

PROTOTIPO DE VERIFICACIÓN DE DATOS CON TARJETA RFID A LA
ENTRADA DE VEHÍCULOS DEL IUPB

BRYAN SANTIAGO GÓMEZ ACEVEDO
JAVIER ALEJANDRO SARRAZOLA CARO
CRISTIAN ARBEY RAMÍREZ GIRALDO

INSTITUCIÓN UNIVERSITARIA PASCUAL BRAVO
TECNOLOGÍA ELECTRÓNICA
MEDELLÍN
2013

PROTOTIPO DE VERIFICACIÓN DE DATOS CON TARJETA RFID A LA
ENTRADA DE VEHÍCULOS DEL IUPB

BRYAN SANTIAGO GÓMEZ ACEVEDO
JAVIER ALEJANDRO SARRAZOLA CARO
CRISTIAN ARBEY RAMÍREZ GIRALDO

Trabajo de grado presentado para optar por el título de
Tecnólogo en Electrónica

Asesor Técnico
Diego Hernando Orozco Gómez
Ingeniero en Instrumentación y Control

INSTITUCIÓN UNIVERSITARIA PASCUAL BRAVO
TECNOLOGÍA ELECTRÓNICA
MEDELLÍN
2013

CONTENIDO

	Pág.
INTRODUCCIÓN	9
1. EL PROBLEMA	10
1.1 FORMULACIÓN DEL PROBLEMA	10
2. JUSTIFICACIÓN	11
3. OBJETIVOS	12
3.1 OBJETIVO GENERAL	12
3.2 OBJETIVOS ESPECÍFICOS	12
4. REFERENTES TEÓRICOS	13
4.1 RFID	13
4.2 CHIP	15
4.3 VEHÍCULO	16
4.4 SOFTWARE	17
4.5 MÓDULO LECTOR/ESCRITOR SL500L	18
5. METODOLOGÍA	23
5.1 TIPO DE PROYECTO	23
5.2 MÉTODO	23
5.3 TÉCNICAS DE RECOLECCIÓN DE INFORMACIÓN	23
5.3.1 Fuentes Primarias	23
5.3.2 Fuentes Secundarias	23
6. RESULTADOS	24
7. CONCLUSIONES	42
8. RECOMENDACIONES	43
CIBERGRAFÍA	44
ANEXOS	45

LISTA DE FIGURAS

	Pág.
Figura 1. Esquema y componentes de un sistema RFID	18
Figura 2. Funcionamiento general del sistema RFID	19
Figura 3. Estructura interna de la tarjeta RFID utilizada	25
Figura 4. Propiedades del puerto COM del SL500L	26
Figura 5. Organización de la memoria EEPROM de la MIFARE1K	27
Figura 6. Panel de lectura y escritura de la Mifare 1K	29
Figura 7. Sección "Operation windows"	30
Figura 8. Ingreso de datos	30
Figura 9. Ventana prototipo del Lector/Escritor	31
Figura 10. DLLs utilizados	32
Figura 11. Base de datos en Microsoft Acces 2007	34
Figura 12. Vista de base de datos con DataGrindView	35
Figura 13. Menú principal del programa	36
Figura 14. Administrador de base de datos	37
Figura 15. Ventana del Lector/Escritor	38
Figura 16. Ventana de verificación de datos	39
Figura 17. Ventana de Login	40
Figura 18. Diagrama de flujo de la interfaz	41

LISTA DE TABLAS

	Pág.
Tabla 1. Frecuencias para sistemas RFID	21
Tabla 2. Características mecánicas y eléctricas del módulo SL500L	24
Tabla 3. Características mecánicas de la etiqueta MIFARE 1K	25

LISTA DE ANEXOS

	Pág.
Anexo A. Código de la ventana menú	45
Anexo B. Código de la ventana administrar	46
Anexo C. Código de la ventana Lector/Escritor	54
Anexo D. Código de la ventana verificar	64
Anexo E. Código de enlace con DLL	72
Anexo F. Código de la ventana Login	74
Anexo G. Manual de usuario	76

GLOSARIO

ADMINISTRADOR: persona que gestiona los recursos informáticos, el hardware y los periféricos.

APLICACIÓN: conjunto de programas que resuelven las necesidades de una persona, una institución o una compañía.

DLL: es el término con el que se refiere a los archivos con código ejecutable que se cargan bajo demanda de un programa por parte del sistema operativo; esta denominación es exclusiva a los sistemas operativos Windows siendo .dll la extensión con la que se identifican estos ficheros, aunque el concepto existe en prácticamente todos los sistemas operativos modernos.

CONTROLADOR: es un programa que permite al sistema operativo interactuar con un periférico, haciendo una abstracción del hardware y proporcionando una interfaz, posiblemente estandarizada para usarlo; se puede esquematizar como un manual de instrucciones que le indica cómo debe controlar y comunicarse con un dispositivo en particular.

FOTOLITOGRAFÍA: es un proceso empleado en la fabricación de dispositivos semiconductores o circuitos integrados; el proceso consiste en transferir un patrón desde una fotomáscara (denominada retícula) a la superficie de una oblea; el silicio en forma cristalina se procesa en la industria en forma de las ya mencionadas obleas.

ODBC: (Open Database Connectivity) interfaz desarrollada en lenguaje de programación que le permite a las aplicaciones acceder a información desde una gran variedad de sistemas de administración de bases de datos DBMS (Data Base Management Systems).

READER: dispositivo con la capacidad de leer o de escribir en el transponder.

REGISTRO: conjunto de campos que contienen información perteneciente a una tabla relacionada a una base de datos.

RFID: identificación por radio frecuencia.

SERIGRAFÍA: es una técnica de impresión empleada en el método de reproducción de documentos e imágenes sobre cualquier material y consiste en transferir una tinta a través de una malla tensada en un marco, el paso de ésta se bloquea en las áreas donde no habrá imagen mediante una emulsión o barniz, quedando libre la zona donde pasara la tinta.

SIDECAR: es un vehículo de una rueda enganchado al costado (normalmente derecho) de una motocicleta, dando como resultado un vehículo de motor de tres ruedas y con capacidad de transportar una y en algunos casos dos personas más; la rueda del sidecar se ubica un poco más adelante de la rueda trasera de la motocicleta, diferenciándose así del triciclo.

TRANSPONDER: dispositivo en el cual se encuentran almacenados los datos de identificación de un elemento en cualquier sistema RFID; normalmente consiste en un elemento de acople de radiofrecuencia (bobina o antena) y un microchip.

RESUMEN

Viendo la necesidad de simplificar la recepción de documentación en la portería principal de la Institución Universitaria Pascual Bravo, se creó un prototipo de verificación de datos que consta de una parte de hardware y una parte de software; la parte de hardware consiste en un módulo RFID producido por la empresa StrongLink, encargado de escribir y leer el código del usuario en las tarjeta o tags de radiofrecuencia, aquí entra la parte de software, la cual gracias a una interfaz sencilla y versátil permite al usuario final hacer de la escritura y la lectura de estas tarjetas un proceso realmente sencillo.

La tarjeta RFID contiene el código del usuario y gracias a él se tiene acceso a toda su información como el nombre completo, el documento de identificación, el carnet estudiantil, el automotor que posee y la respectiva placa; esta información es ingresada inmediatamente a la base de datos .accdb del sistema una vez se escribe el código en la tarjeta; es importante resaltar que el programa también permite modificar, eliminar o buscar registros de la base de datos original, todo esto gracias al apartado del programa llamado Administrar.

