene un mismo estándar, para agilizar el manejo del sistema. (Ver Figura 41).



**Figura 40: “Interfaz Estándar de Trabajo y Operaciones de “RFIDBusinessSoft” ”**

**Fuente:** Los Autores

A través de la Figura 41, se detalla cada una de las opciones que se encontrarán en la interfaz:

- **Información Operaciones y Formularios.-** Es donde se trabajarán con la información y las operaciones del sistema de acuerdo al rol que tenga cada usuario.
- **Menú de Acción y Retorno.-** A través de este menú se podrá realizar varias acciones como ingresos, activaciones, retornos al menú, etc.
- **Menú de Opciones.-** Este menú lleva a diferentes opciones dentro de la misma interfaz, donde encontramos ingresos, consultas, etc. Sin perder el contexto del menú de Acción y Retorno.

#### 4.4. CREACIÓN DE MÓDULOS DEL SISTEMA RFIDBUSSINESSOFT

El sistema RFIDBussinesSoft está desarrollado primordialmente por carpetas o conocidos por otros lenguajes como paquetes, los mismos que contienen diversas clases para diferentes operaciones que son utilizadas por cada una de las interfaces. (Ver Figura 42).

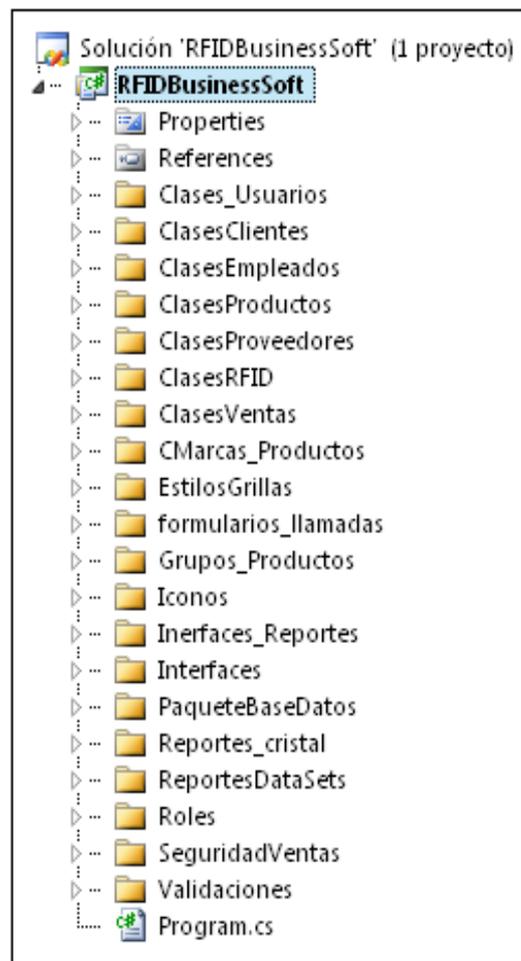


Figura 41: “Paquetes o Carpetas de “RFIDBussinesSoft” ”

Fuente: Los Autores

Como muestra la figura 42, el sistema “RFIDBussinesSoft”, está construido por capas, cada capa tiene diferentes clases que son para las acciones del sistema, también están separados los formularios en una sola capa o carpeta para su mejor uso dentro del sistema.

#### 4.4.1. Paquete Clases Usuario

En este paquete se podrá realizar Ingresos, actualización y consultas de usuarios, como también asignar permisos para trabajar en el sistema dependiendo de su rol, igualmente nos permite validar contraseñas para verificar su ingreso. (Ver Figura 43).

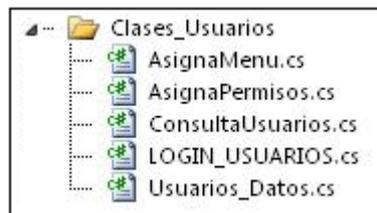


Figura 42: “Paquetes o Carpetas Clases\_Usuarios de “RFIDBussinesSoft” ”

Fuente: Los Autores

#### 4.4.2. Paquete Clases Clientes

Este paquete se usa para realizar Ingresos, actualización y consultas de clientes, como también reportes para verificar el número de clientes. (Ver Figura 44)

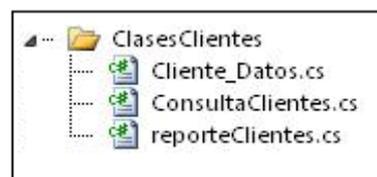
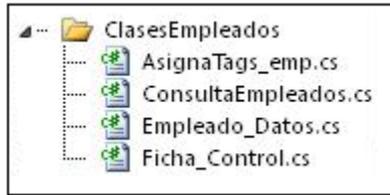


Figura 43: “Paquetes o Carpetas Clases Clientes de “RFIDBussinesSoft” ”

Fuente: Los Autores

#### 4.4.3. Paquete Clases Empleados

Este paquete efectúa Ingresos, actualización y consultas de empleados, como también la opción de asignar tags RFID para su identificación y control, de esta manera también podremos controlar el horario de entrada y de salida de estos. (Ver Figura 45).

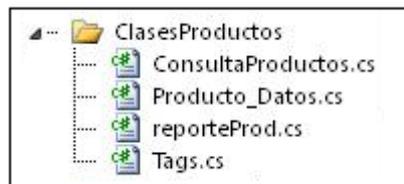


**Figura 44: “Paquetes o Carpetas Clases Empleados de “RFIDBussinesSoft” ”**

**Fuente:** Los Autores

#### **4.4.4. Paquete Clases Productos**

El actual paquete tendrá Ingresos, actualización y consultas de productos, como también la opción de asignar tags RFID para su identificación y de esta manera realizar ventas de forma muy ágil, además se podrá tener reportes de productos para controlar el stock. (Ver Figura 46).

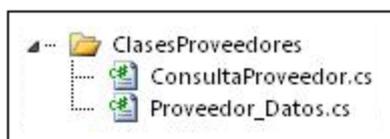


**Figura 45: “Paquetes o Carpetas Clases Productos de “RFIDBussinesSoft” ”**

**Fuente:** Los Autores

#### **4.4.5. Paquete Clases Proveedores**

El actual paquete tendrá Ingresos, actualización y consultas de proveedores. (Ver Figura 47).



**Figura 46: “Paquetes o Carpetas Clases Proveedores de “RFIDBussinesSoft” ”**

**Fuente:** Los Autores

#### 4.4.6. Paquete Clases RFID

Este paquete es uno de los más importante de “RFIDBussinesSoft“, ya que este contiene la clase de conversión de datos para manipular los datos que ofrece el tags RFID con relación a productos y empleados, además tenemos la forma de escribir y leer los Tags RFID, verificar el estado, sus respectivos parámetros y la conexión al lector RFID. (Ver Figura 48).

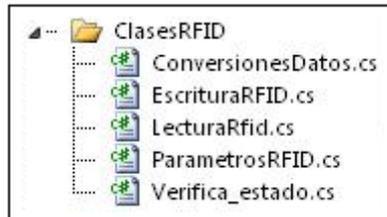


Figura 47: “Paquetes o Carpetas Clases RFID de “RFIDBussinesSoft” ”

Fuente: Los Autores

#### 4.4.7. Paquete Clases Ventas

Con la ayuda de este paquete se podrá facturar de forma ágil y rápida mediante radio frecuencia con la ayuda del lector y tags RFID, además tener una factura lista para imprimir con todos los detalles de la compra, como también la facilidad de cambiar el total a pagar de la factura de números a letras para una ayuda adicional al usuario. (Ver Figura 49).



Figura 48: “Paquetes o Carpetas Clases Ventas de “RFIDBussinesSoft” ”

Fuente: Los Autores

#### 4.4.8. Paquete Clases CMarcas Productos

Este paquete es para ingresar las marcas de los productos para un mejor control de los mismos. (Ver Figura 50).



**Figura 49: “Paquetes o Carpetas Clases CMarcas Productos de “RFIDBussinesSoft” ”**

**Fuente:** Los Autores

#### **4.4.9. Paquete Clases Grupos Productos**

Este paquete sirve para Ingresar grupos de productos, de esta manera cada producto tendrá un grupo específico para identificar cada uno en sus diferentes marcas y versiones. (Ver Figura 51).

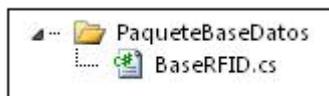


**Figura 50: “Paquetes o Carpetas Clases Grupos Productos de “RFIDBussinesSoft” ”**

**Fuente:** Los Autores

#### **4.4.10. Paquete Cases Base de Datos**

Este paquete ayuda y permite ejecutar sentencias SQL, además de realizar la conexión de base de datos con MySQL, en una sola clase. (Ver Figura 52).



**Figura 51: “Paquetes o Carpetas Clases Base de Datos de “RFIDBussinesSoft” ”**

**Fuente:** Los Autores

#### **4.4.11. Paquete Clases Roles**

Aquí se realizan los roles de trabajo dentro del sistema “RFIDBussinesSoft”, de esta forma se activarán cada panel según su rol. (Ver Figura 53).



Figura 52: “Paquetes o Carpetas Clases Roles de “RFIDBussinesSoft” ”

Fuente: Los Autores

#### 4.4.12. Paquete Clases Seguridad Ventas

Este paquete consta con una clase donde esta permitirá detectar robos o intentos de hurto de productos a la salida del local comercial. (Ver Figura 54).

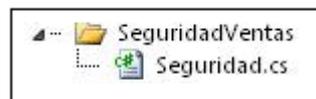


Figura 53: “Paquetes o Carpetas Clases Seguridad Ventas de “RFIDBussinesSoft” ”

Fuente: Los Autores

#### 4.4.13. Paquete Clases Validaciones

Dentro de este paquete esta la clase donde se valida todo el sistema “RFIDBussinesSoft”, en lo que se refiere a ingreso. (Ver Figura 55).

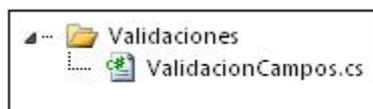


Figura 54: “Paquetes o Carpetas Clases Seguridad Ventas de “RFIDBussinesSoft” ”

Fuente: Los Autores

El sistema “RFIDBussinesSoft” trabaja con drivers tanto para el lector RFID y la base de datos MySql, para su óptimo funcionamiento. (Ver Figura 56).

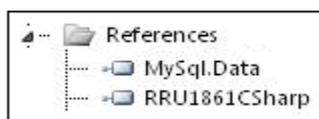


Figura 55: “Carpetas de Referencias para drivers de “RFIDBussinesSoft” ”

Fuente: Los Autores

#### 4.5. Creación de Módulos de Gestión

Los módulos de gestión del sistema RFIDBussinesSoft, constan de acciones similares y diferentes dependiendo de la gestión y su trabajo que realice dentro del sistema, que en el desarrollo se dividió en carpetas o paquetes:

- Gestión de Productos (Clases Productos)
- Gestión de Clientes (Clases Clientes)
- Gestión de Empleados (Clases Empleados)
- Gestión de Ventas (Clases Ventas)
- Gestión de Usuarios (Clases Usuarios)
- Gestión de Proveedores (Clases Proveedores)
- Gestión de Robos (Seguridad)

Existen otras carpetas o componentes que no son de gestión pero trabajan con sus clases para las diversas funcionalidades que el sistema ofrece y estas son:

- Paquete Base de Datos
- Clases RFID
- Validaciones
- Interfaces
- Roles

En la elaboración de los módulos de gestión, las funcionalidades dentro del sistema “RFIDBussinesSoft”, se utilizó para los ingresos, actualización y consultas de todos los módulos clases similares, que específicamente realizan la misma función.

Estas clases se utilizan en los diferentes paquetes de gestión para realizar las tareas de ingresos de información, son similares dependiendo de la información a ingresar puede ser: int, string, double, string sql para sentencia Sql, etc. Ya

que las mismas trabajan con métodos “Set” y “Get”, para la demostración se trabajará con Gestión de Productos (Ver Figura 57).

- Set: coloca datos en la variable.
- Get: recuperar información identificada.

```
public class Producto_Datos
{
    private string codpro,nom, des;
    private string marca, grupo;
    private double preci, prec2;

    public void set_codpro(TextBox t)
    {
        codpro = t.Text.ToUpper();
        codpro = codpro.Trim();
    }
    public string get_codpro()
    {
        return codpro;
    }
    //*****
    public void set_nompro(TextBox t)
    {
        nom = t.Text.ToUpper();
        nom = nom.Trim();
    }
    public string get_nompro()
    {
        return nom;
    }
    //*****
    public void set_descripcion(TextBox t)
    {
        des = t.Text.ToUpper();
        des = des.Trim();
    }
    public string get_descripcion()
    {
        return des;
    }
}
```

Figura 56: “Código C# de la clase “Producto\_Datos” y métodos de ingreso Set y Get de “RFIDBussinesSoft””

Fuente: Los Autores

También se podrá llamar a otras clases desde la clase “Producto\_Datos” como la Clase de conexión de la base de datos “Base RFID”, la clase donde se validarán los campos para el ingreso “Validación Campos”. (Ver Figura 58).

```
public class Producto_Datos
{
    private string sql = "";
    private BaseRFID Base;
    private ValidacionCampos vc;
    private MySqlDataReader dl;
```

Figura 57: “Código C# de invocación a otras clases desde la clase “Producto\_Datos” de “RFIDBussinesSoft””

Fuente: Los Autores

Después de la creación de los respectivos métodos set y get con el tipo dato determinado, se procede a realizar la instancia, todas las clases donde se realizará una o diversas operaciones que se necesiten, instanciamos en el FROM LOAD de la interfaz de Productos, dependiendo en que gestión sea, en este caso es la clase “Producto\_Datos”, que se encuentra en el paquete de Clases Productos. (Ver Figura 59)

```
////*****FORM LOAD*****
private void frmProductos_Load(object sender, EventArgs e)
{
    pd = new Producto_Datos();
    vc = new ValidacionCampos();
    md = new MarcasDatos();
    cp = new ConsultaProductos();
    gp1 = new Grilla_Personalizada();
    escribir_rfid = new EscrituraRFID();
    tag1 = new Tags();
    banac = true;
    cargar_comboG();
    cargar_comboM();
    banderas_desactivar();
    activa_componentes(false);
    asig_tag = false;
    car_for = new Carga_Formularios();
}
```

Figura 58: “Código C# Instancia de la clase de Ingreso “Producto\_Datos” de “RFIDBussinesSoft””

Fuente: Los Autores

A continuación, se volverá a la clase “Producto\_Datos” para crear el método Ingreso productos donde se procede a insertar sentencias Sql para ingresar la información a la base de Datos. (Ver Figura 60).

```
public class Producto_Datos
{
    private string sql = "";
    private BaseRFID Base;
}

public Boolean ingreso_productos()
{
    Boolean ban = false;

    //Sentencia Insert
    sql = "insert into productos(idProductos,Nombre_Producto,Descripcion_Producto"+
        ",Cantidad,Precio_Afiliado,Precio_Venta,Fecha_Ingreso,Imagen,Codigo_Fabricante,"+
        "Serie,Origen,Garantia,Grupo_idGrupo,IdEstado) values('" + get_codpro() + "','" + get_nompro() + "','" +
        "'" + get_descripcion() + "','" + get_cantidad() + "','" + get_precioA() + "','" + get_precioV() + "'" +
        "','" + get_fecha() + "','" + get_imagen() + "','" + get_codfab() + "','" + get_serie() + "'" +
        "','" + get_origen() + "','" + get_garantia() + "','" + get_idGrupos() + "','" + get_estado() + "')";

    Base = new BaseRFID();
    if (Base.comandos(sql))
    {
        ban = true;
    }
    else
    {
        ban = false;
    }
    return ban;
}
```

Figura 59: “Código C# Creación del método Ingreso\_Productos en la clase “Producto\_Datos” de “RFIDBussinesSoft””

Fuente: Los Autores

Ahora mediante la clase “Producto\_Datos” se creará el método de Actualizar productos donde se podrá escribir sentencias Sql para actualizar la información a la base de Datos. (Ver Figura 61).

```
public class Producto_Datos
{
    private string sql = "";
    private BaseRFID Base;
}

public Boolean Actualizar_productos()
{
    Boolean ban = false;

    //Sentencia Insert
    sql = " update productos(idProductos,Nombre Producto,Descripcion Producto"+
        ",Cantidad,Precio_Afiliado,Precio_Venta,Fecha_Ingreso,Imagen,Codigo_Fabricante,"+
        "Serie,Origen,Garantia,Grupo_idGrupo,IdEstado) values('" + get_codpro() + "','" + get_nompro() + "','" +
        "'" + get_descripcion() + "','" + get_cantidad() + "','" + get_precioA() + "','" + get_precioV() + "'" +
        "','" + get_fecha() + "','" + get_imagen() + "','" + get_codfab() + "','" + get_serie() + "'" +
        "','" + get_origen() + "','" + get_garantia() + "','" + get_idGrupos() + "','" + get_estado() + "')";

    Base = new BaseRFID();
    if (Base.comandos(sql))
    {
        ban = true;
    }
    else
    {
        ban = false;
    }
    return ban;
}
```

Figura 60: “Código C# Creación del método Actualizar\_Productos en la clase “Producto\_Datos” de “RFIDBussinesSoft””

Fuente: Los Autores

Para las consultas de la información almacenada utilizamos la clase “ConsultaProductos” esta clase consta con parámetros de búsqueda, este parámetro puede ser por (código, nombre, descripción, etc.) mediante un

COMBO BOX, y también por criterios de búsqueda, que es una búsqueda secuencial por letras. (Ver Figura 62).

```
class ConsultaProductos
{
    private string sql = "";
    private int index;
    private BaseRFID b;
    private string criterio;

    //*****
    public void set_opcion(ComboBox cmb)
    {
        index = cmb.SelectedIndex;
    }
    private int get_opcion()
    {
        return index;
    }
}
```

Figura 61: “Código C# Creación de la clase “ConsultaProductos” y el método de selección en un Combo Box de “RFIDBussinesSoft””

Fuente: Los Autores

En la creación de los siguientes métodos set y get, se podrá realizar la búsqueda por criterios, además de establecer la consulta con otro método llamado “establece\_consulta” para tener datos correctos. (Ver Figura 63).

```
public void set_criterio(TextBox t)
{
    criterio = t.Text.ToString();
    establece_consulta();
}
public void set_criterio_tag(TextBox t)
{
    criterio = t.Text.ToString();
    establece_consulta_tag();
}
private string getcriterio()
{
    return criterio;
}
```

Figura 62: “Código C# Creación del método Set y get Criterios de la clase “ConsultaProductos” de “RFIDBussinesSoft””

Fuente: Los Autores

A continuación, se crea el método establece consultas para verificar los criterios de búsqueda desde sentencias Sql a la base de datos para ser más sencilla la búsqueda. (Ver Figura 64).

```
private void establece_consulta_tag()
{
    switch (get_opcion())
    {
        //*****
        case 0:
            sql = "SELECT p.idProductos as Codigo_Producto, p.Nombre_Producto As Nombre_del_Producto, like ' " + getcriterio() + "%'";
            break;
        //*****
        case 1:
            sql = "SELECT p.idProductos as Codigo_Producto, p.Nombre_Producto As Nombre_del_Producto, like ' " + getcriterio() + "%'";
            break;
        //*****
        case 2:
            sql = "SELECT p.idProductos as Codigo_Producto, p.Nombre_Producto As Nombre_del_Producto, like ' " + getcriterio() + "%'";
            break;
    }
}
```

Figura 63: “Código C# Creación del método establece consulta de la clase “ConsultaProductos” de “RFIDBussinesSoft””

Fuente: Los Autores

Para la creación de los parámetros de búsqueda se realiza un método para establecer las consultas donde se trabajará con el método SWITCH CASE, por medio del cual se podrá escoger los parámetros con sentencias Sql (Ver Figura 65).

```
private void establece_consulta()
{
    switch (get_opcion())
    {
        //*****
        case 0:
            sql = "SELECT p.idProductos as Codigo_Producto like ' " + getcriterio() + "%'";
            break;
        //*****
        case 1:
            sql = "SELECT p.Nombre_Producto As Nombre_del_Producto like ' " + getcriterio() + "%'";
            break;
    }
}
```

Figura 64: “Código C# Creación del método Estable consulta de la clase “ConsultaProductos” de “RFIDBussinesSoft””

Fuente: Los Autores

Después de todo el proceso anterior, se volverá a instanciar la clase “ConsultaProductos” en el FROM LOAD de la interfaz de Productos, (Ver Figura 66)

```

//*****FORM LOAD*****
private void frmProductos_Load(object sender, EventArgs e)
{
    pd = new Producto_Datos();
    vc = new ValidacionCampos();
    md = new MarcasDatos();
    cp = new ConsultaProductos();
    gp1 = new Grilla_Personalizada();
    escribir_rfid = new EscrituraRFID();
    tag1 = new Tags();
    banac = true;
    cargar_comboG();
    cargar_comboM();
    banderas_desactivar();
    activa_componentes(false);
    asig_tag = false;
    car_for = new Carga_Formularios();
}

```

Figura 65: “Código C# Instancia de la clase de consulta “ConsultaProductos” de “RFIDBussinesSoft””

Fuente: Los Autores

Por último, se dará doble click en el BOTÓN “Almacenar Registros” para llamar al método Ingreso Productos, para verificar si los datos se ingresaron correctamente o no, por medios de mensajes de información. (Ver Figura 67).

```

private void btnagregarprod_Click(object sender, EventArgs e)
{
    cargar_info();
    if (pd.ingreso_productos())
    {
        MessageBox.Show("Datos Ingresados Correctamente", "RFID Business");
        GUARDA_MARCA_PROD();
        activa_componentes(false);
        banderas_desactivar();
        LIMPIAR();
        txtCodPro.Focus();
    }
    else
    {
        MessageBox.Show("Existe errores al Ingresar Informacion", "RFID Business");
    }
}

```

Figura 66: “Código C# Llamada de métodos de ingreso de Productos en sus respectivo botón de “RFIDBussinesSoft””

Fuente: Los Autores

También se dará nuevamente doble click en el BOTÓN “Actualizar Registro” para llamar al método Actualizar Productos, para verificar si los datos se actualizaron correctamente o no, por medio de mensajes de información. (Ver Figura 68).

```
private void btnactualizarprod_Click(object sender, EventArgs e)
{
    cargar_info();
    if (pd. Actualizar_productos())
    {
        MessageBox.Show("Datos Ingresados Correctamente", "RFID Business")
        GUARDA_MARCA_PROD();
        activa_componentes(false);
        banderas_desactivar();
        LIMPIAR();
        txtCodPro.Focus();
    }
    else
    {
        MessageBox.Show("Existe errores al Ingresar Informacion", "RFID Business")
    }
}
```

**Figura 67: “Código C# Llamada de métodos de actualización de Productos en su respectivo botón de “RFIDBussinesSoft””**

**Fuente:** Los Autores

Cabe recalcar que todo el proceso anterior tanto de ingresos, actualizaciones y consultas del paquete de gestión de productos conocido como “Clases Productos” que contiene las clases “Producto\_Datos” y “ConsultaProductos”, son similares a las clases de las otras gestiones del sistema “RFIDBussinesSoft” y estas clases son:

- Usuarios\_Datos.cs, ConsultaUsuarios.cs.
- Clientes\_Datos.cs, Consulta\_Clientes.cs.
- Empleados\_Datos.cs, Consulta\_Empleados.cs.
- Proveedor\_Datos.cs, ConsultaProveedor.cs.
- MarcasDatos.cs.
- GruposDatos.cs.

## 4.6. Creación del Paquete Base de Datos

En esta clase, creamos diversos métodos para poder tener una conexión estable a la base de datos llamada "rfdistema", además se trabajará en la conexión por medio del puerto 3306.

### 4.6.1. Llamada a librerías en la clase BaseRFID

Tenemos que llamar todas estas librerías para no tener errores al momento de la conexión, la librería más importante y para trabajar exactamente con MySQL es "Using MySql.Data.MySqlClient". (Ver Figura 69).

- Using MySql.Data.MySqlClient: esta librería hace la llamada al driver, "MySql.Data" que se encuentra en las preferencias del sistema "RFIDBussinesSoft"

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using MySql.Data;
using MySql.Data.Types;
using MySql.Data.MySqlClient;
using System.Data;
using System.Windows.Forms;

namespace RFIDBussinesSoft.PaqueteBaseDatos
{
    class BaseRFID
    {
        private MySqlConnection strCon;

        private String Servidor, nomBase, pw, Usuario, strCad;
        public BaseRFID()
        {
        }
    }
}
```

Figura 68: "Código C# Llamada a librerías en la Clase BaseRFID de "RFIDBussinesSoft"

Fuente: Los Autores

#### 4.6.2. Método Data set Consultar en la clase Base RFID

Por medio de la creación de este método podremos realizar consultas a la base de datos, ya que trabajamos con el constructor “MySqlConnection” y “SqlCommand” que (Ver Figura 70).

- MySqlConnection: inicializa una nueva instancia que establece los valores de lectura y también de escritura.
- SqlCommand: nos permite escribir sentencias Sql en el sistema.

```
public DataSet Consultar(String Tabla, String sql)
{
    DataSet ds = new DataSet();
    MySqlConnection oCon = new MySqlConnection(conectar().ConnectionString);
    // MySqlCommand insertar = new MySqlCommand(sql, oCon);
    MySqlDataAdapter Mda = new MySqlDataAdapter(sql, oCon);

    try
    {
        oCon.Open();
        Mda.Fill(ds, Tabla);
    }
    catch (Exception e)
    {
        Console.WriteLine("Error al crear la coneccion" + e.Message);
    }
    finally
    {
        oCon.Close();
    }

    return ds;
}
```

Figura 69: “Código C# Creación del método DataSet Consultar en la Clase BaseRFID de “RFIDBussinesSoft””

Fuente: Los Autores

#### 4.6.3. Método Comandos en la clase BaseRFID

Con la creación de este método permite el ingreso de datos a la base del sistema llamada “rfidsistema”, de esta forma al momento de insertar información a la base de datos tendremos un mensaje donde nos informa que alguna sentencia está mal ejecutada, trabaja también con “MySqlConnection”, “SqlCommand” además de “ConnectionOpen” y “ExecuteNonQuery” (Ver Figura 71).

- ConnectionOpen: ayuda a abrir la conexión a la base de datos.
- ExecuteNonQuery: por medio de este comando se podrá realizar operaciones de catálogo (por ejemplo, consultar la estructura de una base de datos o crear objetos de base de datos como tablas) o para cambiar la información de una base de datos ejecutando las instrucciones UPDATE, INSERT.

```

public Boolean comandos(String sql)
{
    Boolean res = true;
    MySqlConnection oCon = new MySqlConnection(conectar().ConnectionString);
    MySqlCommand insertar = new MySqlCommand(sql,oCon);
    try
    {
        insertar.Connection.Open();
        insertar.ExecuteNonQuery();
    }
    catch (Exception e)
    {
        res = false;
        MessageBox.Show("Error al Ejecutar Sentencia", "RFID Business")
        Console.WriteLine(e.Message.ToString());
    }
    finally
    {
        insertar.Connection.Close();
    }
    return res;
}

```

Figura 70: “Código C# Creación del método Consultar en la Clase BaseRFID de “RFIDBussinesSoft””

Fuente: Los Autores

#### 4.6.4. Método varios en la Clase BaseRFID

Para tener una conexión óptima del sistema a la base de datos, se tendrá que crear varios métodos los cuales cumplen una función específica en la conexión a la base de datos.

#### 4.6.4.1. Método que crea la cadena de conexión a la Base de Datos

En este método se crea la conexión a la base de datos, esta cadena identifica el servidor, la base de datos, el usuario, el password, y el puerto donde se realizara la conexión. (Ver Figura 72).

```
private string crea_cadena()
{
    strCad = "DataSource =" + getServidor() + " Database=" + getBase() + " Uid=" + getUsuario() +
    " Pwd=" + getPass() + " Port=3306";
    return strCad;
}
```

Figura 71: "Código C# Creación del método crea cadena en la Clase BaseRFID de "RFIDBussinesSoft"

Fuente: Los Autores

#### 4.6.4.2. Métodos de Extracción de la Clase BaseRFID

En estos métodos se colocarán los respectivos nombres que tendrá la conexión a la base de datos como: el nombre del servidor, del usuario, de la base de datos con la que se trabajará y su respectivo password. (Ver Figura 73).

<pre>private string getServidor() {     Servidor = "localhost;";     return Servidor; }</pre>	<pre>private string getUsuario() {     Usuario = "root;";     return Usuario; }</pre>
<pre>private string getPass() {     pw = "12345;";     return pw; }</pre>	<pre>private string getBase() {     nomBase = "rfidsistema;";     return nomBase; }</pre>

Figura 72: "Código C# Creación del método de extracción en la Clase BaseRFID de "RFIDBussinesSoft"

Fuente: Los Autores

## 4.7. Creación del Paquete Clases RFID

Este paquete de Clases es uno de los más importantes del sistema “RFIDBussinesSoft”, ya que por medio de esta se controlará la conexión con el lector RFID, también podemos escribir y leer tags RFID y por ultimo verificamos los parámetros del lector RFID, además se podrá abrir el puerto de comunicación según la configuración del equipo con su código de colores.

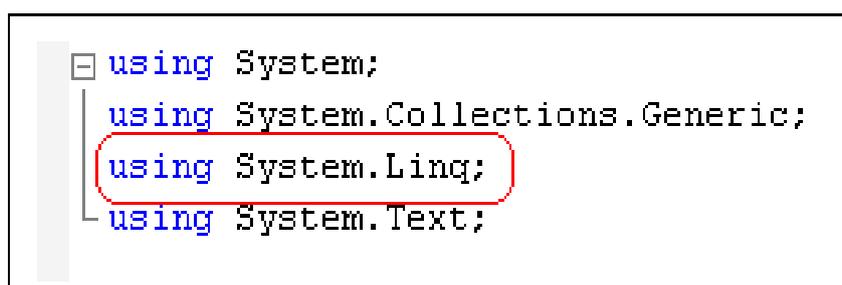
### 4.7.1. Clase Conversión de Datos

En esta clase se realizarán métodos de conversiones para interpretar la información que transmiten los Tags RFID, para poder leerla, manipularla y escribirla.

#### 4.7.1.1. Llamada a librerías en la Clase ConversionesDatos

En esta clase solo se trabaja con 3 librerías, una de ellas es la librería “System.Linq” (Ver Figura 74)

- System.Linq: realiza espacio de nombres cuando se especifica una instrucción using.



```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;
```

Figura 73: “Código C# Librerías Utilizadas en la clase ConversionesDatos de “RFIDBussinesSoft””

Fuente: Los Autores

#### 4.7.1.2. Creación de Métodos de Conversión en la Clase ConversionesDatos

La creación de métodos de conversión se realiza para manipular los datos, que llegan de los tags RFID, esta conversión es necesaria para asignar información del sistema a los Tags RFID, para esta conversión de cadena a hexadecimal y devolvemos un “array” de bytes se utilizo la función “Replace”, y la función “Substring”. (Ver Figura 75).

- Replace: reemplaza una subcadena en una cadena con otra en el sistema.
- Substring: recupera una subcadena de la instancia.
- Array: es una estructura de datos que contiene una serie de variables llamadas de los elementos parecida a una matriz.

```
public byte[] HexStringToByteArray(string s)
{
    s = s.Replace(" ", "");
    byte[] buffer = new byte[s.Length / 2];
    for (int i = 0; i < s.Length; i += 2)
        buffer[i / 2] = (byte)Convert.ToByte(s.Substring(i, 2), 16);
    return buffer;
}
```

Figura 74: “Código C# Creación del método de conversión cadena a hexadecimal en la Clase ConversionesDatos de “RFIDBussinesSoft””

Fuente: Los Autores

A continuación, se creará otro método para otra conversión, esta vez la conversión es de bytes a hexadecimal y devolvemos una cadena, con la función “Append”. Con esta conversión se podrá ya manipular la información para el uso de asignaciones en el sistema y almacenarla en la base de datos. (Ver Figura 76).

- Append: añade una copia de una subcadena especificada al final de una instancia.

```
public string ByteArrayToHexString(byte[] data)
{
    StringBuilder sb = new StringBuilder(data.Length * 3);
    foreach (byte b in data)
        sb.Append(Convert.ToString(b, 16).PadLeft(2, '0'));
    return sb.ToString().ToUpper();
}
```

Figura 75: “Código C# Creación del método de conversión de bytes a hexadecimal en la Clase ConversionesDatos de “RFIDBussinesSoft””

Fuente: Los Autores

### 4.7.2. Clase Escritura RFID

Esta clase sirve únicamente para escribir datos en los tags RFID y de esta forma se asigna un Tag de identificación cada producto o empleado, almacenado en la base de datos.

#### 4.7.2.1. Llamadas a Librerías en la Clase EscrituraRFID

Las siguientes librerías fueron llamadas para poder trabajar con esa clase, en esta clase llamamos a la librería de la base de datos “MySql.Data.MySqlClient”, porque se trabajará directamente con la base de datos “rfdistema”. (Ver Figura 77).

- MySql.Data.MySqlClient: contiene todas las funciones de MySql.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using ReaderB;
using RFIDBussinesSoft.PaqueteBaseDatos;
using MySql.Data.MySqlClient;
```

Figura 76: “Código C# Librerías utilizadas en la clase EscrituraRFID de “RFIDBussinesSoft””

Fuente: Los Autores

#### 4.7.2.2. Método Escribir Datos en la Clase EscrituraRFID

Por medio de la creación de este método se escribirán datos en los tags RFID, para ello se utilizarán diversas funciones como: “Convert”, el método Escribir Datos muestra un mensaje de información de datos grabados correctamente. (Ver Figura 78).