INTRODUCCIÓN

El auge de las tecnologías de identificación automática ha cambiado la forma en que se lleva a cabo múltiples procesos; los sistemas de control de acceso, hoy en día generalmente disponen de un módulo que actúa como cerebro y que se encarga de realizar varias tareas en cortos espacios de tiempo, dicho módulo se conoce como lector/escritor, que acompañado con su debido software es capaz de procesar información a velocidades considerables.

Con el fin de aumentar la seguridad y proteger así la integridad de las personas que intervienen en este proceso, la tecnología de estos sistemas ha ido cambiando sus características para facilitar el trabajo a las personas encargadas de la verificación de documentos.

Debido a que en la Institución no existen tales dispositivos, se pretende crear un prototipo de verificación de datos con tarjeta RFID que satisfaga esta necesidad, la cual surge en la portería vehicular y de este modo contribuir a la movilidad de la comunidad pascualina.

1. EL PROBLEMA

La Institución Universitaria Pascual Bravo (IUPB) ubicada en la ciudad de Medellín posee una vocación hacia el saber técnico y científico, y tiene como propósito cubrir las necesidades industriales del país; a partir de la calidad en la formación de los estudiantes, ésta se ha ganado una alta reputación en los diferentes sectores de la industria.

La entrada principal de IUPB Robledo está vigilada por cuatro porteros los cuales al momento de acercarse al vehículo, sea motocicleta o automóvil verifican los papeles pertinentes: carnet estudiantil y matrícula del automotor; si los papeles están en orden y sin alteraciones los porteros permiten el ingreso o salida.

El Pascual Bravo presta sus servicios en horarios múltiples y gran parte de la comunidad cuenta con vehículo, lo que genera una congestión a la hora de ingresar y salir; cuando esto sucede la comodidad del conductor se ve implicada y los porteros, en algunos casos, para suprimir este tipo de embotellamiento optan por hacer circular los vehículos sin revisar los documentos necesarios, permitiendo el ingreso a cualquier individuo.

Debido a lo expresado anteriormente, muchas universidades, empresas, industrias y locales están implementando tecnología avanzada de verificación ID para mejorar esta situación, ya sea por código de barras o identificación por radiofrecuencia, siendo esta última la más rentable.

1.1 FORMULACION DEL PROBLEMA

¿Con la implementación de este sistema de control se mejorará el acceso de vehículos?

2. JUSTIFICACIÓN

Este trabajo tiene como finalidad realizar un modelo de sistema de control que consiste en un software dedicado a administrar la base de datos de los usuarios, un escritor y un lector de tarjetas RFID; el cual pueda ser implementado en la portería de la Institución Universitaria Pascual Bravo, para disminuir los constantes problemas de congestión vehicular que se presentan en el momento de realizar la revisión de documentos.

Gracias al diseño de este prototipo se ofrece a los conductores gran comodidad a la hora de identificarse en la portería, ya que sus papeles se verán reducidos a un solo documento que contiene los datos personales del usuario y los de su vehículo; además de esto el orden relacionado con la verificación de papeles podría aumentar a medida que el número de vehículos incremente, lo cual se ve reflejado en un ahorro de tiempo.

3. OBJETIVOS

3.1 OBJETIVO GENERAL

Implementar un prototipo de verificación de datos con tarjeta RFID a la entrada de vehículos del IUPB.

3.2 OBJETIVOS ESPECIFICOS

Implementar en una tarjeta RFID la información del usuario y de su automotor.

Establecer un escritor que imprima el código en las tarjetas RFID.

Implementar un lector de etiquetas RFID.

Diseñar un software encargado de administrar la base de datos del sistema.

Realizar el acople entre el hardware y el software del prototipo.

4. REFERENTES TEÓRICOS

4.1 RFID

(Siglas de Radio Frequency Identification, en español identificación por radiofrecuencia), es un sistema de almacenamiento y recuperación de datos remotos que usa dispositivos denominados etiquetas, tarjetas, transpondedores o tag; el propósito fundamental de esta tecnología es transmitir la identidad de un objeto similar a un número de serie único mediante ondas de radio; las tecnologías RFID se agrupan dentro de las denominadas auto ID (automatic identification, o identificación automática).

Las etiquetas son unos dispositivos pequeños, similares a una pegatina, que pueden ser adheridas o incorporadas a un producto, un animal o una persona; contienen antenas para permitir recibir y responder a peticiones por radiofrecuencia desde un emisor-receptor; las etiquetas pasivas no necesitan alimentación eléctrica interna, mientras que las activas sí lo requieren; una de las ventajas del uso de radiofrecuencia, en lugar de infrarrojos, es que no se requiere visión directa entre emisor y receptor.

El modo de funcionamiento de los sistemas RFID es el siguiente; la etiqueta, 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, el cual se encarga de analizar la información y pasarla en formato digital a la aplicación específica que utiliza el módulo; este sistema consta de los siguientes tres componentes:

Etiqueta o transpondedor: compuesta por una antena, un transductor radio y un material encapsulado o chip; el propósito de la antena es permitirle al chip, el cual contiene la información, transmitir la información de identificación de la etiqueta; existen varios tipos de etiquetas; el chip posee una memoria interna con una

capacidad que depende del modelo y varía de una decena a millares de bytes; existen varios tipos de memoria:

Solo lectura: el código de identificación que contiene es único y es personalizado durante la fabricación de la etiqueta.

De lectura y escritura: la información de identificación puede ser modificada por el lector.

Anticolisión: se trata de etiquetas especiales que permiten que un lector identifique varias al mismo tiempo, habitualmente las etiquetas deben entrar una a una en la zona de cobertura del lector.

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

Subsistema de procesamiento de datos o Middleware RFID: proporciona los medios de proceso y almacenamiento de datos.

4.2 CHIP

Es una pastilla pequeña de material semiconductor de algunos milímetros cuadrados de área, sobre la que se fabrican circuitos electrónicos generalmente mediante fotolitografía y que está protegida dentro de un encapsulado de plástico o cerámica; el encapsulado posee conductores metálicos apropiados para hacer conexión entre la pastilla y un circuito impreso; existen al menos tres tipos de circuitos integrados:

Circuitos monolíticos: están fabricados en un solo monocristal, habitualmente de silicio, pero también existen en germanio, arseniuro de galio, silicio-germanio, entre otros.

Circuitos híbridos de capa fina: son muy similares a los circuitos monolíticos, pero además, contienen componentes difíciles de fabricar con tecnología monolítica.

Circuitos híbridos de capa gruesa: se apartan bastante de los circuitos monolíticos; de hecho suelen contener circuitos monolíticos sin capsula, transistores, diodos, entre otros, sobre un sustrato dieléctrico interconectados con pistas conductoras; los resistores se depositan por serigrafía y se ajustan haciéndoles cortes con láser; todo ello se encapsula, en capsulas plásticas o metálicas, dependiendo de la disipación de energía calórica requerida; en muchos casos, la cápsula no está formada, sino que simplemente se cubre el circuito con una resina epoxi para protegerlo; en el mercado se encuentran circuitos híbridos para aplicaciones en módulos de radio frecuencia (RF), fuentes de alimentación, circuitos de encendido para automóvil, entre otros.

4.3 VEHÍCULO

Se refiere principalmente a un vehículo autopulsado por un motor propio y destinado al transporte terrestre de personas o mercancías sin necesidad de carriles.

Aunque el término automóvil es utilizado por antonomasia para referirse a los automóviles de turismo, existen otros tipos de automóviles, como:

Camión: es un vehículo motorizado para el transporte de bienes; a diferencia de los coches, que suelen tener una construcción monocasco, muchos camiones se construyen alrededor de una estructura resistente llamada chasis; la mayoría están formados por un chasis portante, generalmente un marco estructural, una cabina y una estructura para transportar la carga; hay camiones de muchos tamaños y de todo tipo desde camiones pequeños hasta los trenes de carretera, pasando por los camiones todoterreno de 200 toneladas usados en minería.