- Convert: convierte un tipo de datos base en otro tipo de datos base, como por ejemplo el “Convert.ToByte” es un método que convierte un valor específico en un entero de 8 bytes sin signo.

```
public void escribir_datos_rfid(TextBox cod,TextBox pass)
{
    carga_parametros();
    cd = new ConversionesDatos();
    ferrorcode = -1;
    byte[] WriteEPC = new byte[100];
    byte WriteEPClen;
    byte ENum;
    if (pass.Text.Length < 8) {
        return;
    }
    if ((cod.Text.Length % 4) != 0) {
        return;
    }
    WriteEPClen = Convert.ToByte(cod.Text.Length / 2);
    ENum = Convert.ToByte(cod.Text.Length / 4);
    byte[] EPC = new byte[ENum];
    EPC = cd.HexStringToByteArray(cod.Text);
    fPassWord = cd.HexStringToByteArray(pass.Text);
    fCmdRet = StaticClassReaderB.WriteEPC_G2(ref fComAdr, fPassWord, EPC,
    if (fCmdRet == 0)
        MessageBox.Show("Datos Grabados Correctamente", "Informacion"); WriteEPClen. ref ferrorcode. frmcomportindex);
}
```

Figura 77: “Código C# creación del método escribir datos en la clase EscrituraRFID de “RFIDBussinesSoft “”

Fuente: Los Autores

#### 4.7.3. Clase Lectura RFID

Esta clase se utilizará para leer los datos que transmiten los tags RFID, de esta manera se trabajará en las diferentes gestiones del sistema ya que permite leer directamente desde la base de datos.

#### 4.7.3.1. Llamadas a Librería en la Clase LecturaRFID

La clase “LecturaRFID” llama a la librería de la base de datos “MySQL.Data.MySqlClient”, porque se trabajará directamente con la base de datos “rfidsistema”. (Ver Figura 79).

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using ReaderB;
using MySql.Data.MySqlClient;
using RFIDBussinesSoft.PaqueteBaseDatos;
```

Figura 78: “Código C# Librerías utilizadas en la clase LecturaRFID de “RFIDBussinesSoft””

Fuente: Los Autores

#### 4.7.3.2. Método Lectura de Datos en la Clase LecturaRFID

En este método se realizará la lectura de los tags RFID, después de haberlos escrito, en este método se trabajará con “ToHexString”, también llamamos a los métodos de las clases “ConversionesDatos”, para crear las conversiones, y de esta manera leer los tags RFID. (Ver Figura 80)

- ToHexString: convierte un valor hexadecimal a cadena.

```
public string Lectura_de_Datos()
{
    carga_parametros();
    int CardNum = 0;
    int TotalLen = 0;
    int EPCLen, m;
    byte[] EPC = new byte[5000];
    int CardIndex;
    string temps = "";
    string sEPC = "";

    fCmdRet = StaticClassReaderB.Inventory_G2(ref fComAdr, EPC, ref TotalLen, ref CardNum, frmcomportindex);
    if ((fCmdRet == 1) | (fCmdRet == 2) | (fCmdRet == 3) | (fCmdRet == 4) | (fCmdRet == 0xFB))
    {
        cd = new ConversionesDatos();
        byte[] daw = new byte[TotalLen];
        Array.Copy(EPC, daw, TotalLen);
        temps = cd.ByteArrayToHexString(daw);
        fInventory_EPC_List = temps;

        m = 0;
        for (CardIndex = 0; CardIndex < CardNum; CardIndex++)
        {
            EPCLen = daw[m];
            //recomendacion
            sEPC = temps.Substring(m * 2 + 2, EPCLen * 2);
            m = m + EPCLen + 1;
            //if (sEPC.Length != EPCLen * 2)
        }
    }
}
```

Figura 79: “Código C# creación del método lecturas de datos en la clase LecturaRFID de “RFIDBussinesSoft””

Fuente: Los Autores

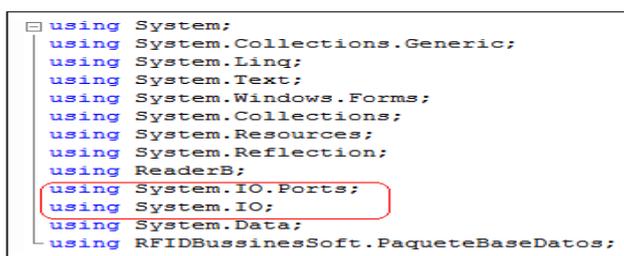
#### 4.7.4. Clase Parámetros RFID

En esta clase se verá y establecerá los parámetros para trabajar con el lector RFID, donde verificaremos el estado de memoria, memoria bloqueada, entre otros.

##### 4.7.4.1. Llamada a Librerías en la Clase ParametrosRFID

La clase “ParametrosRFID” llama dos importantes librerías “System.IO.Ports” y “System.IO” que permitirán abrir y cerrar los puertos, seleccionar un puerto “COM” del lector RFID. (Ver Figura 81).

- System.IO.Ports: contiene clases para el control de los puertos seriales.
- System.IO: contiene tipos que proporcionan archivos básicos y el apoyo del directorio que permite la lectura y escritura de los archivos y flujos de datos.



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Collections;
using System.Resources;
using System.Reflection;
using ReaderB;
using System.IO.Ports;
using System.IO;
using System.Data;
using RFIDBussinesSoft.PaqueteBaseDatos;
```

Figura 80: “Código C# llamada a librerías en la clase LecturaRFID de “RFIDBussinesSoft””

Fuente: Los Autores

##### 4.7.4.2. Método Abrir Puerto en la Clase ParametrosRFID

Mediante este método, se podrá abrir los puertos seriales para conectarse al lector RFID ya sea de forma automática o escogiendo un COM, para esto se utilizará “Serial Port” y “AutoOpenComPort” de esta forma la comunicación PC-Lector RFID se completa. (Ver Figura 82).

```

public void AbrirPuerto(TextBox Edit_CmdComAddr, ComboBox ComboBox_COM, ComboBox ComboBox_baud,
{
    inicial_var();
    int port = 0;
    int openresult, i;
    openresult = 30;
    string temp="";

    if (Edit_CmdComAddr.Text == "")
        Edit_CmdComAddr.Text = "FF";
    fComAdr = Convert.ToByte(Edit_CmdComAddr.Text, 16); // $FF;
}

```

**Figura 81: “Código C# creación del método abrir puerto en la clase ParametrosRFID de “RFIDBussinesSoft””**

**Fuente:** Los Autores

En este mismo método se creará la conexión de forma automática y manual, desde un COMBO BOX para la comunicación. (Ver Figura 83).

```

if (ComboBox_COM.SelectedIndex == 0)//si se elige Auto
{
    openresult = StaticClassReaderB.AutoOpenComPort(ref port, ref fComAdr, fBaud, ref frmcomportindex);
    fOpenComIndex = frmcomportindex;
    if (openresult == 0)
    {
        ComOpen = true;

        if (fBaud > 3)
        {
            ComboBox_baud.SelectedIndex = Convert.ToInt32(fBaud - 2);
        }
        else
        {
            ComboBox_baud.SelectedIndex = Convert.ToInt32(fBaud);
        }
    }
    Establecer_datos(Edit_Version, Edit_NewComAdr, Edit_CmdAdr, Edit_scantime, Edit_Type, ISO180006B,
    if ((fCmdRet == 0x35) | (fCmdRet == 0x30))
    {
        MessageBox.Show("Error al Comunicarse Coneccion ocupada", "Informacion");
        StaticClassReaderB.CloseSpecComPort(frmcomportindex);
        ComOpen = false;
    }
}

```

**Figura 82: “Código C# creación del método abrir puerto de forma automática en la clase ParametrosRFID de “RFIDBussinesSoft””**

**Fuente:** Los Autores

#### 4.7.4.3. Método de Cerrar Puerto en la Clase ParametrosRFID

Con este método, se cerrará el puerto de comunicación para no desperdiciar recursos dentro del sistema. (Ver Figura 84).

```

public void cerrar_puerto(ComboBox ComboBox_AlreadyOpenCOM)
{
int port;
//string SelectCom ;
b = new BaseRFID();
string temp,sql;
sql = "Update estadorfid set idEstado=2 where Id=1";
b.comandos(sql);
try
{
if (ComboBox_AlreadyOpenCOM.SelectedIndex < 0 )
{
MessageBox.Show("Escoja el Puerto a cerrar", "Informacion");
}
else
{
temp = ComboBox_AlreadyOpenCOM.SelectedItem.ToString();
port = Convert.ToInt32(temp.Substring(3, temp.Length - 3));
fCmdRet = StaticClassReaderB.CloseSpecComPort(port);
if (fCmdRet == 0)
{
ComboBox_AlreadyOpenCOM.Items.RemoveAt(0);
if (ComboBox_AlreadyOpenCOM.Items.Count != 0)
{
temp = ComboBox_AlreadyOpenCOM.SelectedItem.ToString();
port = Convert.ToInt32(temp.Substring(3, temp.Length - 3));
StaticClassReaderB.CloseSpecComPort(port);
fComAdr = 0xFF;
StaticClassReaderB.OpenComPort(port,ref fComAdr, fBaud,ref frmcompportindex);
fOpenComIndex = frmcompportindex;
}
}
}
}
}

```

Figura 83: “Código C# creación del método cerrar puerto en la clase ParametrosRFID de “RFIDBussinesSoft””

Fuente: Los Autores

Todas estas clases y métodos del Paquete Clases RFID, se instanciarán en el FROM LOAD en el formulario de RFIDConfig, para llamar a este formulario en todo el sistema, se realizará en el BOTON “Activar RFID”. (Ver Figura 85).

```

private void btnactivarrfid_Click(object sender, EventArgs e)
{
frmRFIDConfig frid;
frid = new frmRFIDConfig();
frid.Show();
}

```

Figura 84: “Código C# Llamada de configuraciones RFID en el botón Activar RFID de “RFIDBussinesSoft””

Fuente: Los Autores

#### 4.7.5. Clase Verifica Estado

En esta clase se verificará el estado del lector RFID desde la base de datos para revisar su estado dentro del sistema. (Ver Figura 86).

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using RFIDBussinesSoft.PaqueteBaseDatos;
using MySql.Data.MySqlClient;
namespace RFIDBussinesSoft.ClasesRFID
{
    public class Verifica_estado
    {
        private BaseRFID b;
        private int estado;
        private string selec;
        public int get_estado()
        {
            return estado;
        }
        public string get_seleccion()
        {
            return selec;
        }
        public void carga_parametros_f()
        {
            MySqlDataReader msd;
            b = new BaseRFID();
            msd = b.consulta_reader("Select * from estadorfid where id=1");
            while (msd.Read())
            {
                estado = msd.GetInt32(1);
                selec = msd.GetString(3);
            }
        }
    }
}

```

Figura 85: “Código C# creación de la clase verificar estado y el método cargar parámetros de “RFIDBussinesSoft””

Fuente: Los Autores

## 4.8. Creación de Gestión de Empleados

En esta gestión, se tendrá aparte de las clases de ingreso y consultas, dos clases importantes, que son las clases asignación de tag y Ficha control.

### 4.8.1. Creación de la Clase Asignación de Tags

Con la creación de esta clase y sus respectivos métodos, se asignará una identificación adicional a los empleados, esta identificación es la del tag RFID, de esta manera ya podremos controlar horarios de trabajo.

#### 4.8.1.1. Llamada a librerías en la Clase Asigna Tags Empleados

Esta clase llama a la librería referente a las clases de la base de datos y a las clases RFID. (Ver Figura 87).

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using RFIDBussinesSoft.PaqueteBaseDatos;
using RFIDBussinesSoft.ClasesRFID;

```

Figura 86: “Código C# llamada de librerías en la clase Asigna Tags empleados de “RFIDBussinesSoft””

Fuente: Los Autores

#### 4.8.1.2. Método de Asignación de Tags en la Clase Asignación Tags Empleados

Por medio de este método se asignará un tag de identificación a cada empleado creado desde un GRID VIEW y todos estos cambios se guardarán en la base de datos del sistema. (Ver Figura 88).

- DataGridView: permite tomar los valores de la grilla de la posición y columna deseada

```

public void escoger_grilla(DataGridView tbl, DataGridViewCellEventArgs ex)
{
    codemp = tbl.Rows[ex.RowIndex].Cells[0].Value.ToString();

    public void Tags_asignacion(TextBox tcod, TextBox tpass)
    {
        escri_rfid = new EscrituraRFID();
        escri_rfid.escribir_datos_rfid(tcod, tpass);
    }
}

```

Figura 87: “Código C# Creación de método Tags asignación en la clase Asigna Tags empleados de “RFIDBussinesSoft””

Fuente: Los Autores

De la misma forma se instanciará esta clase en el FROM LOAD del formulario de empleados, y para la asignación de tags a los empleados se realizará en el BOTÓN “Asignar Tags”. (Ver Figura 89).

```

private void btnasignatags_Click(object sender, EventArgs e)
{
    if (txttag.Text != "")
    {
        tag1.Tags_asignacion(txttag, txtpass);

        txttag.Text = "";
    }
    else
    {
        MessageBox.Show("DEBE SELECCIONAR UN ELEMNTO DE LA TABLA DE TAGS", "ERROR RFDIBUSINESS");
    }
}

```

**Figura 88: “Código C# Llamada de asignación de tags RFID en el botón Asignar Tags de “RFIDBussinesSoft””**

**Fuente:** Los Autores

## 4.8.2. Creación de la Clase Ficha Control

En esta clase podremos verificar la entrada y salida de los empleados (horas exactas) por medio del reloj del sistema, y se podrá visualizar esta información en un DataGridView.

### 4.8.2.1. Llamadas a Librerías en la clase Ficha Control

En esta clase se tendrá las librerías de base de datos y de RFID (Ver Figura 90)

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using RFIDBussinesSoft.Interfaces;
using RFIDBussinesSoft.ClasesRFID;
using RFIDBussinesSoft.PaqueteBaseDatos;

```

**Figura 89: “Código C# llamada de librerías en la clase Ficha Control de “RFIDBussinesSoft””**

**Fuente:** Los Autores

#### 4.8.2.2. Método mostrar ingresos en la Clase Ficha Control

Este método muestra el ingreso de los empleados con sentencias Sql, por medio de “DataSource” y “DataMember”. (Ver Figura 91).

- DataSource: obtiene o establece el origen de datos cuyos datos se están mostrando en el control DataGridView.
- DataMember: obtiene o establece la lista específica en una propiedad DataSource para la cual el control DataGridView muestra una cuadrícula.

```
private void mostrar_ingresos(DataGridView tabla1)
{
    string sql="";
    sql = "select Empleados_idEmpleados, Hora_Entrada, Hora_Salida, Anio, Observaciones,
Fecha from ficha_trabajo where fecha='" + DateTime.Now.ToShortDateString() + "'";
    ba = new BaseRFID();
    tabla1.DataSource = ba.Consultar("ficha_trabajo", sql);
    tabla1.DataMember = "ficha_trabajo";
}
```

Figura 90: “Código C# Creación de mostrar ingresos en la clase Ficha control de “RFIDBussinesSoft””

Fuente: Los Autores

#### 4.8.2.3. Método Fecha del Sistema en la Clase Ficha Control

En este método tomamos los datos de fecha del sistema con “DateTime”, para poder comprobar los horarios de trabajo de los empleados. (Ver Figura 92).

- DateTime: Representa un instante de tiempo, normalmente expresado en forma de fecha y hora del día.