Autobús: es un vehículo diseñado para el transporte de personas; generalmente es usado en los servicios de transporte público urbano e interurbano, y con trayecto fijo; su capacidad puede variar entre 10 y 120 pasajeros.

Motocicleta: es un vehículo de dos ruedas, impulsado por un motor que acciona la rueda trasera, salvo raras excepciones; el cuadro o chasis y las ruedas constituyen la estructura fundamental del vehículo; la rueda directriz es la delantera; pueden transportar hasta dos personas, y tres si están dotadas de sidecar.

4.4 SOFTWARE

El concepto de software va más allá de los programas de computación en sus distintos estados: código fuente, binario o ejecutable, también su documentación, los datos a procesar e incluso la información de usuario forman parte del software, es decir, abarca todo lo intangible, todo lo no físico relacionado; las características más importantes son: el software se desarrolla, no se manufactura, el software no se descompone, la mayoría del software se hace a la medida y no se ensambla de componentes; los tipos de software más destacados son:

Software de sistema: su objetivo es desvincular adecuadamente al usuario y al programador de los detalles del sistema informático en particular que se use, aislándolo especialmente del procesamiento referido a las características internas de: memoria, discos, puertos y dispositivos de comunicaciones; el software de sistema le procura al usuario y programador, adecuadas interfaces de alto nivel, controladores, herramientas y utilidades de apoyo que permiten el mantenimiento del sistema global; incluye entre otros: sistemas operativos, controladores de dispositivos, herramientas de diagnóstico, herramientas de corrección y optimización, servidores.

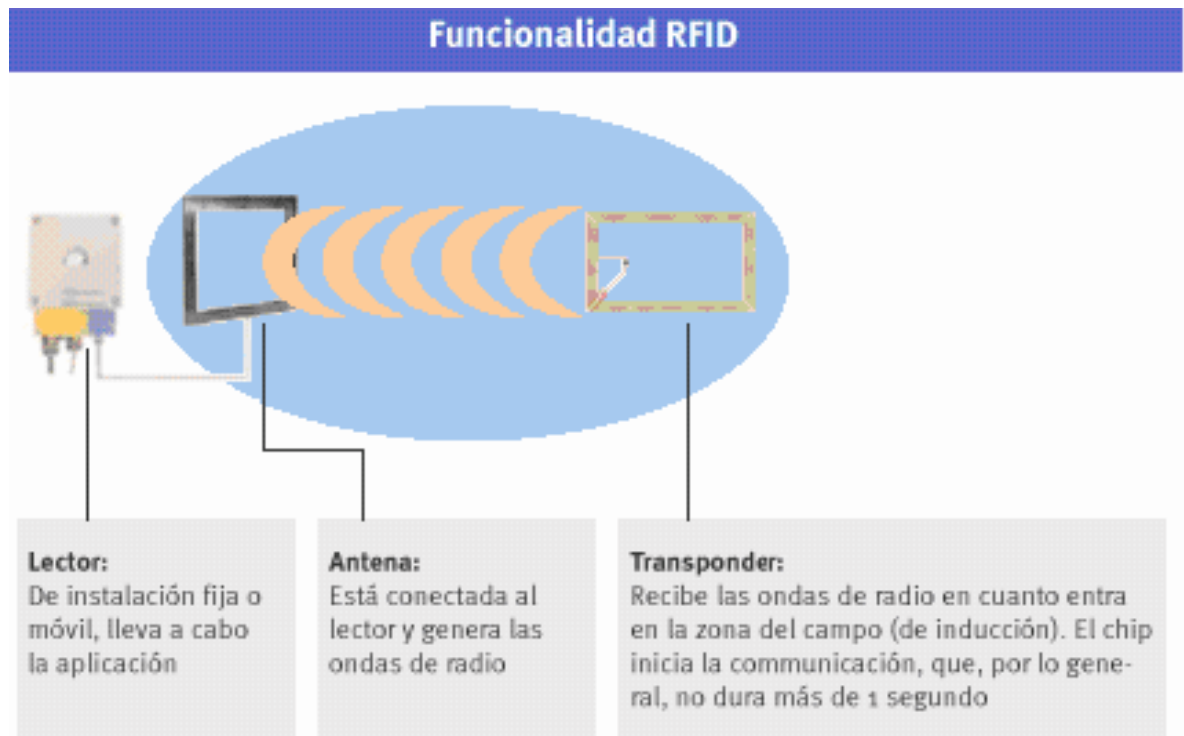
Software de programación: es el conjunto de herramientas que permiten al programador desarrollar programas informáticos, usando diferentes alternativas y lenguajes de programación de una manera práctica.

Software de aplicación: es aquel que permite a los usuarios llevar a cabo una o varias tareas específicas en cualquier campo de actividad susceptible de ser automatizado o asistido, con especial énfasis en los negocios; como el software de automatización industrial, bases de datos, videojuegos, entre otros.

4.5 MÓDULO LECTOR/ESCRITOR RFID

Un dispositivo RFID está constituido por un pequeño circuito con una antena integrada, que al recibir energía vía radio desde un emisor externo responde con una señal que indica su estado; su principal ventaja es que detecta los equipos que incorporen este sistema de manera inalámbrica y sin requerimientos de línea de vista.

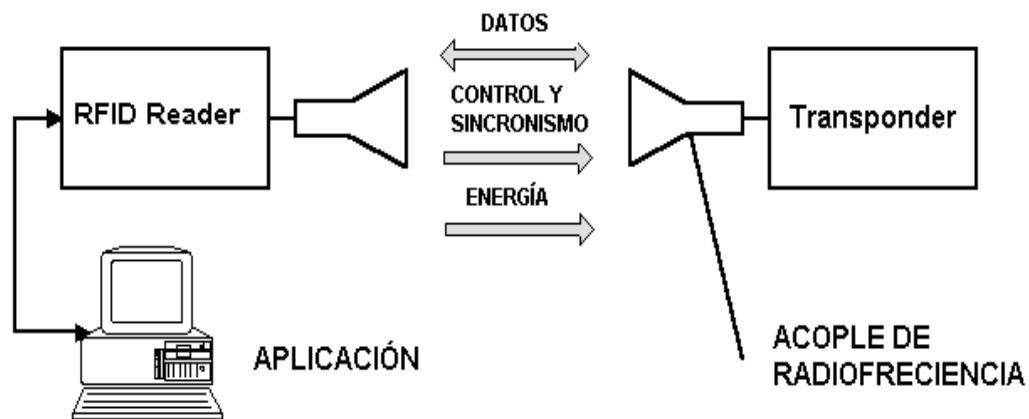
Figura 1. Esquema y componentes de un sistema RFID



Este sistema siempre es un arreglo de dos componentes: el transponder que es localizado sobre el objeto para ser identificado y el reader que dependiendo del diseño y la tecnología usada puede ser un dispositivo con la capacidad de leer y escribir sobre el transponder.

Un reader típicamente contiene un módulo a cierta frecuencia (el transmisor y el receptor), una unidad de control y un elemento que se acopla al transponder; además, algunos readers vienen con una interfaz adicional (RS232, RS485) lo que permite enviar los datos recibidos a otro sistema.

Figura 2. Funcionamiento general del sistema RFID



El transponder es el dispositivo en el cual se encuentran almacenados los datos de identificación de un elemento en cualquier sistema RFID; normalmente consiste en un elemento de acople de radiofrecuencia (bobina o antena) y un microchip; cuando el transponder, que dependiendo del caso posee o no su propio suministro de voltaje no está dentro de la distancia de respuesta del reader se comporta como un elemento totalmente pasivo; el transponder sólo es activado cuando está dentro de la distancia de respuesta de un reader y la potencia requerida para su activación es suministrada por la unidad de acople de radiofrecuencia así como también suministra las señales de control y sincronismo, y los datos como tal; por otro lado, el reader genera un campo de radiofrecuencia, normalmente conmutando (conectando y desconectando) una bobina a alta frecuencia; las frecuencias usuales van desde 125 kHz hasta la banda ISM (Industrial Scientific and Medical) de 2,4 GHz e incluso más.

El campo de radiofrecuencia genera una corriente eléctrica sobre la bobina de recepción del dispositivo, esta señal es rectificadora y de esta manera se alimenta el circuito; cuando esta llega a ser suficiente, el circuito transmite sus datos, el reader, entonces detecta los datos transmitidos por la tarjeta como una perturbación del propio nivel de la señal.

Para el caso de los transponders pasivos la señal recibida por el reader desde la tarjeta debe estar a un nivel mínimo de -60dB por debajo de la portadora de transmisión para ser leída efectivamente, normalmente el intervalo de lectura para la mayoría de los casos está entre los 30 y 60 centímetros de distancia entre reader y tarjeta.

Los sistemas RFID que generan e irradian ondas electromagnéticas son clasificados como sistemas de radio; la función de otros servicios de radio en ningún caso debe ser interrumpida o perjudicada por la operación de estos sistemas; es importante asegurar que los sistemas de identificación por radiofrecuencia no interfieran con la radio cercana, servicios de radio y televisión, móviles (la policía, servicios de seguridad valor, industria), servicios de radio marítimos o aeronáuticos.

La necesidad de ejercer el cuidado con respeto a otros servicios de radio restringe considerablemente la gama de frecuencias convenientes de operaciones disponibles a un sistema RFID; por esta razón, solo es posible usar intervalos de frecuencia reservados específicamente para aplicaciones industriales, científicas o médicas o para dispositivos de corto alcance; estas son frecuencias clasificadas mundialmente como ISM (Industrial-Scientific-Medical) o SRD (Short range devices).

Tabla 1. Frecuencias para sistemas RFID

Frecuencias para sistemas RFID		
Frecuencia	Comentario	Potencia de transmisión/ Intensidad de campo permitida
<135 kHz	Baja frecuencia, acople inductivo	72 dB μ A/m ⁴
6,765 ... 6,795 MHz	Frecuencia media (ISM), acople inductivo	42 dB μ A/m
7,400 ... 8,800 MHz	Frecuencia media, usada para solo para EAS (electronic article surveillance) que se refiere a vigilancia de artículos electrónicos	9 dB μ A/m
13,553..13,567 MHz	Frecuencia media (13,56 MHz, ISM), acople inductivo, espectro ampliado usado para gestión de ítems así como en tarjetas y etiquetas inteligentes	42 dB μ A/m
26,957..27,283 MHz	Frecuencia media (ISM), acople inductivo, solo para aplicaciones especiales.	42 dB μ A/m
433 MHz	UHF (ISM), Raramente usada para RFID	10 ... 100 mW
868 ... 870 MHz	UHF (SRD), Nueva frecuencia, sistemas en desarrollo.	500 mW, Europa

902 ... 928 MHz	UHF (SRD), Diversos sistemas.	4W –Espectro ensanchado, USA/Canadá
2,400 ... 2,483 GHz	SHF (ISM), Identificación de vehículos.	4W– Espectro ensanchado, USA/Canadá 500 mW, Europa
5,725 ... 5,875 GHz	SHF (ISM), Raramente usada para RFID	4W USA/Canadá, 500 mW Europa

5. METODOLOGÍA

5.1 TIPO DE PROYECTO

El tipo de proyecto es una implementación, debido a que este sistema ya está en funcionamiento en diversas empresas e instituciones.

5.2 MÉTODO

Este proyecto se elabora con un método deductivo, ya que parte de unas bases teóricas establecidas con el objeto de dar solución a un problema.

5.3 TECNICAS DE RECOLECCION DE INFORMACIÓN

5.3.1 Fuentes primarias. Este trabajo se realizó con el fin de disminuir los problemas de congestión vehicular que se presentan en el momento de la revisión de documentos en la entrada principal del IUPB.

5.3.2 Fuentes secundarias. Para llevar a cabo este proyecto se procede a recolectar la información necesaria por medio de fuentes como Internet, libros, revistas y manuales.

6. RESULTADOS

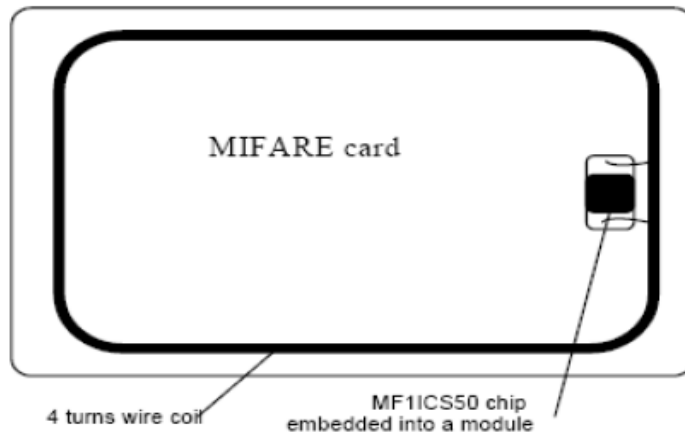
Para este proyecto se importó un módulo lector/escritor SL500L USB producido por StrongLink con las siguientes características:

Tabla 2. Características mecánicas y eléctricas del módulo SL500L

Frecuencia	13.56 MHz
Etiqueta apoyo	ISO14443A
Interfaz	USB
Rango de temperatura	-20°C - +50°C
Dimensión	110 x 80 x 26 mm
Peso	100 g
Voltaje operacional	4.5 - 5.5 VDC
Corriente de trabajo	150 mA
Sistema	Windows 98 \ 2000 \ XP \ NT \ ME \ Vista/7

Lo más destacable de este módulo es su gran compatibilidad ya que soportan las ISO14443A de tarjetas pasivas, marcas y etiquetas dando la oportunidad de escoger entre varios tipos de tags; de acuerdo a las necesidades del proyecto se escogió la etiqueta MIFARE 1K ya que por su capacidad de memoria es perfecta para guardar la información requerida y la programación no es muy compleja.

Figura 3. Estructura interna de la tarjeta RFID utilizada.



La MIFARE 1K funciona con el integrado NXP MF1ICS50 y una antena embebidos en una tarjeta de PVC, no necesita baterías y la comunicación es conforme con la normativa ISO/ICE 14443A -2-3; es apropiada para usarse en el transporte público, el sistema de peaje, el sistema de aparcamiento, entre otros.