```

private void set_fecha_hora()
{
    fec = DateTime.Now.ToShortDateString();
    hora_entra_sale = DateTime.Now.ToShortTimeString();
    dia = Convert.ToInt32(DateTime.Now.DayOfWeek);
    mes = DateTime.Now.Month;
    anio = DateTime.Now.Year;
    hor = DateTime.Now.Hour;
    min = DateTime.Now.Minute;
}

```

Figura 91: “Código C# Creación del método Fecha del sistema en la clase Ficha control de “RFIDBussinesSoft””

Fuente: Los Autores

## 4.9. Creación de Gestión de Ventas

En la creación de gestión de ventas se tendrá tres clases, las mismas servirán para facturar, ver reportes de ventas y una clase especial para cambiar números a letras al momento de facturar.

### 4.9.1. Clases Facturar

Esta es una de las clases más importantes de todo el sistema ya que por medio de ella, se realizará la facturación rápida ágil de todos los productos comprados a la vez, por medio de un DataGridView, también podremos ingresar datos del cliente, del vendedor, fecha y numero de factura.

#### 4.9.1.1. Llamada a librerías en la Clase Facturar

La clase facturar trabaja con varias librerías, la librería de base de datos es la más importante ya que por medio de esta, se consultará toda la información de la factura. (Ver Figura 93).

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using RFIDBussinesSoft.PaqueteBaseDatos;
using RFIDBussinesSoft;
using MySql.Data.MySqlClient;
using System.Windows.Forms;
using System.IO;

```

Figura 92: “Código C# llamada de librerías en la clase Facturar de “RFIDBussinesSoft””

Fuente: Los Autores

#### 4.9.1.2. Método Cargar Factura en la Clase Facturar

Este método permite cargar la factura, en las grillas del DataGridView para verificar que productos son los que se van a vender. (Ver Figura 94).

```

public void carga_factura(DataGridView tabla1, string dato)
{
    int tam;
    Boolean b = true;
    tam = tabla1.RowCount;
    try
    {
        if (!dato.Equals(""))
        {
            for (int i = 0; i < tam-1; i++)
            {
                if (dato.Equals(tabla1.Rows[i].Cells[1].Value.ToString()))
                {
                    b = false;
                    break;
                }
            }
            if (b)
            {
                if (comprueba_dato(dato))
                {
                    inserta_grilla(tabla1, dato, tam-1);
                }
            }
        }
    } catch (Exception)

```

Figura 93: “Código C# Creación del método cargar factura de la clase facturar de “RFIDBussinesSoft””

Fuente: Los Autores

#### 4.9.1.3. Método Cargar Factura Base en la Clase Facturar

Este método carga toda la información de la factura a la base de datos desde un DataGridView, para ello se trabajará con sentencias Sql, y con mensaje de

información si se encuentran problemas al momento de verificar los detalles de ventas. (Ver Figura 95).

```
public void carga_factura_Base(DataGridView tabla1)
{
    string sql = "";
    int tam;
    b = new BaseRFID();
    tam = tabla1.RowCount;

    try
    {
        sql = "Insert into ventas(IdVentas, Clientes_idClientes, Empleados_idEmpleados, Numero_Factura," +
            " Fecha, SubTotal, Descuentos, Iva, Total,ANIO,IDMES, IDDLA, IdEstado_Fac)" +
            " values (" + get_Num_fac() + "," + get_cod_clie() + "," + get_cod_vendedor() + "," +
            " " + get_Num_fac() + "," + get_fec() + "," + get_subTotal() + "," + get_descuento() + "," +
            " " + get_iva() + "," + get_total() + "," + get_anio() + "," + get_mes() + "," +
            " " + get_dia() + "," + get_estado_Fac() + ")";

        b.comandos(sql);

        for (int i = 0; i < tam - 1; i++)
        {
            sql = "Update productos set Cantidad=Cantidad-1 where idProductos='" + tabla1.Rows[i].Cells[0].Value.ToString() +
            "'; if (b.comandos(sql))
            {
            }
            else
            {
                MessageBox.Show("No se ejecuto update productos");
            }
            sql = "Update seguridad_prod set IdEstado=2 where Tag='" + tabla1.Rows[i].Cells[1].Value.ToString() + "'";
            if (b.comandos(sql))
            {
            }
        }
    }
}
```

**Figura 94: “Código C# Creación del método cargar factura base de la clase facturar de “RFIDBussinesSoft””**

**Fuente:** Los Autores

Como anteriormente se explicó se debe instanciar en el FROM LOAD del formulario “Ventas”, y para poder realizar la factura damos un clic en el BOTÓN “Facturar”. (Ver Figura 96).

```

private void button1_Click(object sender, EventArgs e)
{
    if (!vc.validar_vacios(txtcredencial, ptrv1) && !vc.validar_vacios(txtdir, ptrv3)
    {
        factura.set_fecha(dtmFecha);
        factura.set_SubTotal(dtgFactura);
        factura.set_descuento(txtdes);
        factura.mostrar_valores_fac(txtsub, txtdes, txtiva, txttot, lbltot);
        factura.carga_factura_Base(dtgFactura);
        rv.impresion_factura(int.Parse(txtnumfact.Text.ToString()), fc);
        limpiar_todo();
    }
    else
    {
        MessageBox.Show("No Se puede Generar una Factura Sin Datos", "RFID Business",
    }
}

```

**Figura 95: “Código C# Llamada a métodos de facturación desde el botón Facturar de “RFIDBussinesSoft””**

Fuente: Los Autores

#### 4.9.2. Clase Reporte Ventas

En esta clase, se visualizará la factura para después imprimirla por cada venta realizada, estas facturas son creadas en “CristalReports”.

- CristalReports: es una aplicación de inteligencia empresarial utilizada para diseñar y generar informes desde una amplia gama de fuentes de datos.

##### 4.9.2.1. Llamada a librerías de la Clase Reportes Ventas

Las librerías que se llaman en esta clase son las más comunes pero las más importantes son la de la base de datos y la librería de CristalReports para la plantilla de facturación. (Ver Figura 97).

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using RFIDBussinesSoft.Reportes_cristal;
using RFIDBussinesSoft.ReportesDataSets;
using RFIDBussinesSoft.PaqueteBaseDatos;
using MySql.Data.MySqlClient;
using RFIDBussinesSoft.Reportes_cristal;
using RFIDBussinesSoft.Inerfaces_Reportes;

```

Figura 96: “Código C# llamada de librerías en la clase Reportes Ventas de “RFIDBussinesSoft””

Fuente: Los Autores

#### 4.9.2.2. Métodos Impresión Factura en la Clase Reporte Ventas

En este método se extraerá todo lo que tiene la factura para imprimir y enviarla al CristalReports por medio de sentencias Sql. (Ver Figura 98).

```

public void impresion_factura(int num_fact, frm_fact_ventas frm)
{
    rpt_p = new rptFac_Datos();

    string sql = "", sql1 = "";
    sql = "Select v.IdVentas as Num Factura, c.ApellidosNombres as Cliente," +
        " c.Credencial as RUC_CC, c.Direccion as Direccion, c.Telefono as Telefono," +
        " e.ApellidosNombres as Vendedor, v.SubTotal as Subtotal, v.Descuentos as Descuento," +
        " v.Iva as Iva, v.Total as Total from ventas v, empleados e, clientes c where " +
        " c.idClientes=v.Clientes_idClientes and v.Empleados_idEmpleados=e.idEmpleados and v.IdVentas=" + num_fact

    sql1 = "select vd.Cantidad as Cantidad, p.Descripcion_Producto as Detalle," +
        " m.Nombre as Marca, vd.Valor_Compra as Precio from ventas v," +
        " detalle_venta vd, marca m, marca_producto mp, productos p where " +
        " p.idProductos=vd.Productos_idProductos and p.idProductos=mp.Productos_idProductos" +
        " and m.idMarca=mp.Marca_idMarca and vd.Ventas_idVentas=v.idVentas and v.IdVentas=" + num_fact + ";
    rpt_p.Subreports["rpt_detalle1"].SetDataSource(crea_reporte_ventas("detalle_venta", sql1).Tables[1]);
    rpt_p.SetDataSource(crea_reporte_detalle("ventas", sql).Tables[1]);
    frm = new frm_fact_ventas();

    frm.crystalReportViewer1.ReportSource = rpt_p;
    frm.Show();
    frm.crystalReportViewer1.Show();
}

```

Figura 97: “Código C# método de imprimir facturas en la clase Reportes Ventas de “RFIDBussinesSoft””

Fuente: Los Autores

### 4.9.3. Clase Numero Letras

Esta clase convierte el total a pagar de la factura de números a letras, con esto se ayudará al vendedor a tener una visión más cómoda del total a pagar.

#### 4.9.3.1. Llamada a librerías de la Clase Numero Letras

A continuación, se presentan las librerías que necesita la clase Número letras para su funcionamiento. (Ver Figura 99).

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Text.RegularExpressions;
```

Figura 98: “Código C# Llamada librerías de la clase Números Letras de “RFIDBussinesSoft””

Fuente: Los Autores

#### 4.9.3.2. Método convertir de la Clase Números Letras

En este método se convertirán los números a letras, según el formato de entrada que da el total a pagar de la factura. (Ver Figura 100).

```
public String Convertir(String numero, bool mayusculas)
{
    String literal = "";
    String parte_decimal;
    //si el número utiliza (.) en lugar de (,) -> se reemplaza
    numero = numero.Replace(".", ",");

    //si el número no tiene parte decimal, se le agrega ,00
    if (numero.IndexOf(",") == -1)
    {
        numero = numero + ",00";
    }
    //se valida formato de entrada -> 0,00 y 999 999 999,00
    r = new Regex(@"^\d{1,9};\d{1,2}");
    MatchCollection mc = r.Matches(numero);
    if (mc.Count > 0)
    {
        //se divide el número 0000000,00 -> entero y decimal
        String[] Num = numero.Split(',');

        //de da formato al número decimal
        parte_decimal = Num[1] + "/100 Dolares";
        //se convierte el número a literal
        if (int.Parse(Num[0]) == 0)
        {
            //si el valor es cero
            literal = "cero ";
        }
        else if (int.Parse(Num[0]) > 999999)
        {
            //si es millon
            literal = getMillones(Num[0]);
        }
    }
}
```

Figura 99: “Código C# método convertir de la clase Números Letras de “RFIDBussinesSoft””

Fuente: Los Autores

## 4.10. Creación de Gestión de Usuarios

En esta gestión, se otorgará al usuario permisos para ingresar al sistema, donde cada usuario tendrá acceso a las diversas gestiones según sea su rol en el sistema.

### 4.10.1. Clase Asigna Permisos

En esta clase, se asignará permisos de uso del sistema a los usuarios, según su permiso se le activaran las gestiones.

#### 4.10.1.1. Llamadas a librerías de la Clase Asigna Permisos

Esta clase trabaja con librerías importante como la librería “System.Resources”. (Ver Figura 101).

- System.Resources: utiliza los elementos de la pantalla.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Resources;
using RFIDBussinesSoft.PaqueteBaseDatos;
using RFIDBussinesSoft;
using MySql.Data.MySqlClient;
```

Figura 100: “Código C# Llamadas a librerías De la clase Asigna Permisos de “RFIDBussinesSoft””

Fuente: Los Autores

#### 4.10.1.2. Método permisos en la Clase Asigna Permisos

Con la creación de este método se asignará permisos a cada usuario y se podrá activar el menú del sistema según su trabajo, se utilizará código Sql para la activación de las de gestiones del sistema. (Ver Figura 102).

```

public void permisos()
{
    string sql = "";
    sql = "Update menu_usuario set IdEstado=" + get_gestion_productos ();
    b.comandos(sql);
    sql = "Update menu_usuario set IdEstado=" + get_gestion_ven ();
    b.comandos(sql);
    sql = "Update menu_usuario set IdEstado=" + get_gestion_clie ();
    b.comandos(sql);
    sql = "Update menu_usuario set IdEstado=" + get_gestion_emp();
    b.comandos(sql);
    sql = "Update menu_usuario set IdEstado=" + get_gestion_seg();
    b.comandos(sql);
    sql = "Update menu_usuario set IdEstado=" + get_gestion_rfid();
    b.comandos(sql);
    sql = "Update menu_usuario set IdEstado=" + get_gestion_prov ();
    b.comandos(sql);
    sql = "Update menu_usuario set IdEstado=" + get_gestion_us();
    b.comandos(sql);
}

```

**Figura 101: “Código C# método permisos de la clase Asigna Permisos de “RFIDBussinesSoft””**

**Fuente:** Los Autores

#### 4.10.2. Clase Asigna Menú

Esta clase asignan las gestiones según el permiso otorgado al usuario.

##### 4.10.2.1. Llamadas a librerías de la Clase Asigna Menú

A continuación, se verificarán las librerías de la clase Asigna Menú (Ver Figura 103).

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using RFIDBussinesSoft.PaqueteBaseDatos;
using MySql.Data.MySqlClient;
using System.Windows.Forms;

```

**Figura 102: “Código C# Llamada a librerías de la clase Asigna Menú de “RFIDBussinesSoft””**

**Fuente:** Los Autores

#### 4.10.2.2. Método Control Menú de la Clase Asigna Menú

Creamos métodos set y Get por cada gestión que tiene el sistema, para asignar los permisos, de esta forma se procederá a realizar el método Control Menú, donde se activa cada gestión que utilizará el usuario. (Ver Figura 104).

```
public void control_menu(Button b1, Button b2, Button b3, Button b4, Button b5, Button b6, Button b7, Button b8)
{
    ges_clientes();
    ges_personal();
    ges_productos();
    ges_proveedores();
    ges_RFID();
    ges_seguridad();
    ges_usuarios();
    ges_ventas();

    b1.Enabled = b_prod;
    b2.Enabled = b_vent;
    b3.Enabled = b_clie;
    b4.Enabled = b_per;
    b5.Enabled = b_seg;
    b6.Enabled = b_rfid;
    b7.Enabled = b_prov;
    b8.Enabled = b_usu;
}
```

Figura 103: “Código C# Método control menú de la clase Asigna Menú de “RFIDBussinesSoft””

Fuente: Los Autores

#### 4.10.3. Clase Login Usuarios

En esta clase se darán los parámetros para el ingreso al sistema, según los permisos otorgados a los usuarios.

##### 4.10.3.1. Llamadas a librerías de la Clase Login Usuario

Verificamos las librerías de la clase Login Usuario (Ver Figura 105).

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using RFIDBussinesSoft.ClasesProductos;
using RFIDBussinesSoft.PaqueteBaseDatos;
using RFIDBussinesSoft.Grupos_Productos;
using RFIDBussinesSoft.Interfaces;
using System.Windows.Forms;

```

**Figura 104: “Código C# Llamada a librerías de la clase Login Usuario de “RFIDBussinesSoft””**

**Fuente:** Los Autores

#### 4.10.3.2. Método de validación de Usuario de la Clase Login Usuario

En este método por medio de sentencias Sql, se verificará el usuario y contraseña para el ingreso a sistema y sus respectivas activaciones sea cual sea su rol. (Ver Figura 106).

```

public Boolean valida_usuario(PictureBox ptr)
{
    try
    {
        int num_filas = 0;

        base2 = new BaseRFID();
        sql = "select * from usuarios where " +
            " Nombre_Usuario='" + get_usuario() + "' and Contraseña='" + get_contraseña() + "'";
        num_filas = base2.Consultar("usuarios", sql).Tables["usuarios"].Rows.Count;

        if (num_filas > 0)
        {
            r = true;
            ptr.ImageLocation = Application.StartupPath + @"\Iconos\LIST01.png";

        }
        else
        {
            r = false;
            ptr.ImageLocation = Application.StartupPath + @"\Iconos\MAL1.png";

        }

    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message.ToString ());
        r = false;
    }

    return r;
}

```

**Figura 105: “Código C# Método valida usuario de la clase Login Usuario de “RFIDBussinesSoft””**

**Fuente:** Los Autores

## 4.11. Creación de Gestión de Productos

Una vez ingresados los productos, se podrá asignar tags RFID de identificación a cada uno de ellos, para al momento de facturar cada tag nos de la información del producto, de esta manera se autorizará a facturar de una manera rápida y tener un control de stock de productos.

### 4.11.1. Clase Tags

En esta clase se generará códigos de tags para asignar a cada producto un tag de esta forma tener identificado cada producto en su respectivo grupo.

#### 4.11.1.1. Llamadas a librerías de la Clase Tags

En esta clase se verificará las librerías de la clase Tags (Ver Figura 107).

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using RFIDBussinesSoft.PaqueteBaseDatos;
using RFIDBussinesSoft.ClasesRFID;
```

Figura 106: “Código C# Llamada a librerías de la clase Tags de “RFIDBussinesSoft””

Fuente: Los Autores

#### 4.11.1.2. Método Genera tags de la Clase Tags

En este método se generará código de tags dependiendo del grupo de cada producto para después poder asignarlo, esta generación de tags van de 0 a 1000. (Ver Figura 108).

```

public void genera_tags(ProgressBar prol)
{
    string tag = "";
    for (int i = 1; i <= 1000; i++)
    { if (i <= 9)
      { tag = get_cod() + "000" + i;
      }else if(i>=10 && i<=99){
        tag = get_cod() + "00" + i;
      } else if (i >= 100 && i <= 999)
      { tag = get_cod() + "0" + i;
      } else if (i >= 1000)
      {
          tag = get_cod() + i;
      }
      prol.Value = i;
      ingresa_tags(tag); }
}

```

Figura 107: “Código C# Método genera tag de la clase Tags de “RFIDBussinesSoft””

Fuente: Los Autores

#### 4.11.1.3. Métodos escoger grilla Tags de la Clase Tags

Este método fue creado para seleccionar un código en el DataGridView de generación de código para que de esta manera asignar a un producto y escribirlo en el tags RFID con “Rows” y “RowIndex”. (Ver Figura 109).

- Rows y RowIndex: permite tomar los valores de la grilla de la posición y columna deseada

```

public Boolean escoger_grilla(DataGridView tbl1, DataGridViewCellEventArgs ex)
{
    string estado = "";

    codpro = tbl1.Rows[ex.RowIndex].Cells[0].Value.ToString();
    estado = tbl1.Rows[ex.RowIndex].Cells[4].Value.ToString();
    if (estado.Equals("ACTIVADO") || codpro.Equals(""))
    {
        ban = false;
    }
    else
    {
        ban = true;
    }
    return ban;
}

```

**Figura 108: “Código C# Método escoger grilla de la clase Tags de “RFIDBussinesSoft””**

**Fuente:** Los Autores

#### 4.11.1.4. Método escoge código tag asignación de la Clase Tags

En este método seleccionado el código tag generado automáticamente para asignar a un producto deseado, desde dos DataGridView por medio de un BOTÓN. (Ver Figura 110)

```

public void escoge_cod_tag(DataGridView tbl1, DataGridViewCellEventArgs ex, TextBox t)
{
    codtag = tbl1.Rows[ex.RowIndex].Cells[0].Value.ToString();
    tag_a = tbl1.Rows[ex.RowIndex].Cells[2].Value.ToString();
    t.Text = tag_a;
}
private int get_codtag()
{
    return int.Parse(codtag);
}
public void Tags_asignacion(TextBox tcod, TextBox tpass)
{
    string sql;
    sql = "Update seguridad_prod set IdEstado=1 where IdSeg=" + get_codtag() + "";
    b = new BaseRFID();
    b.comandos(sql);
    escri_rfid = new EscrituraRFID();
    escri_rfid.escribir_datos_rfid(tcod, tpass);
}

```

**Figura 109: “Código C# Método escoger cod tag y tags asignados de la clase Tags de “RFIDBussinesSoft””**

**Fuente:** Los Autores

#### 4.11.2. Clase Reportes Productos

Esta clase fue creada para ver los reportes de productos donde se verifica cada producto y el stock de los mismos por medio de una lista de productos y estadísticas en forma de pastel.