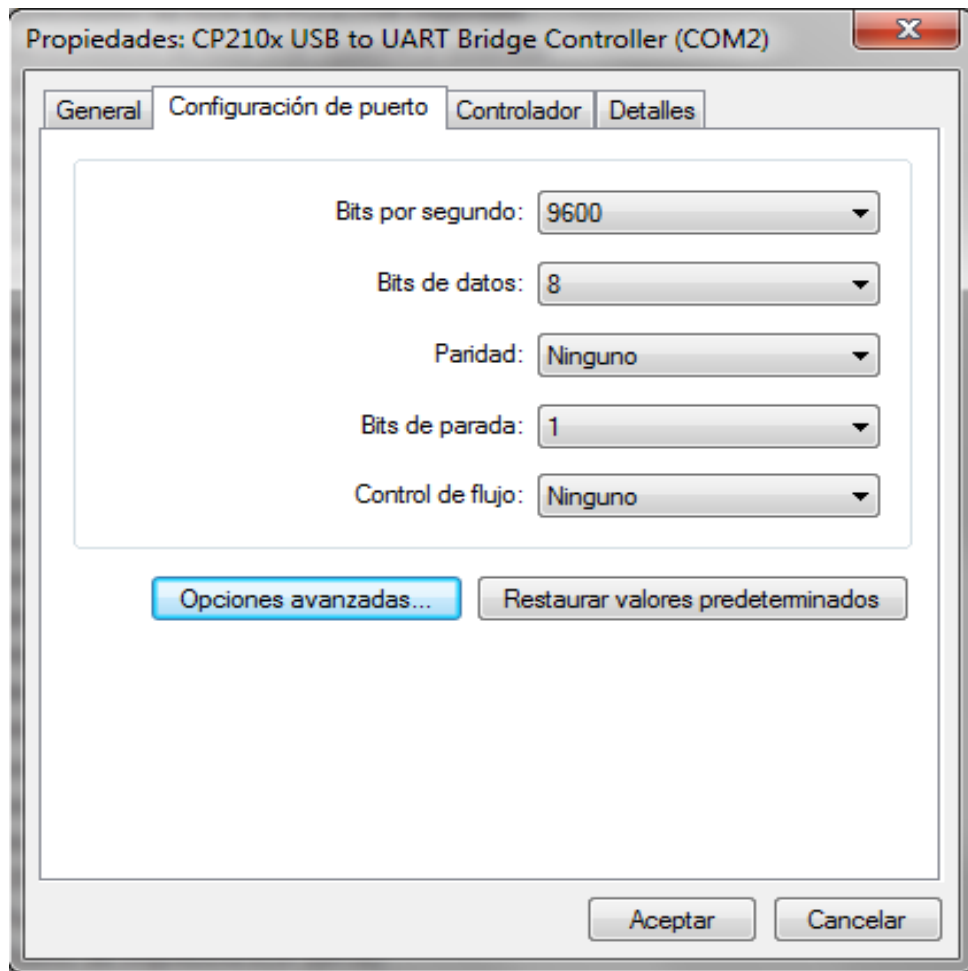
Tabla 3. Características mecánicas de la etiqueta MIFARE 1K

Frecuencia	13.56 MHz
Protocolo	ISO14443A
Tamaño EEPROM	1024 Bytes
Temperatura	-20°C - +50°C
Dimensión	85.6 × 54 × 0.86 mm

Luego se procedió a instalar el driver del módulo RFID para que el computador reconociera el dispositivo y el software ICTransfer encargado de escribir y leer la información de las etiquetas; se debe identificar en que puerto virtual COM se conecta el dispositivo ya que el programa ICTransfer lo solicita para realizar una conexión correcta, para hacerlo se debe abrir el administrador de dispositivos de Windows y se debe presionar el botón izquierdo del mouse en la opción "Puertos";

para poder programar la Mifare 1K se debió configurar este puerto COM de acuerdo a las instrucciones del fabricante.

Figura 4. Propiedades del puerto COM del SL500L



El TAG cuenta con una memoria EEPROM de 1024 X 8 bit organizada en 16 sectores con 4 bloques de 16 bytes cada uno.

Figura 5. Organización de la memoria EEPROM de la MIFARE1K

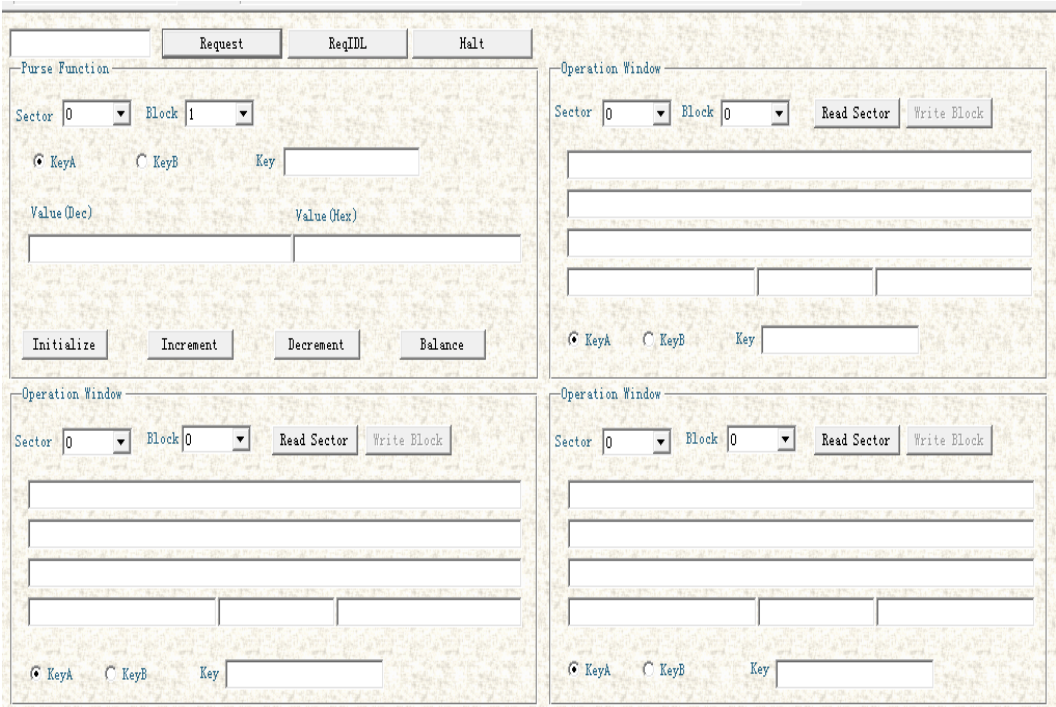
Sector	Block	Byte Number within a Block																Description
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
15	3	Key A					Access Bits				Key B							Sector Trailer 15
	2																	Data
	1																	Data
	0																	Data
14	3	Key A					Access Bits				Key B							Sector Trailer 14
	2																	Data
	1																	Data
	0																	Data
:	:																	
:	:																	
:	:																	
1	3	Key A					Access Bits				Key B							Sector Trailer 1
	2																	Data
	1																	Data
	0																	Data
0	3	Key A					Access Bits				Key B							Sector Trailer 0
	2																	Data
	1																	Data
	0																	Manufacturer Block

El bloque del fabricante contiene la información del circuito integrado, en sí, el número serial del TAG, este sector es de solo lectura; como se puede observar cada sector cuenta con 3 bloques para almacenar datos, estos bloques pueden ser configurados de acuerdo a los bits de acceso como bloques de lectura/escritura para aplicaciones de control de acceso o como bloques de valor para aplicaciones de pago electrónico en los que pueden emplearse comandos de incremento o decremento de dicho valor; también es posible observar que cada sector contiene un bloque denominado como Sector Tráiler, estos bloques contienen las claves secretas A y B para poder acceder a la memoria y las condiciones de acceso de los cuatro bloques de cada sector; esta información es muy importante teniendo en cuenta que antes de llevar a cabo cualquier operación con la memoria es necesario que la tarjeta sea seleccionada y se realice una autenticación con las claves secretas A y B descritas así como las condiciones de acceso de cada bloque.

Después de haber estudiado que tipo de clave (A o B) se usarían y que sectores de bloque de la memoria se iban a utilizar se implementó en primera medida el software ICTransfer para hacer la programación del TAG; el software permite configurar y hacer pruebas de conexión con el módulo SL500L antes de hacer cualquier procedimiento de lectura o escritura, para ello se selecciona la opción SET(P) en el programa y se escoge la opción Port set, a continuación se debe escoger el número de puerto COM y el número de Bits por segundo el cual debe ser de 9600 B/s ya que con un valor mayor puede haber pérdida de información; también se pueden apreciar opciones como el cambiar el color del led indicativo del módulo, ver que etiquetas soporta, apagar la antena del módulo, hacer un ping para ver si el dispositivo está conectado, obtener la información o número de serie del módulo y cambiar la duración del sonido que hace al ser conectado.