##### 4.11.2.1. Llamada a librerías de la Clase Reportes Productos

Verificamos las librerías de la clase Reportes productos (Ver Figura 111).

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using RFIDBussinesSoft.Reportes_cristal;
using RFIDBussinesSoft.ReportesDataSets;
using RFIDBussinesSoft.PaqueteBaseDatos;
using MySql.Data.MySqlClient;
```

Figura 110: “Código C# Llamada a librerías de la clase Reporte Productos de “RFIDBussinesSoft””

Fuente: Los Autores

##### 4.11.2.2. Métodos Crear reporte de la Clase Reportes Productos

En este método se llamará al CristalReports para crear los reportes de productos. (Ver Figura 112).

```

public dataProductos  crea_reporte_prod(string tabla,string sql)
{
dataProductos dtsRe;
MySQLConnection oCon;
MySQLDataAdapter Mda;
b = new BaseRFID();
dtsRe = new dataProductos ();

oCon = new MySqlConnection(b.conectar().ConnectionString);
Mda = new MySQLDataAdapter(sql, oCon);
try{
    oCon.Open();
    Mda.Fill(dtsRe, tabla);

}catch(Exception){

}finally{
    oCon.Close();
}

return dtsRe;
}

```

**Figura 111: “Código C# Método Crear reporte de la clase Reportes Productos de “RFIDBussinesSoft””**

**Fuente:** Los Autores

## **4.12. Creación de Gestión de Robos**

En esta gestión se registrará de forma automática los intentos de hurtos de productos, del local comercial a través de dos DataGridView, donde uno mostrará el producto que se está sustrayendo y el otro registrará todas las incidencias que se generen.

### **4.12.1. Clase Seguridad**

En esta clase de seguridad se tendrá consultas a los tags asignados, pero solo los que siguen habilitados, no los que ya fueron deshabilitados por compra realizada.

### **4.12.2. Llamada a Bibliotecas de la Clase Seguridad**

A continuación, se analizará las librerías de la clase seguridad (Ver Figura 113).

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Resources;
using RFIDBussinesSoft;
using RFIDBussinesSoft.Interfaces;
using RFIDBussinesSoft.PaqueteBaseDatos;
using MySql.Data.MySqlClient;

```

**Figura 112: “Código C# Llamada a librerías de la clase Seguridad de “RFIDBussinesSoft””**

**Fuente:** Los Autores

#### 4.12.3. Método de Consulta tags asignado de la Clase Seguridad

Este método de consulta es para verificar si un producto está siendo sustraído, se trabajará con sentencias Sql, desde el DataGridView. (Ver Figura 114).

```

public void consultar_Tags_asig(DataGridView tabla, string dato)
{
    string sql="";
    if(comprueba_dato(dato))
    {
        sql = "Select s.IdSeg, s.CodPro, s.Tag, p.Nombre_Producto, p.Descripcion_Producto, e.Tipo from seguridad_prod s,
        ba = new BaseRFID();
        tabla.DataSource = ba.Consultar("seguridad_prod", sql);
        tabla.DataMember = "seguridad_prod";
    }
}

```

**Figura 113: “Código C# Método consulta tag asignados de la clase Seguridad de “RFIDBussinesSoft””**

**Fuente:** Los Autores

#### 4.12.4. Método Comprueba Dato de la Clase Seguridad

Este método es para comprobar el estado del producto si esta activado o desactivado de esta forma:

- 1 activado
- 2 desactivado

De esta forma podrá registrar que producto está siendo robado. (Ver Figura 115).

```
private Boolean comprueba_dato(string t)
{
    Boolean ban = false;
    string sql = "";
    int total_filas = 0;
    try
    {
        ba = new BaseRFID();
        sql = "Select * from seguridad_prod where Tag='" + t + "' and IdEstado=1";
        total_filas = ba.Consultar("seguridad_prod", sql).Tables["seguridad_prod"].Rows.Count;
        if (total_filas > 0)
        {
            ban = true;
        }
        else
        {
            ban = false;
        }
    }
    catch (Exception e)
    {
        Console.WriteLine(e.Message.ToString());
    }
    return ban;
}
```

**Figura 114: “Código C# Método comprueba dato de la clase Seguridad de “RFIDBussinesSoft””**

**Fuente:** Los Autores

## CAPÍTULO V

### **5. PRUEBAS Y RESULTADOS DEL SISTEMA PROTOTIPO “RFIDBUSSINESSOFT”**

En este capítulo, se verificará el funcionamiento óptimo del sistema “RFIDBussinesSoft”, con la tecnología RFID (Identificación por radio Frecuencia).

RFIDBussinesSoft, es un sistema que está escrito plenamente en código C#, por lo que es un software desarrollado en código abierto, la plataforma donde se lo elaboró es Visual Estudio .NET 2008 express versión para experimentos y estudiantes, es así que sus respectivas pruebas fueron realizadas en el sistema operativo Windows versión XP.

#### **5.1. Pruebas y Resultados del Sistema Prototipo “RFIDBussinesSoft”**

Para verificar los resultados que arrojan las pruebas del sistema “RFIDBussinesSoft”, se realizará pruebas de conectividad con el lector RFID, donde se verificará el rango de cobertura adecuada para poder identificar los Tags tanto de usuarios como los de productos.

También se realizarán pruebas a la conexión con la base de datos ya que el sistema trabaja de manera frecuente con la base de datos, por lo que, se manejará abundante información de las diferentes gestiones del sistema.

##### **5.1.1. Conexión del Puerto serial del Lector RFID a la máquina donde trabaja el Sistema “RFIDBussinesSoft”**

Antes de iniciar el sistema se tendrá que conectar el puerto serial del Lector RFID en la máquina en la que va a trabajar el sistema. Verificando el código de colores que tiene el equipo. (Observar Tabla 26 y Ver Figura 116).

Lector Integrado RFID	
Rojo: +9 -12 VDC	Negro: Poner a Tierra (GND)
Verde: Transmitir Datos (TXD)	Blanco: Recibir Datos (RXD)
Amarillo: Datos 0	Azul: Datos1
Púrpura: 485 R+	Naranja: 485 R-
Plomo: Disparador	Café: Poner a Tierra (GND)

**Tabla 26: Código de Colores para la “Configuración de Lector RFID”**

**Fuente:** Los Autores



**Figura 115: “Conectores “Lector RFID””**

**Fuente:** Los Autores

### 5.1.2. Ingreso al Sistema Prototipo “RFIDBussinesSoft”

Para ingresar al sistema se necesitará:

- Nombre de Usuarios
- Contraseña

Estos dos parámetros de entrada son previamente creados por el administrador de “RFIDBussinesSoft”, de esta manera el sistema tiene seguridad de ingreso. Una vez escrito el nombre del usuario y la contraseña correctamente, se ingresará al sistema. (Ver Figura 117).



Figura 116: “Pantalla de ingreso de “RFIDBussinesSoft””

Fuente: Los Autores

### 5.1.3. Menú del Sistema Prototipo “RFIDBussinesSoft”

A continuación, se podrá observar el menú del sistema. Este menú consta de ocho gestiones de trabajo además de una opción para salir del sistema y otra para cambiar de usuario, se puede verificar que usuario abrió sección en la parte inferior izquierda de la pantalla. (Ver Figura 118).



**Figura 117: “Pantalla del Menú de “RFIDBussinesSoft””**

**Fuente:** Los Autores

La imagen anterior muestra el menú del sistema con todas las opciones activadas este menú es del usuario administrador, las opciones activadas del menú depende del tipo de usuario. Si el usuario por ejemplo es “Guardia” solamente se le activara la opción “Gestión Robos”. (Ver Figura 119).



**Figura 118: “Pantalla del Menú de Usuario Guardia “RFIDBussinesSoft””**

**Fuente:** Los Autores

#### 5.1.4. Gestión de Productos del Sistema Prototipo “RFIDBussinesSoft”

En la opción Gestión de productos se encontrarán dos operaciones “Control de Stock” y Reportes y estadísticas, las mismas que ayudarán a gestionar los productos. (Ver Figura 120).

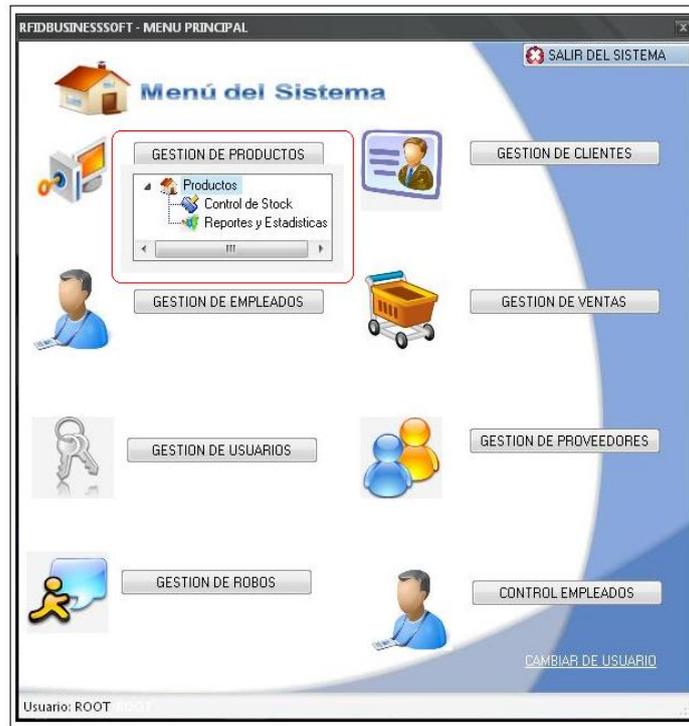


Figura 119: “Pantalla del Menú del Sistema opciones Gestión de Productos “RFIDBussinesSoft””

Fuente: Los Autores

##### 5.1.4.1. Control de Stock

En esta operación de Gestión de Productos, se podrá realizar diversas actividades como ingresos, actualizaciones, búsquedas y asignaciones de tags RFID a productos, todas estas operaciones en una pantalla, con tres pestañas diferentes. (Ver Figura 121).



Figura 120: “Pantalla Ingreso Actualización de Gestión Productos “RFIDBussinesSoft””

Fuente: Los Autores

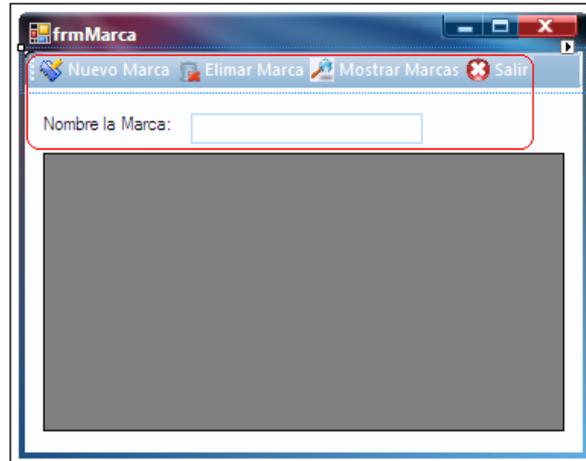
A continuación se ingresará o actualizará un producto y de esta manera, se probará si la información por ingresar es correcta, por medio de vistos verdes e incorrecta por medio de equis rojas. (Ver Figura 122).



Figura 121: “Pantalla Ingreso/Actualización verificación de ingreso y actualización Gestión de Productos “RFIDBussinesSoft””

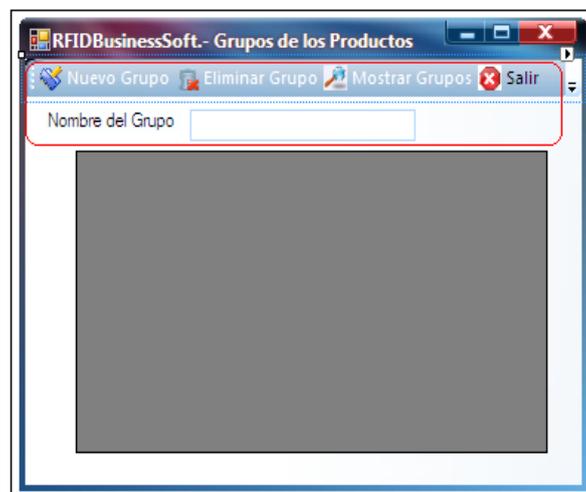
Fuente: Los Autores

En los ingresos de productos, se tendrán dos opciones adicionales para ingresar nueva marca y nuevo grupo, estos son para identificar en forma más fácil cada producto ya que un mismo producto puede tener varias marcas o grupos. (Ver Figura 123 y 124).



**Figura 122: “Pantalla Ingreso de Marca Gestión de Productos “RFIDBussinesSoft””**

**Fuente:** Los Autores



**Figura 123: “Pantalla Ingreso de Grupo Gestión de Productos “RFIDBussinesSoft””**

**Fuente:** Los Autores

Como muestra las figuras anteriores, se ingresan nuevos grupos y marcas, por cada producto y volvemos a la pantalla de ingreso/Actualización.

Ahora se verificará si el ingreso de un nuevo producto es correcto y se guardará en la base de datos del sistema, mediante un mensaje de información. (Ver Figura 125).



Figura 124: “Pantalla Ingreso mensaje de ingreso de productos Gestión de Productos “RFIDBussinesSoft””

Fuente: Los Autores

Una vez ingresado correctamente el producto se probará la consulta. En la consulta se tendrán parámetros y criterios de búsqueda. (Ver Figura 126).

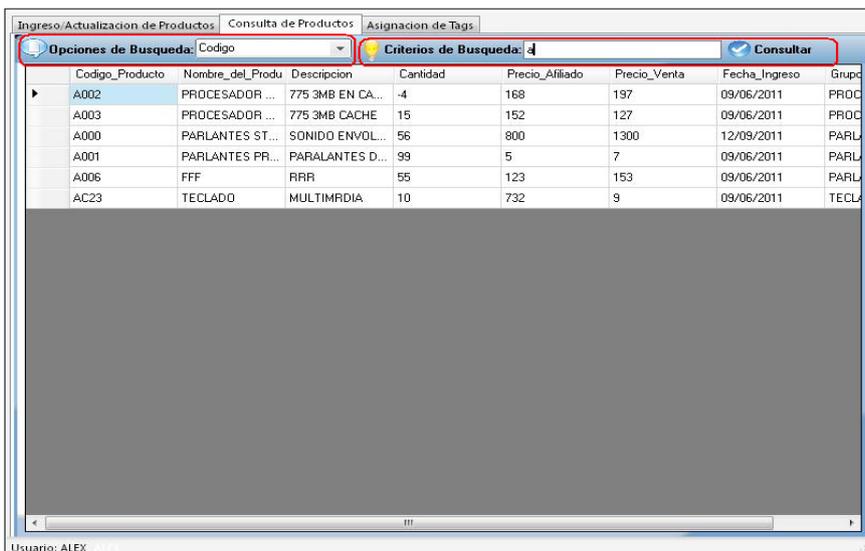


Figura 125: “Pantalla consulta de productos Gestión de Productos “RFIDBussinesSoft””

Fuente: Los Autores

Es importante señalar que todo el proceso anterior es similar para las otras gestiones en lo que se refiere a ingreso, actualización y consultas.

Después de ingresar, actualizar y consultar productos, se tendrá que asignar un tag RFID a cada uno de ellos, en esta pantalla tendremos la opción de generar tags, asignar tags y Activar RFID. (Ver Figura 127).

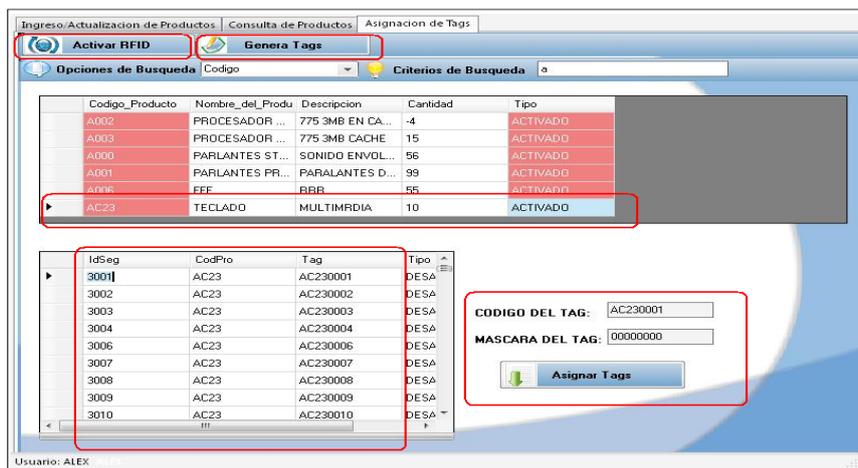


Figura 126: “Pantalla Asignación de Tags Gestión de Productos “RFIDBusinessSoft””

Fuente: Los Autores

Antes de asignar un tags RFID a un producto se tendrá que activar al Lector RFID, esta activación es abrir el puerto serial, y seleccionar la forma que se usará el puerto ya sea automática o manual para la comunicación entre el lector y el sistema. (Ver Figura 128).