Luego de hacer las pruebas necesarias con el módulo se procede a escoger que tipo de tarjeta se va a utilizar, para ello se da clic derecho en la opción Choose Card (I), ya que la Mifare 1K utiliza el protocolo ISO14443A por obligación se debe seleccionar esa opción o de lo contrario se accederían a sectores de manufactura no autorizados provocando la pérdida del serial de la tarjeta; a continuación y como paso final aparece la ventana de lectura y escritura.

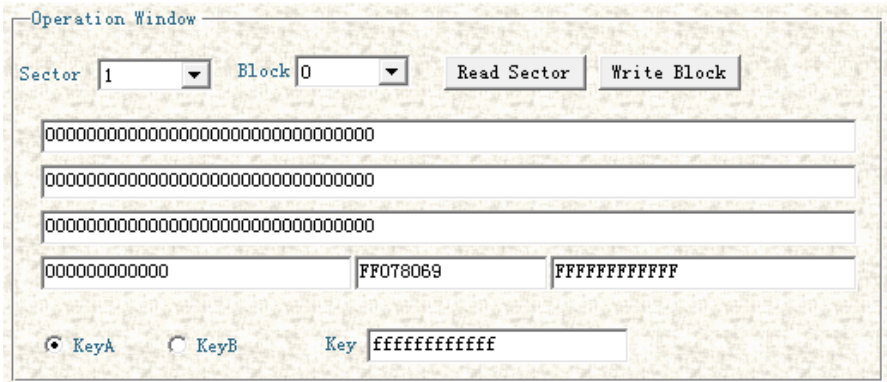
Figura 6. Panel de lectura y escritura de la Mifare 1K



Aquí solo se trabajó con la sección "Operation window", ya que el apartado "Purse function" es solo para procesos que tengan que ver con dinero o valores cuantificables como por ejemplo la tarjetas de transporte público a las cuales se les recarga o resta dinero por cada uso haciendo uso de los botones Increment y Decrement pero al igual que el apartado "Operation windows" se debe trabajar con una clave de autenticación A o B, por otro lado está el botón Request con el cual obtenemos el serial de la tarjeta.

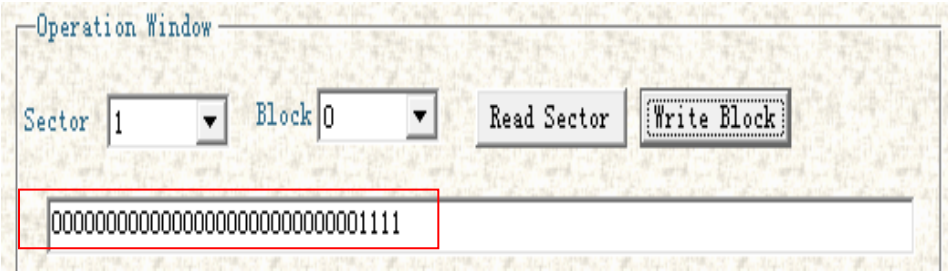
Lo primero que se debe hacer es asignar la clave A o B para poder acceder a la información de la tarjeta, la clave A se usa para poder realizar operaciones de lectura/escritura mientras que la clave B se usa para operaciones de solo lectura, para el caso de la tarjeta Mifare 1K la clave A es FFFFFFFF; a continuación se selecciona el sector y el bloque en los cuales se hace la lectura o la escritura, para este caso se escogió el sector 1/bloque 0 y se le da clic derecho al botón "Read sector".

Figura 7. Sección "Operation windows"



Se puede observar que aparece la respectiva clave, el tipo de tarjeta (FF078069) y los valores que tiene el bloque en ese momento; ahora para poner el valor que se necesita solo hay que cambiar los últimos dígitos del valor actual en la primera fila ya que lo que se desea es asignar a cada usuario un código de 4 dígitos, quedando el valor del bloque de la siguiente manera.

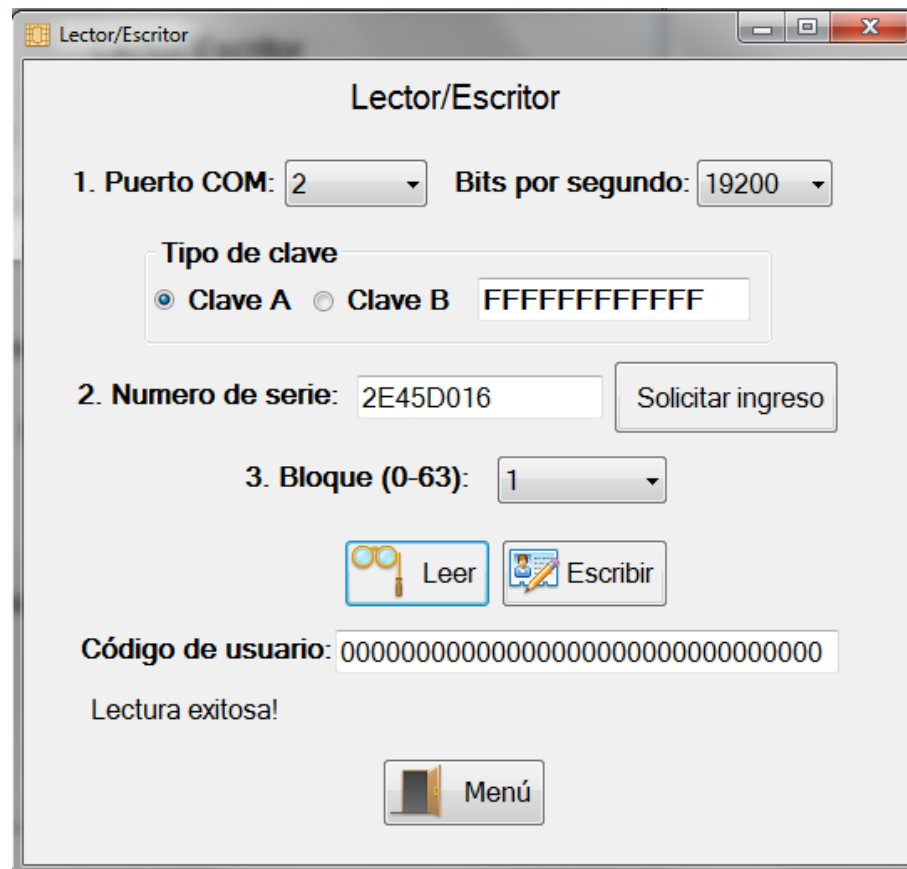
Figura 8. Ingreso de datos



Ya solo basta con dar clic izquierdo en "Write Block" y la escritura será exitosa; si en vez de haber asignado una clave A se hubiera asignado una clave B, el botón "Write Block" estaría desactivado ya que esa es una clave de solo lectura.

Viendo que esta opción de escritura de la tarjeta RFID era un poco complicada para el usuario final, se optó por desarrollar un software en lenguaje Visual Basic con una interfaz más sencilla (el código se encuentra en el apartado de anexos).

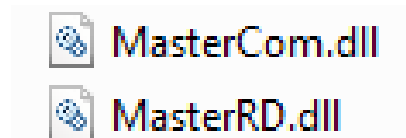
Figura 9. Ventana prototipo del Lector/Escritor



Este programa cumple con las mismas funciones de escritura del ICTransfer solo que es más sencilla su manipulación; en primera instancia se selecciona el puerto COM en donde está conectado el dispositivo del menú desplegable, luego se da clic derecho en solicitar ingreso, a continuación se selecciona el bloque a escribir y para finalizar se da clic izquierdo en leer, cuando muestre los datos de la tarjeta que en ese momento estarán en ceros podremos modificar sus dígitos los cuales serán el código del usuario; cabe resaltar que ya no hay que escoger el tipo de clave ni tampoco el número de bits por segundo ya que el programa hace el reconocimiento

automático; para poder que el programa funcionara bien y reconociera el dispositivo se trabajó con bibliotecas de enlace dinámico o DLL, estas bibliotecas se extraen del mismo sistema operativo Windows ya que contienen todos los ficheros para que las conexiones de puerto COM y su comunicación de datos sea correcta, estos ficheros fueron enlazados con el programa de Visual Basic para ser manipulados por series de instrucciones (el código de enlace se encuentra en el apartado de anexos).

Figura 10. DLLs utilizados.



En resumen, la memoria de 1024 X 8 bits del chip de la tarjeta RFID se organiza en 16 sectores con 4 bloques de 16 Bytes cada uno y para el acceso a cada uno de los sectores se necesita de una secuencia de autenticación en la cual el módulo lector/escritor especifica el sector a ser accedido y escoge la clave de ingreso (que puede ser la A o la B); luego hay una verificación de información entre la tarjeta y el dispositivo lector y finalmente se recibe una respuesta por parte del chip de la tarjeta en la cual se autoriza o no a realizar modificaciones en el valor de sus bloques.

Para crear la parte que relaciona el sistema con el usuario se utilizaron varios métodos de programación pero sin obtener los resultados deseados, uno de ellos fue por Macros utilizando Excel pero al carecer este de funciones avanzadas no se siguió implementando, al final se decidió trabajar con el último software de desarrollo Microsoft Visual Studio 2012, el cual utiliza un lenguaje de programación basado en objetos y métodos, entre otras características.

Este es un lenguaje de programación visual, también llamado lenguaje de cuarta generación; esto quiere decir que un gran número de tareas se llevan a cabo sin escribir código, simplemente con operaciones gráficas realizadas con el mouse, sobre la pantalla.

Para comunicar el computador con el módulo RFID, de manera que el usuario pueda manipularlo a través de una aplicación de Windows, se utilizó un puerto USB y se instaló el controlador del dispositivo (contenido en el CD) para que sea reconocido por el equipo, aunque este módulo cuenta con tecnología Plug-and-play que le permite ser conectado a una computadora sin tener que configurarlo.

Cuando los datos son adquiridos por Visual Basic desde el módulo RFID, estos se utilizan para compararlos con la base de datos, donde se mantiene el registro de los usuarios.

Esta base de datos se creó con el programa Microsoft Access y obligatoriamente debe ser instalada en el equipo y no se debe mover de la ruta en la que se encuentre, por lo que se requirió realizar un enlace o vínculo entre el programa y la base de datos mediante un código desarrollado en Visual Basic; se debe resaltar que por causas de seguridad, el ingreso a la base de datos mediante Access 2007/2010 solo puede hacerse por medio de un usuario junto con una contraseña, ya que en casos de manipulación imprudente se provocarían fallas considerables en el software de administración.

Figura 11. Base de datos en Microsoft Acces 2007.



	Codigo	Nombre	Tipo	Documento	Carnet	Vehiculo	Placa
user: Tabla	xxxxxx	Bryan Gomez	CC	xxxxx	xxxxxxxx	xxxxxx	xxxxxxx
*							

El software desarrollado para la aplicación realiza todas sus transacciones a través de una interfaz gráfica; es por esto que se hace necesario manejar las bases de datos por medio de código y la comunicación entre Visual Basic y una base de datos .accdb (formato de Access 2007) se hace utilizando objetos ADO (Activate Data Object).

ADO contiene la colección de objetos para crear una conexión a base de datos y leer datos desde tablas, trabajando como una interfaz hacia la fuente de datos; sin embargo, no se comunica directamente con la base de datos, sino que accede a ella a través de una interfaz intermediaria, llamada OLE DB (OLE Data Base).

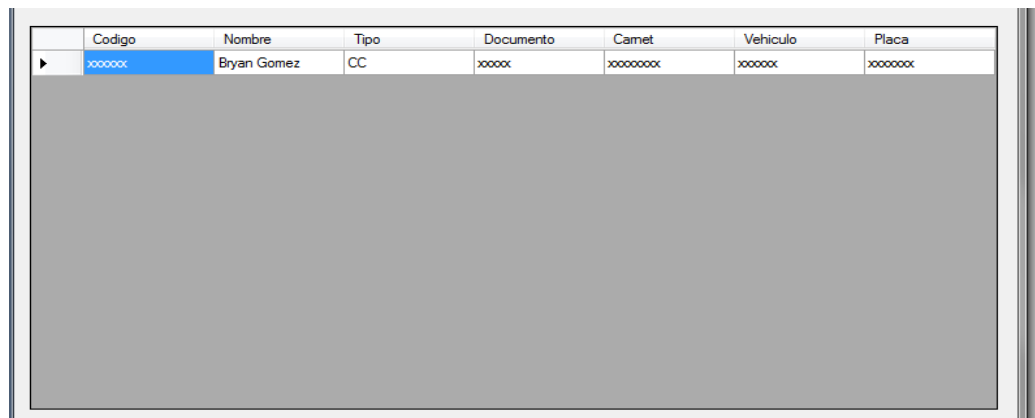
En general, después de crear una conexión a la base de datos, se puede ignorar la existencia de OLE DB, debido a que este driver hace todo su trabajo en background; existen dos maneras para que el proveedor OLE DB brinde acceso a una base de datos: directamente, en la cual se accede mediante un driver ODBC (Open Data Base Connectivity) o indirectamente, modo en el que accede mediante un driver OLE DB nativo.

Para este enlace con la base de datos "usuarios.accdb" se utilizó la siguiente línea de código:

```
Dim CN As New OleDb.OleDbConnection ("PROVIDER=MICROSOFT.ACE.OLEDB.12.0;DATASOURCE=C:\Users\FAMIGOM\Desktop\usuarios.accdb")
```

En donde se nombró la una variable CN como conexión para usar los comandos .OPEN/.CLOSE y se especificó el proveedor de la base de datos con su respectiva ruta en el equipo; posteriormente se crea una rutina en la que la base de datos se muestre en el programa sin necesidad de abrir Acces por medio del comando DATAGRINDVIEW.

Figura 12. Vista de base de datos con DataGrindView.