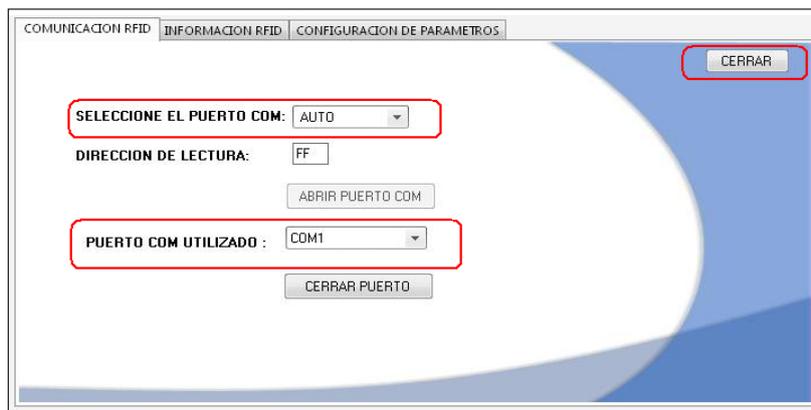


Figura 127: “Pantalla Abrir puertos seriales Gestión de Productos “RFIDBusinessSoft””

Fuente: Los Autores

En la ventana anterior también se tendrán dos opciones más que son de verificación de información y parámetros del lector RFID. (Ver Figura 129).

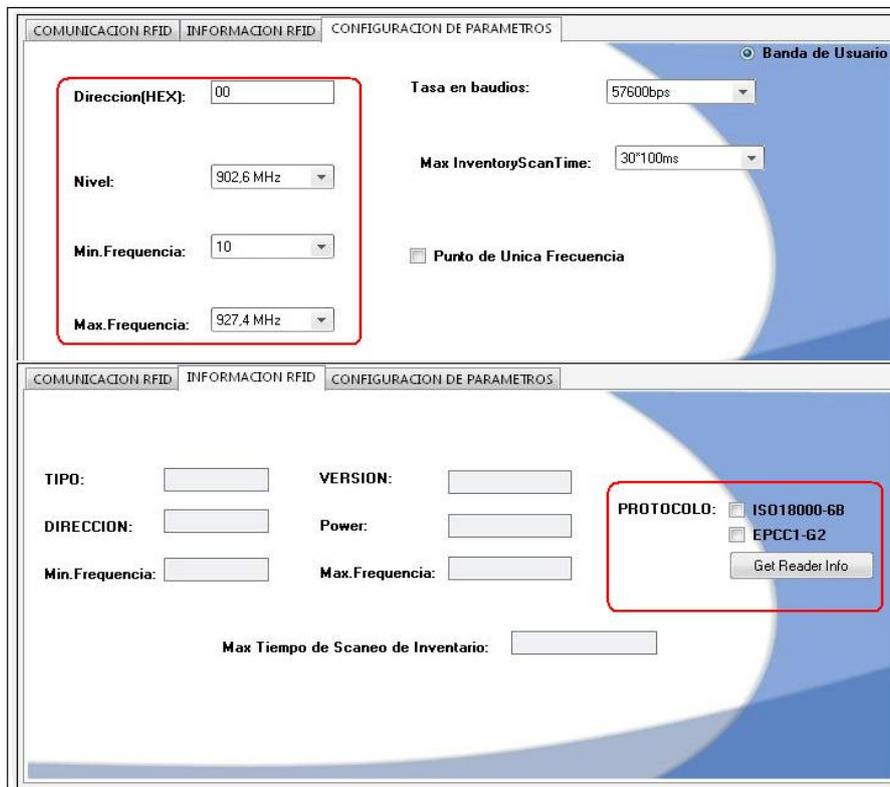


Figura 128: “Pantalla información y parámetros RFID Gestión de Productos “RFIDBussinesSoft””

Fuente: Los Autores

Cuando ya se tenga activado el puerto de conexión con el lector RFID, y generados los códigos de tags RFID, se podrá asignar un tag RFID a cada producto. (Ver figura 130).



Figura 129: “Asignación Tags RFID a producto Gestión de Productos “RFIDBussinesSoft””

Fuente: Los Autores

### 5.1.4.2. Reportes y Estadísticas

En esta opción se podrá verificar reportes estadísticos del stock de productos por medio del sistema. (Ver Figura 131).

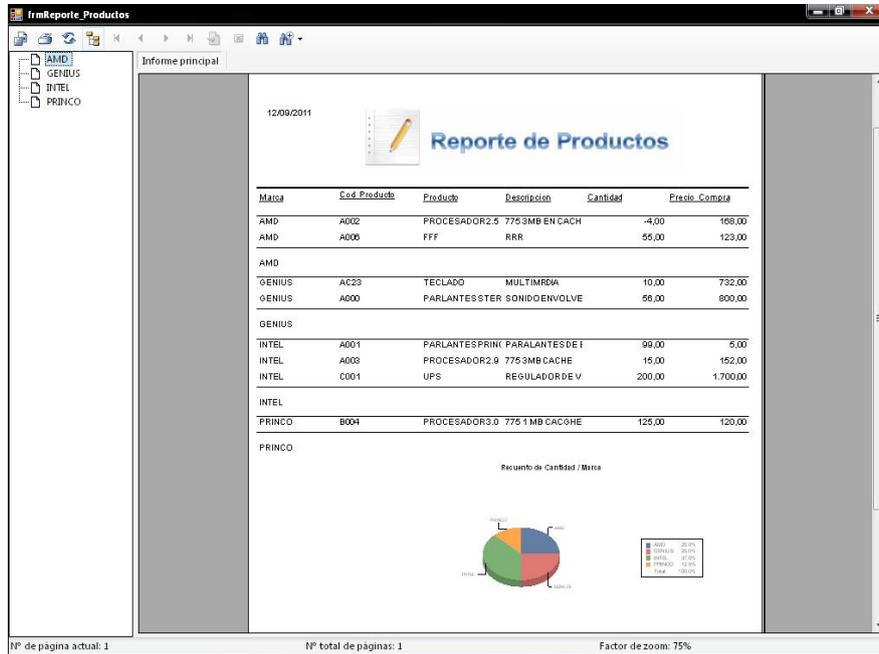


Figura 130: “Reportes de producto Gestión de Productos “RFIDBussinesSoft””

Fuente: Los Autores

### 5.1.5. Gestión de Clientes del Sistema Prototipo “RFIDBussinesSoft”

En esta gestión se trabajará con dos opciones en la primera se realizará ingresos, actualización o consultas de clientes y en la segunda se visualizará los reportes de estos. (Ver Figura 132).



Figura 131: “Pantalla del Menú del Sistema opciones Gestión de Clientes “RFIDBussinesSoft””

Fuente: Los Autores

### 5.1.5.1. Control de Clientes

Para el ingreso, actualización y consultas, se trabaja de la misma forma que en gestión de productos, la única diferencia que no se asignan Tags RFID. (Ver Figura 133 y 134).



Figura 132: “Pantalla Ingreso mensaje de ingreso de Clientes Gestión de Clientes “RFIDBussinesSoft””

Fuente: Los Autores

Codigo_Cliente	Nombres_del_Cl	Direccion	Telefono_Conve	Celular	Fecha_Ingreso	Provincia	Canton
1	ALEX LENIN ...	SOLANDA	22680667	87744992	13/06/2011	PICHINCHA	QUITO
2	VINICIO ART...	EL VALLE	22680667	98925026	20/06/2011	PICHINCHA	RUMINAHUI
3	JUAN JOSE F...	DAULE BAJO	234567891	14785236	20/06/2011	GUAYAS	DAULE
4	LUCIA GUAD...	PASAJE	111111111	111111111	23/06/2011	EL_ORO	EL_GUABO
5	JORGE MAN...	CALES EL SO...	22680667	8123456	12/09/2011	IMBABURA	PIMAMPIRO
C001	JUAN MARCO...	LA PRADERA	22845114	87123674	07/06/2011	CARCHI	MIRA

Figura 133: “Pantalla Consulta clientes Gestión de Clientes “RFIDBussinesSoft””

Fuente: Los Autores

### 5.1.5.2. Reportes

En esta opción se podrá verificar reportes de clientes por medio del sistema. (Ver Figura 135).

Codigo	Nombre	Direccion	Telefono	Fecha ingreso
1	ALEX LENIN SALAZAR	SOLANDA	22680667	13/06/2011
2	VINICIO ARTURO SALA	EL VALLE	22680667	20/06/2011
3	JUAN JOSE FLORES RI	DAULE BAJO	234567891	20/06/2011
4	LUCIA GUADALUPE AR	PASAJE	111111111	23/06/2011
5	JORGE MANUEL RICAL	CALES EL SOL 185 Y U	22680667	12/09/2011
C001	JUAN MARCO POLO BI	LAPRADERA	22845114	07/06/2011

Figura 134: “Reporte clientes Gestión de Clientes “RFIDBussinesSoft””

Fuente: Los Autores

### 5.1.6. Gestión de Empleados del Sistema Prototipo “RFIDBussinesSoft”

En esta gestión se trabajará con dos opciones en la primera se realizará ingresos, actualización o consultas y asignación de tags empleados, en cambio en la segunda se visualizará los reportes de estos. (Ver Figura 136).



Figura 135: “Pantalla del Menú del Sistema opciones Gestión de Empleados “RFIDBussinesSoft””

Fuente: Los Autores

#### 5.1.6.1. Gestión Empleados

Para el ingreso, actualización y consultas, se trabaja de la misma forma que en gestión de productos, también posee una opción de asignación de Tags RFID diferente a la de productos. (Ver Figura 137 y 138).



Figura 136: “Pantalla Ingreso mensaje de ingreso de empleados Gestión de Empleados “RFIDBussinesSoft””

Fuente: Los Autores

Codigo_Empleado	Numero_de_Identifi	Nombres_del_Empl	Cargo	Telefono	Celular	Fecha_Ingreso	
E001	171525255	JULIAN BASTID...	VENDEDOR	545121212	98552521	14/06/2011	N
E002	171123544	CARLOS JUSTO...	VENDEDOR	545452245	92421542	14/06/2011	N
E003	1711459887	MARIA DEL CAR...	VENDEDOR	256852187	95181441	09/06/2011	N
E009	1715598692	ALEX LENIN SA...	EJECUTIVO	22680667	87744992	12/06/2011	N

Figura 137: “Pantalla Consulta empleados Gestión de Empleados “RFIDBussinesSoft””

Fuente: Los Autores

Una vez ingresado correctamente el empleado, se dirigirá a la opción de asignar tags para asignar tags RFID a los empleados para su control de horarios, una vez seleccionado el empleado se asignará el tag y por medio de un mensaje de información se sabrá que el tag fue asignado correctamente al empleado y guardado en la base de datos. (Ver Figura 139).

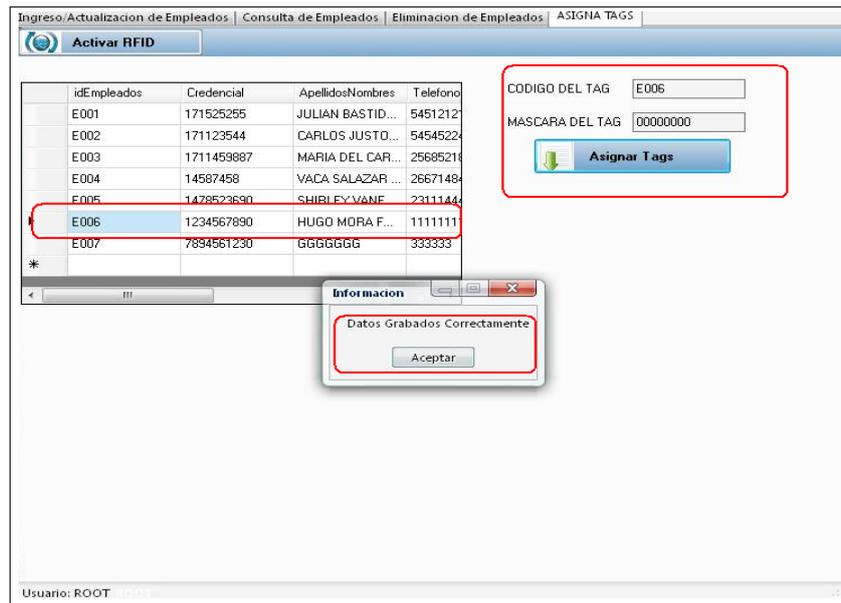


Figura 138: “Pantalla asigna tags a empleados Gestión de Empleados “RFIDBusinessSoft””

Fuente: Los Autores

### 5.1.6.2. Reportes

En esta opción se podrá verificar reportes de empleados por medio del sistema. (Ver Figura 140).

ID Empleado	Apellido/Nombre	Direccion	Telefono	Fecha Ingreso
1	ALEX LENIN SALAZAR	SOLANDA	22880067	13/06/2011
2	VINICIO ARTURO SALA EL VALLE		22880067	20/06/2011
3	JUAN JOSE FLORES RIVERA	PAULLEBAJO	234637891	23/06/2011
4	LUCIA GUADALUPE RIVERA	PASAJE	1111111111	23/06/2011
5	JORGE MANUEL RICAL CALES	EL SOL 185 YU	22880067	12/06/2011
0001	JUAN MARCO POLO BOLA	LA PRADERA	228-8114	07/06/2011

Figura 139: “Reporte clientes Gestión de Clientes “RFIDBusinessSoft””

Fuente: Los Autores

### 5.1.7. Gestión de Ventas del Sistema Prototipo “RFIDBussinesSoft”

En esta gestión se trabajará con dos opciones en la primera se realizará Facturación, y en la segunda se visualizará los detalles de facturación por medio de reportes. (Ver Figura 141).



Figura 140: “Pantalla del Menú del Sistema opciones Gestión de Ventas “RFIDBussinesSoft””

Fuente: Los Autores

#### 5.1.7.1. Facturar

Para la facturación se tendrá que ingresar los datos de la factura que son consultas a la base de datos automáticas, la típica activación del lector RFID y la del Tag RFID, se podrá agregar clientes y consultar productos. (Ver Figura 142).

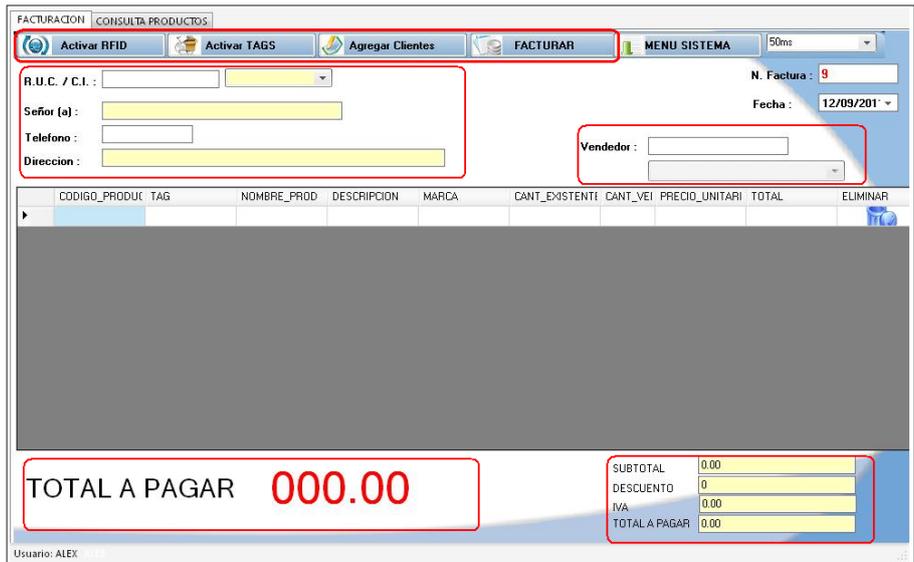


Figura 141: “Pantalla de Facturación Gestión de Ventas “RFIDBussinesSoft””

Fuente: Los Autores

Una vez activado el lector RFID y los Tags se podrá probar la facturación de manera rápida y automática de todos los productos a facturar. (Ver Figura 143).

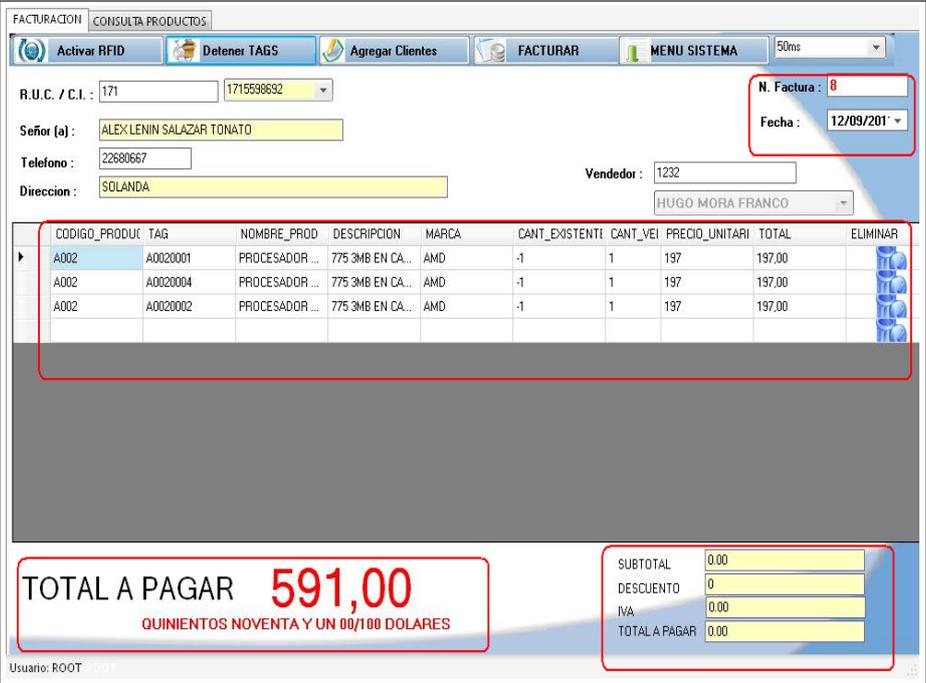


Figura 142: “Pantalla de Facturando por venta Gestión de Ventas “RFIDBussinesSoft””

Fuente: Los Autores

Una vez que se realizó todo el proceso de la figura anterior, se imprimirá la factura por la compra realizada en el botón “FACTURAR”.

### 5.1.7.2. Detalles de Venta

En este reporte se podrá visualizar la factura que se imprimirá por cada venta realizada. (Ver Figura 144).

Informe principal

12092011 **FACTURA Nº** 7

**Cliente** ALEX LENIN SALAZAR TOBATO **Dirección** SOLANDA

RUC/CC 1716586992 Telefono 22680667 Vendedor JULIAN BASTIDAS

Cantidad	Detalle	Materia	Precio
1,00	175.3M F EN CACHE	AMD	197,20
1,00	175.3M F EN CACHE	AMD	197,20

Subtotal	41,28
Descuento	0
<b>Total</b>	<b>41,28</b>

FIRMA AUTORIZADA \_\_\_\_\_ RECIBI CONFORME \_\_\_\_\_

Nº de página actual: 1 Nº total de páginas: 1 Factor de zoom: 60%

Figura 143: “Factura por venta Gestión de Ventas “RFIDBussinesSoft””

Fuente: Los Autores

### 5.1.8. Gestión de Usuarios del Sistema Prototipo “RFIDBussinesSoft”

En esta gestión se trabajará con dos opciones: en la primera se realizará control de usuarios, y en la segunda se observará los usuarios que tiene el sistema por medio de reportes. (Ver Figura 145).



**Figura 144: “Pantalla del Menú del Sistema opciones Gestión de Usuarios “RFIDBussinesSoft””**

Fuente: Los Autores

### 5.1.8.1. Control de Usuarios

Para el ingreso, actualización, se trabaja de la misma forma que en gestión de productos, pero además en esta operación también se otorgará permisos de trabajo y entrada al sistema. (Ver Figura 146).



**Figura 145: “Pantalla Ingreso de usuarios Gestión de Usuarios “RFIDBussinesSoft””**

Fuente: Los Autores

Después de ingresar los usuarios se procederá a otorga los permisos de uso del sistema según su rol en la empresa. (Ver Figura 147).

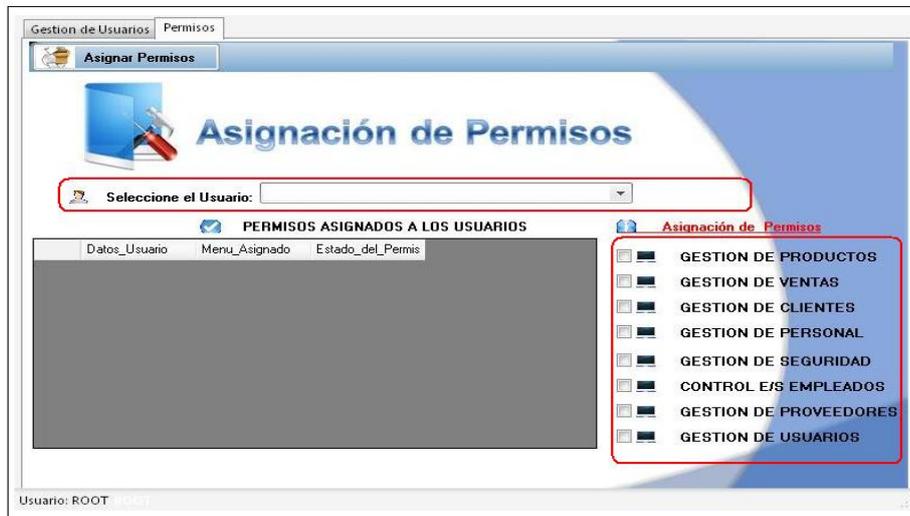


Figura 146: “Pantalla asignación de permisos a usuarios Gestión de Usuarios “RFIDBussinesSoft””

Fuente: Los Autores

Una vez en la pantalla de asignación de permisos se seleccionará un usuario, y se procederá a asignar permisos de las gestiones de acuerdo a su rol o cargo, dando clic en el botón Asignar permisos. (Ver Figura 148).

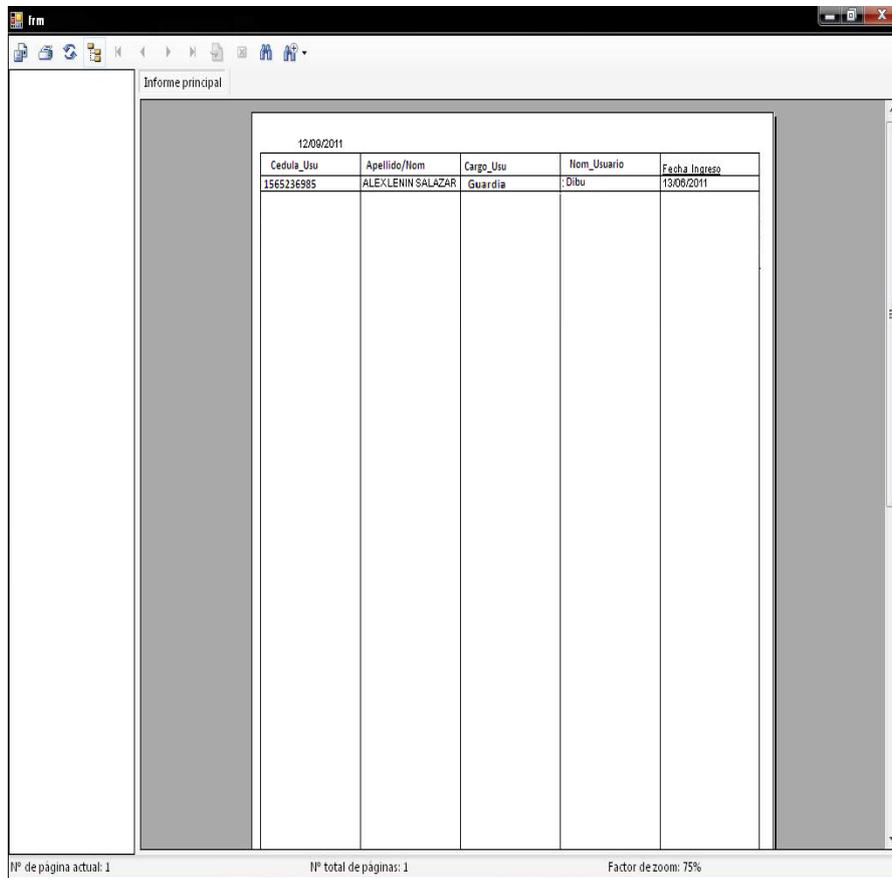


Figura 147: “Pantalla asignando permisos a usuarios Gestión de Usuarios “RFIDBussinesSoft””

Fuente: Los Autores

### 5.1.8.2. Reportes

En esta opción se podrá verificar reportes de usuarios que se tendrá en el sistema. (Ver Figura 149).



The screenshot shows a window titled 'Informe principal' with a table of user data. The table has five columns: 'Cedula\_Usu', 'Apellido/Nom', 'Cargo\_Usu', 'Nom\_Usuario', and 'Fecha\_Ingreso'. The data row shows a user with ID 1565236985, name ALEXLENN SALAZAR, position Guardia, name Dibu, and login date 13/08/2011. The window also displays '12/08/2011' at the top and status information at the bottom: 'N° de página actual: 1', 'N° total de páginas: 1', and 'Factor de zoom: 75%'.

12/08/2011				
Cedula_Usu	Apellido/Nom	Cargo_Usu	Nom_Usuario	Fecha_Ingreso
1565236985	ALEXLENN SALAZAR	Guardia	Dibu	13/08/2011

Figura 148: “Reporte usuarios Gestión de Usuarios “RFIDBussinesSoft””

Fuente: Los Autores

### 5.1.9. Gestión de Proveedores del Sistema Prototipo “RFIDBussinesSoft”

En esta gestión se trabajará con dos opciones en la primera se realizará ingresos, actualización o consultas de clientes y en la segunda se visualizará los reportes de estos. (Ver Figura 150).



Figura 149: “Pantalla del Menú del Sistema opciones Gestión de Proveedores “RFIDBussinesSoft””

Fuente: Los Autores

### 5.1.9.1. Control de Proveedores

Para el ingreso, actualización y consultas, se trabaja de la misma forma que en gestión de productos, la única diferencia que no se asignan Tags RFID. (Ver Figura 151 y 152).



Figura 150: “Pantalla Ingreso de proveedores Gestión de Proveedores “RFIDBussinesSoft””

Fuente: Los Autores

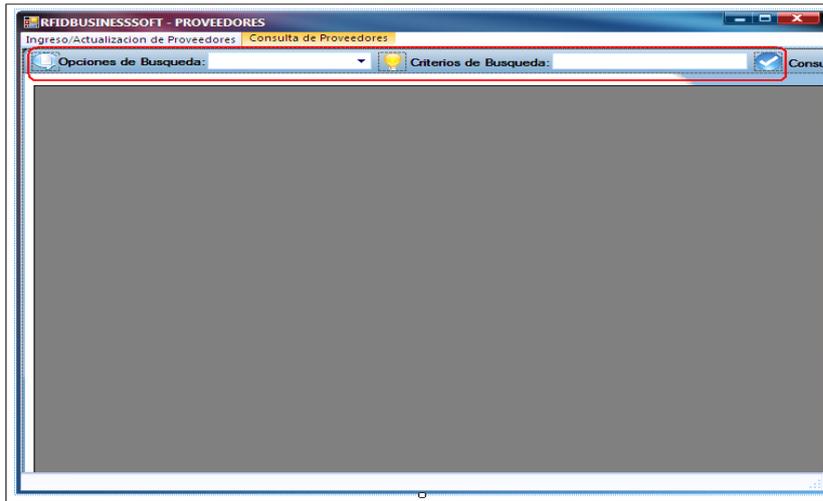


Figura 151: “Pantalla Consulta proveedores Gestión de Proveedores “RFIDBussinesSoft””

Fuente: Los Autores

### 5.1.9.2. Reportes

En esta opción se podrá verificar reportes de proveedores por medio del sistema. (Ver Figura 153).

12/08/2011				
Codigo	Nombre	Direccion	Telefono	Fecha Ingreso
1	ALEXLENIN SALAZAR	SOLANDA	22680067	13/06/2011
2	VINICIARTURO SALA	EL VALLE	22680067	20/06/2011

Figura 152: “Reporte proveedores Gestión de Proveedores “RFIDBussinesSoft””

Fuente: Los Autores

### 5.1.10. Gestión de Robos del Sistema Prototipo “RFIDBussinesSoft”

Esta gestión ayuda a verificar intentos de robo de productos que no fueron pagados, por medio de lector RFID. (Ver Figura 154).

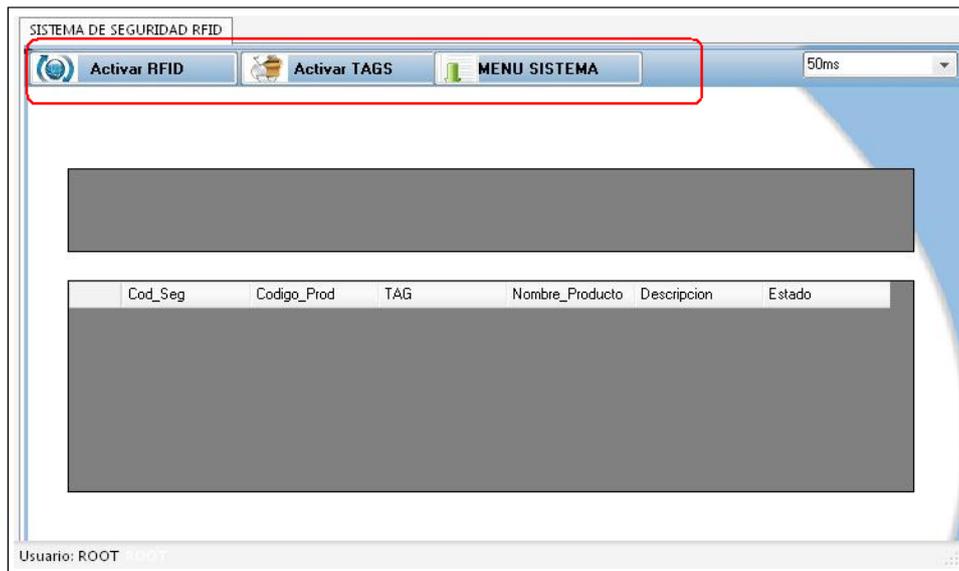


Figura 153: “Pantalla del Menú del Sistema opciones Gestión de Robos “RFIDBussinesSoft””

Fuente: Los Autores

#### 5.1.10.1. Control Antirobo

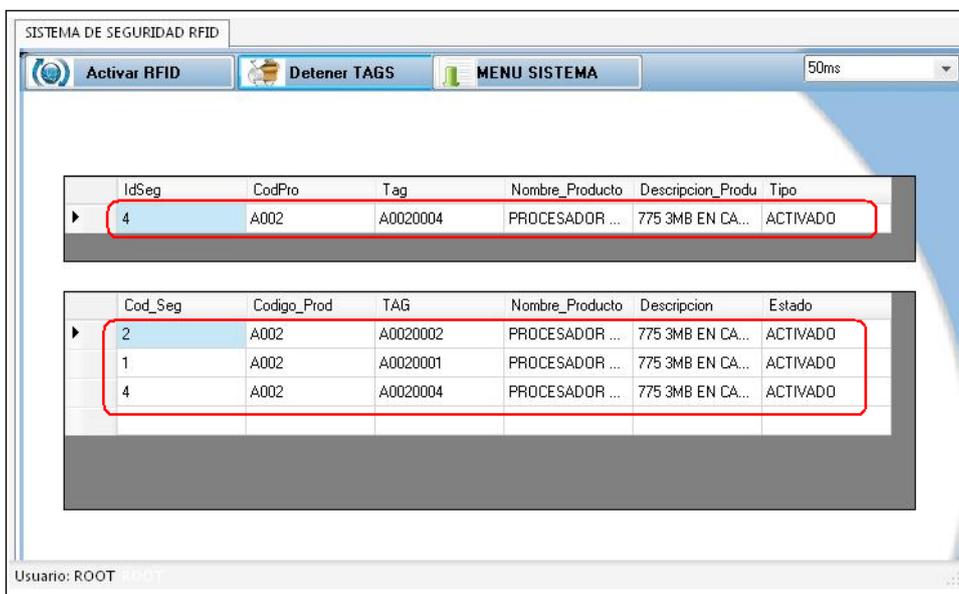
Para visualizar los intentos de robos de productos automáticamente primero tenemos que activar el lector RFID y los tags RFID, de esta manera la radio frecuencia detectará productos que están siendo sustraídos sin pagar. (Ver Figura 155).



**Figura 154: “Pantalla Anti robo Gestión de Robos “RFIDBussinesSoft””**

Fuente: Los Autores

Una vez activado el lector y tags RFID, se podrá visualizar los intentos de robos de productos de esta manera. (Ver Figura 156).



**Figura 155: “Pantalla Anti robo velicación de productos Gestión de Robos “RFIDBussinesSoft””**

Fuente: Los Autores

Según la figura 153 en el primer DataGridView se visualiza el producto que está siendo sustraído en ese momento y en el segundo DataGridView se tiene

una lista de todos los intentos de robo de productos que fueron detectados.

### 5.1.11. Control de Empleados del Sistema Prototipo “RFIDBussinesSoft”

En esta gestión ayuda a verificar las horas de entra y salida de los empleados a su lugar de trabajo, por medio de lector RFID. (Ver Figura 157).



Figura 156: “Pantalla del Menú del Sistema opciones Control Empleados “RFIDBussinesSoft””

Fuente: Los Autores

A continuación, se podrá verificar la hora de entrada y de salida de un empleado de forma automática, la hora de entrada y de salida está programada en el sistema por medio del reloj del sistema, primero se tendrá que activar el lector y los tags RFID (Ver Figura 158 y 159).

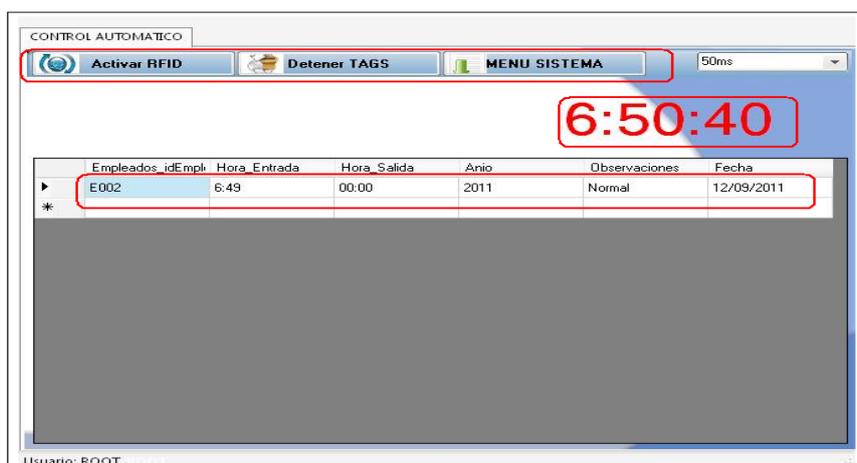


Figura 157: “Pantalla Control de entrada empleados Control Empleados “RFIDBussinesSoft””

Fuente: Los Autores



Figura 158: “Pantalla Control de salida empleados Control Empleados “RFIDBussinesSoft””

Fuente: Los Autores

### 5.1.12. Cambiar usuarios del Sistema Prototipo “RFIDBussinesSoft”

En esta opción del menú del sistema, se podrá cambiar de usuario según sea su necesidad. (Ver Figura 160 y 161).



Figura 159: “Pantalla del Menú del Sistema opciones Control Empleados “RFIDBussinesSoft””

Fuente: Los Autores



Figura 160: “Pantalla de ingreso usuario en Cambiar Usuario “RFIDBusinessSoft””

Fuente: Los Autores

## CONCLUSIONES Y RECOMENDACIONES

### Conclusiones:

Al terminar el presente proyecto, se ha llegado a las siguientes conclusiones:

- La tecnología RFID (Identificación por Radio Frecuencia), se ha convertido en una tecnología relevante al momento de diseñar software de identificación, porque cada vez se busca dar automatización de detección de información manejada y proporcionar al usuario, hoy en día es el anhelo de cualquier empresa de identificación de objetos.
- Hoy en día en nuestro país, no existen sistemas que verifiquen los precios de varios productos a la vez al momento de facturar, por no tener una estructura tecnológica adecuada ya sea por precios o falta de conocimientos. Una opción es la tecnología RFID ya que ha revolucionado la industria de la identificación automática, ofreciendo avances significativos en comparación con sistemas tradicionales como códigos de barras, tarjetas de banda magnéticas, y chip de contacto a proximidad.
- Se espera que la tecnología RFID a finales del 2012 en Ecuador baje sus costos considerablemente, lo cual permitirá la utilización continua de esta tecnología y así tener la oportunidad de desarrollar varias aplicaciones que aprovechen sus beneficios.
- Fueron tres las facultades que consideramos para trabajar con la tecnología RFID: la capacidad de poder leer tags a larga distancia sin necesidad de línea de vista, la capacidad de lectura/ escritora y la forma de identificar a objetos como únicos, estas características son fundamentales y representan una gran diferencia al comparar RFID con otras tecnologías de auto identificación.
- Es importante resaltar que la tecnología RFID no tiene un estándar normalizado definitivamente, por lo cual se presenta problemas con sus

especificaciones. En el proyecto se tuvieron dos problemas primordiales: la distancia y el número de tags que lee el lector a la vez por minuto, y su tipo de interfaz de conexión, por lo que es necesario estudiar con detenimiento las especificaciones de cada equipo RFID que se usará según sea el sistema.

- El sistema prototipo construido y desarrollado en C# intentó involucrar todos los elementos de programación, base de datos y tecnología RFID, para tener como resultado un sistema funcional que permite facturar de forma automática y rápida, contar con una base de datos sólida y tener una fácil configuración de RFID, para activar RFID, asignar y leer tags.
- Con la construcción del sistema prototipo “RFIDBusinessSoft”; no solo se demostró que se puede utilizar la tecnología RFID para un área específica, sino que se puede utilizar en múltiples áreas en diferentes funcionamientos, ya que la identificación y localización de objetos, animales y personas en los últimos años ha tomado fuerza, por diversas situaciones.
- Aunque el sistema fue desarrollado en la plataforma punto net 2008 express (para experimentos y estudiantes), el código está completamente escrito en C#, por lo que es código abierto, fomentando así el desarrollo intelectual de programadores, no solo para unos pocos si no para todos los que estén en constante aprendizaje de la tecnología.
- C Sharp admite los conceptos de encapsulación, herencia y polimorfismo, su desarrollo es más sencillo por todas las variables y métodos, y sus clases pueden heredar directamente de una clase primaria, pero pueden implementar cualquier número de interfaces.
- “RFIDBusinessSoft” proporciona una Interfaz mucho más fácil de captar, el manejo en sus diversas gestiones que otros sistemas en general, aunque posee limitaciones por ser un software prototipo, ha sido diseñado para que los usuarios sepan lo que están haciendo y observando lo que ocurre al momento de interactuar con el sistema.

- El sistema fue diseñado para que trabaje a su máxima capacidad en los sistemas operativos Windows XP SP2 -3 y Windows 7, ya que se ha probado que funciona sin problema alguno, el motivo por que el sistema no es multiplataforma, es por los drivers del lector RFID, que únicamente funcionan en sistemas operativos propietarios, la solución es desarrollar un driver que interprete el funcionamiento del lector RFID en sistemas operativos con distribución Linux.
- Para la creación del diseño conceptual de la base de datos se utilizó el programa DBDesigner 4.0. Cabe resaltar que la base con la que se trabajó es MySQL por que soporta gran cantidad de tipos de datos para las columnas, como también puede gestionar usuarios y passwords, manteniendo un muy buen nivel de seguridad en los datos y este motor de base de datos es gratuito para trabajar en cualquier proyecto.
- Para este proyecto, se trabajó con un lector RFID, el mismo que realiza la función de lectura/escritura, y tiene una antena integrada de 7 dbi, para aumentar su rango de lectura de 3 a 5 metros y Tags RFID UHF EPC G2, de esta forma las pruebas realizadas con estos componentes en conexión con el sistema, fueron satisfactorias.

## Recomendaciones

En el estudio y elaboración del sistema “RFIDBussinesSoft” para una mejor funcionalidad recomendamos:

- Al trabajar con punto net 2008 se recomienda agregar referencia de la base de datos MySql colocándonos sobre la pestaña proyecto agregar referencia, sobre la opción. NET buscamos “MySql. Data” y aceptamos; ya que de otra manera no se podrá realizar ningún tipo de operación, ni de conexión con la base de datos.
- Para trabajar con el lector RFID y manipular la información que nos brindan los tags por medio del sistema, este se realizó mediante programación con código C#, agregando el driver de funcionamiento “RRU1861” que es exclusivo para C# en la carpeta de referencias del sistema “RFIDBussinesSoft”.
- Para la validación de los ingresos de las diferentes gestiones, se recomienda crear una clase general de validaciones donde cada clase de ingreso llame a los métodos de la clase validación, de esta forma ahorramos recursos del sistema y tiempo en la programación.
- Para verificar el estado en los ingresos se trabaja con marcas y banderas, se necesita activar los elementos “THIS” que no están activados en esta versión de .NET, en la clase de los formularios de ingresos de este modo si la información escrita en un texto al momento de ingresar información a la base de datos no es la correcta aparece una cruz roja y si esta correcta aparece un visto verde.
- Es recomendable añadir librerías “USING” en las clases según su uso: por ejemplo para trabajar entre clases de RFID, base de datos o de las diferentes gestiones, de no ser así al momento de realizar cualquier operación el sistema arrojará errores.

- Para descargar código y paquetes que brindan funcionalidad adicionales al sistema prototipo en C# visitar la página de [solocodigo.com](http://solocodigo.com) que día a día esta innovando con aportaciones de diferentes programadores para tener ideas claras al momento de realizar algún proyecto.
- Es recomendable que el equipo donde trabajará el sistema “RFIDBusinessSoft” sea robusto tanto en el procesador como en la RAM, ya que el sistema constantemente trabaja con consultas a la base de datos, conexión de lector RFID y activación de Tags UHF, por lo que ocupa una cantidad considerable de recursos.
- Se recomienda para que el sistema sea optimo utilizar tags que se puedan leer a larga distancia y sean multi escritura, para la gestión de empleados ya que algunos tags solo permiten escribir una sola vez.
- Para la gestión de ventas y productos es recomendable utilizar Tags UHF de lectura y escritura adhesivos desechables ya que una vez que el producto sea comprado, el tag no se podrá utilizar jamás en el sistema.
- Se recomienda la utilización de la Tecnología RFID en: controles de acceso, en las bibliotecas, pago de peajes en las autopistas, identificación de equipajes en los aeropuertos, inmovilización de automóviles, localización de personal en edificios, y otras aplicaciones en las que no afecte la intimidad de las personas.
- Para desarrollar sistemas grandes con la de Tecnología RFID, se recomienda esperar hasta que alcance un buen nivel de desarrollo masivo y los costos de los equipos disminuyan considerablemente sobre todo en Latino América.
- Debido a la gran cantidad de información y su importancia, la misma que se está almacenando en la base, se recomienda sacar un respaldo semanal de la base de datos.

- Para proteger el funcionamiento del sistema, se recomienda el uso de reguladores de voltaje en el lector RFID ya que una variación de la misma podría causar daños graves al equipo.
- Antes de la conexión del equipo RFID a la PC se recomienda estudiar y analizar el código de colores que trae el lector, para no tener complicaciones al momento de trabajar con este.
- Se recomienda al momento de la creación del usuario administrador poner un nombre de usuario claro y una contraseña segura para no tener intrusos y personas ajenas al sistema.
- Es recomendable planear adecuadamente, donde se ubicará el lector RFID, para que no existan interferencias de ningún tipo, ya sea lector tags o viceversa, en el momento de facturación, control de robos y registro de entrada/salida del trabajo por seguridad, operatividad del sistema.
- Se recomienda utilizar Tags RFID especiales para metal, ya que el metal distorsiona la señal de radio frecuencia, y podría haber pérdidas considerables por este problema que tiene RFID.

# BIBLIOGRAFÍA

## Libros:

- **ARROYO LUIS - BRAIS QUIRÓS - MURILLO GERARDO.** “RFID y su capacidad de identificaciones - Universidad de las Ciencias Tecnológicas”, Volumen I, San José - Costa Rica, Ed. McGraw-Hill, 2007.
- **FERNANDO J – GLORIA E.** “Estudio de un sistema de RFID basado en EPC - Universidad de España, Volumen I, Zaragoza -España, Ed. CCS, 2005
- **JOSÉ LUIS NÚÑEZ MORA.** “Metodología para habilitar un proceso de configuración de RFID”, Volumen I, México Distrito Federal – México, Ed. Interamericano, 2009.
- **LUIS MIGUEL GODÍNEZ GONZALES.** “RFID, Oportunidades y riesgos su aplicación práctica”, Volumen I, Barcelona – España, Ed. Elsevier, 2009.
- **SYED A. AHSON - MOHAMMAD ILYAS.** “RFID Handbook: Applications, Technology, Security, and Privacy”, Volumen II, Boston - Massachusetts – Estados Unidos, Ed. CRC, 2008.
- **HARVEY LEHPAMER.** “RFID Design Principles”, Volumen II, New York City – New York - Estados Unidos, Ed. Artech House Publishers, 2009.
- **PAUL SANGHERA.** “RFID+: CompTIA RFID+ Study Guide and Practice Exam”, Volumen I, USA, Ed. Syngress, 2010.
- **HAROLD G. CLAMPITT.** “RFID Certification Textbook”, Volumen I, Chicago – Illinois – Estados Unidos, Ed. American RFID Solutions, 2010.

- **MEDINA CESAR.,** “Los Sistemas Automáticos de Identificación RFID - Universidad Autónoma Metropolitana”, trabajo de investigación el 22 de marzo del 2009.
- **PABLO ALVARES - IAN GILFILLAN.,** “La Biblia de MySql en Español”, Volumen I en español, Buenos Aires – Argentina, Ed. Progreso-a, 2008.
- **STEVENS POOLEY.,** “Utilización de UML en ingeniería de Software con objetos y componentes”, Volumen IV, Madrid – España, Ed. AddisonWesley, 2008.

### **Formatos PDF:**

- **STEVE RESNICK – RICAR CRONE – CHRIS BOWEN.,** “Fundamentos de Comunicación esencial punto Net”, Archivo PDF.
- **JHON SHARP.,** “Visual C# 2008 paso a paso”, Archivo PDF.
- **PAUL KIMMEL.,** “UNLEASHED FOR C#”, Archivo PDF.
- **DANIEL SOLIS.,** “Illustrated C# 2008”, Archivo PDF.
- **PRESSMAN.,** “Ingeniería de Software”, Edición 5, Archivo PDF.
- **SILVERSCATHZ.,** “Fundamentos de Base de Datos” Archivo PDF.
- **CONNOLLY.,** “Sistemas de Base de Datos”, Archivo PDF
- **ID CONSULTANTS.,** Catalogo 2011 de Equipos RFID de UHF de largo alcance, Archivo PDF.
- **UNIVERSIDAD FEDERAL DE SÃO CARLOS.,** “Implementando RFID en sistemas formaticos”, segunda edición, Brasil 2010, Archivo PDF.

## Páginas Web:

- Artículo educacional: “*Contenidos de Definición de Radio Frecuencia*”, Febrero 2009, “<http://www.asifunciona.com/>”, consultada Febrero de 2011.
- Artículo: “*Arquitectura RFID*”, Marzo 2010, “<http://www.todoarquitectura.com/>”, consultada Febrero de 2011.
- Artículo de DYNAMICS AX: “*RFID una tecnología en evolución*”, Julio 2009, “<http://www.aprendedynamics.com/rfid.html>”, consultada Marzo de 2011
- Artículo Investigativo: “*TAGSYS RFID presenta tags UHF*”, Septiembre 2010, “<http://www.rfidpoint.com/lanzamientos>”, consultada Marzo Diciembre de 2011.
- Artículo de debate: “*Lectores RFID*”, Enero 2011, “<http://www.rfidpoint.com/foros/lectores-rfid/>”, consultada Abril de 2011.
- Videos de demostración: “*Beneficios de la RFID en la actualidad*”, Diciembre 2010, “<http://www.rfidpoint.com/videos/beneficios-de-la-rfid-en-la-moda-y-el-retail-2/>”, consultada Abril de 2011.
- Artículo de investigación: “*El estándar EPC*”, Diciembre 2010, “<http://www.rfidpoint.com/el-estandar-epc/>”, consultada Abril de 2011.
- Artículo de investigación: “*Riesgos y Oportunidades de RFID*”, Octubre 2010, “<http://www.rfid-spain.com/articulo/70051/rfid/otros/rfid-opportunidades-y-riesgos-su-aplicacion-practica>”, consultada Mayo 2011.

- Artículo de investigación: *“Recomendaciones sobre RFID”*, junio 2010, **“<http://www.versvs.net/anotacion/recomendaciones-ue-sobre-rfid>”**, consultada Mayo 2011.
- Artículo de investigación: *“Aprenda C# en tres pasos”*, junio 2006, **“[http://www.elguille.info/NET/AprendaCSharp3/parte3\\_lo\\_nuevo.aspx](http://www.elguille.info/NET/AprendaCSharp3/parte3_lo_nuevo.aspx)”**, consultada Mayo 2011.
- Artículo de investigación: *“Ingeniería de Software paso a paso”*, Noviembre 2004, **“<http://www.monografias.com/trabajos5.shtml>”**, consultada Junio 2011.
- Artículo de investigación: *“Precios de lectores RFID y tags UHF”*, enero 2011, **“[www.idconsultants.us/RFID/precios](http://www.idconsultants.us/RFID/precios)”**, consultada Febrero 2011.
- Artículo de investigación: *“Máxima transparencia a todos los niveles con los nuevos sistema RFID en el mundo”*, enero 2011, **“<http://www.automation.siemens.com/mcms/identification-systems/es/sistemas-rfid/Pages/default.aspx>”**, consultada Julio 2011.
- Artículo de investigación: *“Clasificación de los Tags RFID”*, septiembre 2010, **“[http://www.dipolerfid.es/productos/RFID\\_tag.aspx](http://www.dipolerfid.es/productos/RFID_tag.aspx)”**, consultada Julio 2011.
- Artículo de investigación: *“Diagramas UML con Enterprise Architect profesional 7.5 en español”*, enero 2011, **“<http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=EnterpriseArchitectUML2.x>”**, consultada Agosto 2011.

# ANEXOS

## **Anexo 1: Diseño Conceptual de La Base de Datos “RFIDBussinesSoft”**

## **Anexo 2: Diagrama de Clases de “RFIDBussinesSoft”**

### **Anexo 3: Diagrama de Caso de Uso de “RFIDBussinesSoft”**

## **Anexo 4: Diagrama de Actividades de “RFIDBussinesSoft”**

## **Anexo 5: Diagrama de Secuencia de “RFIDBussinesSoft”**

## **Anexo 6: Diagrama de Componentes de “RFIDBussinesSoft”**

# Manual de Usuario de "RFIDBusinessSoft"