	Codigo	Nombre	Tipo	Documento	Carnet	Vehiculo	Placa
▶	xxxxxx	Bryan Gomez	CC	xxxxxx	xxxxxxxx	xxxxxx	xxxxxxxx

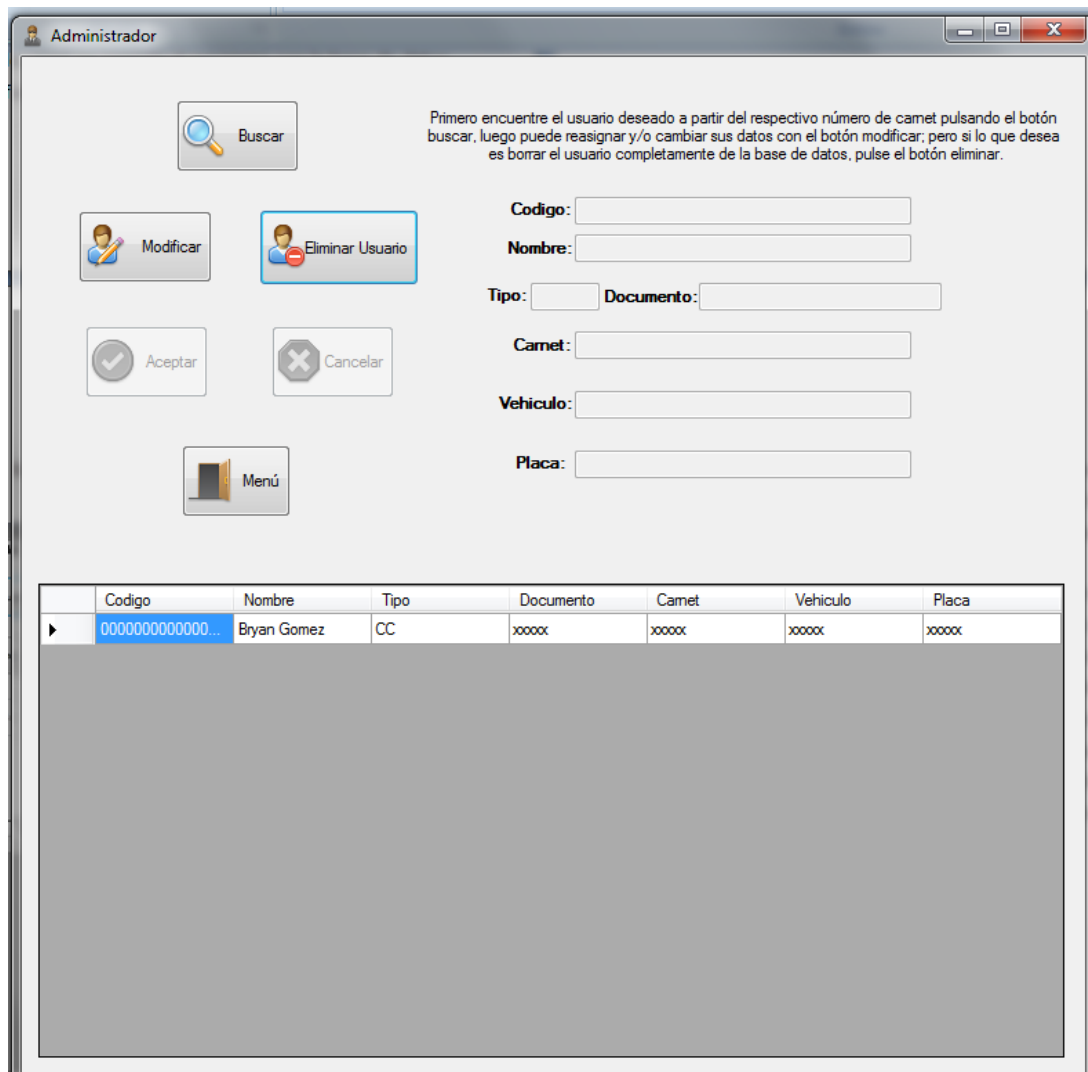
Ahora que ya se ha explicado cómo se conecta el programa con la base de datos se procederá a explicar las funciones y parte del código empezando por el menú.

Figura 13. Menú principal del programa.



El menú es la parte fundamental del programa ya que es la que hace el vínculo a todas las opciones y es la interfaz principal conteniendo botones como "administrar" que lleva directamente a la ventana "Administrador" en donde se puede buscar, eliminar y modificar un usuario; dotado con una interfaz sencilla y una visualización de la base de datos en tiempo real, también se han agregado comentarios para guiar al administrador en el uso del programa.

Figura 14. Administrador de base de datos



El botón "Lector/Escritor" lleva a la ventana donde se hace la escritura y lectura de la tarjeta RFID, al principio este programa debió adquirirse ya que se creía que el módulo SL500L solo funcionaba con él, siendo totalmente exclusivo pero sin tener que pagar por una licencia (Freeware) pero al final se optó por crear un programa que gracias a las bibliotecas de enlace dinámico o DLL cumple con las mismas funciones del ICTransfer pero con una interfaz mucho más simplificada; cuando se termine de llenar los datos solicitados se da click derecho en el botón escribir y la tarjeta RFID quedara con su respectivo código e igualmente el usuario quedara

registrado en la base de datos del sistema; el código de usuario es la combinación de ceros con el número de documento por ejemplo 000000000000000000000001040739166.

Figura 15. Ventana del Lector/Escritor.

Lector/Escritor

Coloque la tarjeta sobre el lector, seleccione el puerto COM donde esta conectado el modulo, luego de click izquierdo en el botón Solicitar ingreso, despues elija el bloque 1 y de click izquierdo en el botón Leer, para finalizar ingrese la información solicitada en el panel de datos. Para asignar el codigo remplace los ultimos valores de 0s por el numero de documento ya sea CC o TI, dejando los otros valores intactos.

1. Puerto COM: 1

2. Numero de serie: [text box] Solicitar ingreso

3. Bloque (0-63): 4

Leer Escribir Cancelar

Menú

Código de usuario: [text box]

Nombre: [text box]

Tipo: [text box] Documento: [text box]

Carnet: [text box]

Vehiculo: [text box]

Placa: [text box]

El botón "verificar" lleva a una ventana donde gracias a la información proporcionada por el módulo lector RFID se hacen las comparaciones con la base de datos en Access para verificar el código del usuario y así poder mostrar su respectiva información.

Figura 16. Ventana de verificación de datos.

Verificación de datos

Coloque la tarjeta sobre el lector, seleccione el puerto COM donde esta conectado el modulo, luego de click derecho en el boton Solicitar ingreso, despues elija el bloque 1 y para finalizar de click izquierdo en el botón Leer.

1. Puerto COM: 1

2. Numero de serie: Solicitar ingreso

3. Bloque (0-63): 4

Leer Menú

Código de usuario:

Nombre:

Tipo: Documento:

Carnet:

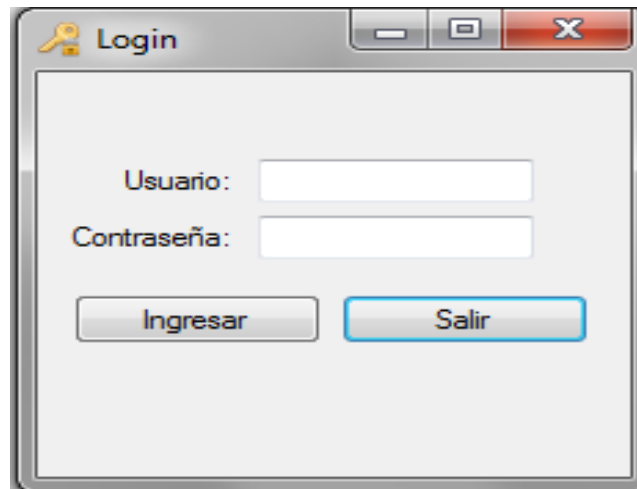
Vehiculo:

Placa:

El botón "ver archivo .accdb" abre el archivo fuente usuarios.accdb por si se desean hacer modificaciones en la tabla, ya sea cambiar nombre de columnas, poner un número límite de dígitos o letras, cambiar nombre de ítems y muchas otras modificaciones; se recomienda hacer estas modificaciones solo si se está familiarizado con la programación del software ya que al hacer cambios en la tabla se deberá hacer también la modificación en el código para evitar que el programa quede obsoleto; para hacer uso de este botón es necesario contar con el programa Microsoft Acces 2007 o superior.

También se hizo el acceso al programa por medio de un usuario y una contraseña para mejorar la seguridad y por último se empaqueto todo el programa en un archivo instalador para que pudiera ser utilizado en cualquier computador.

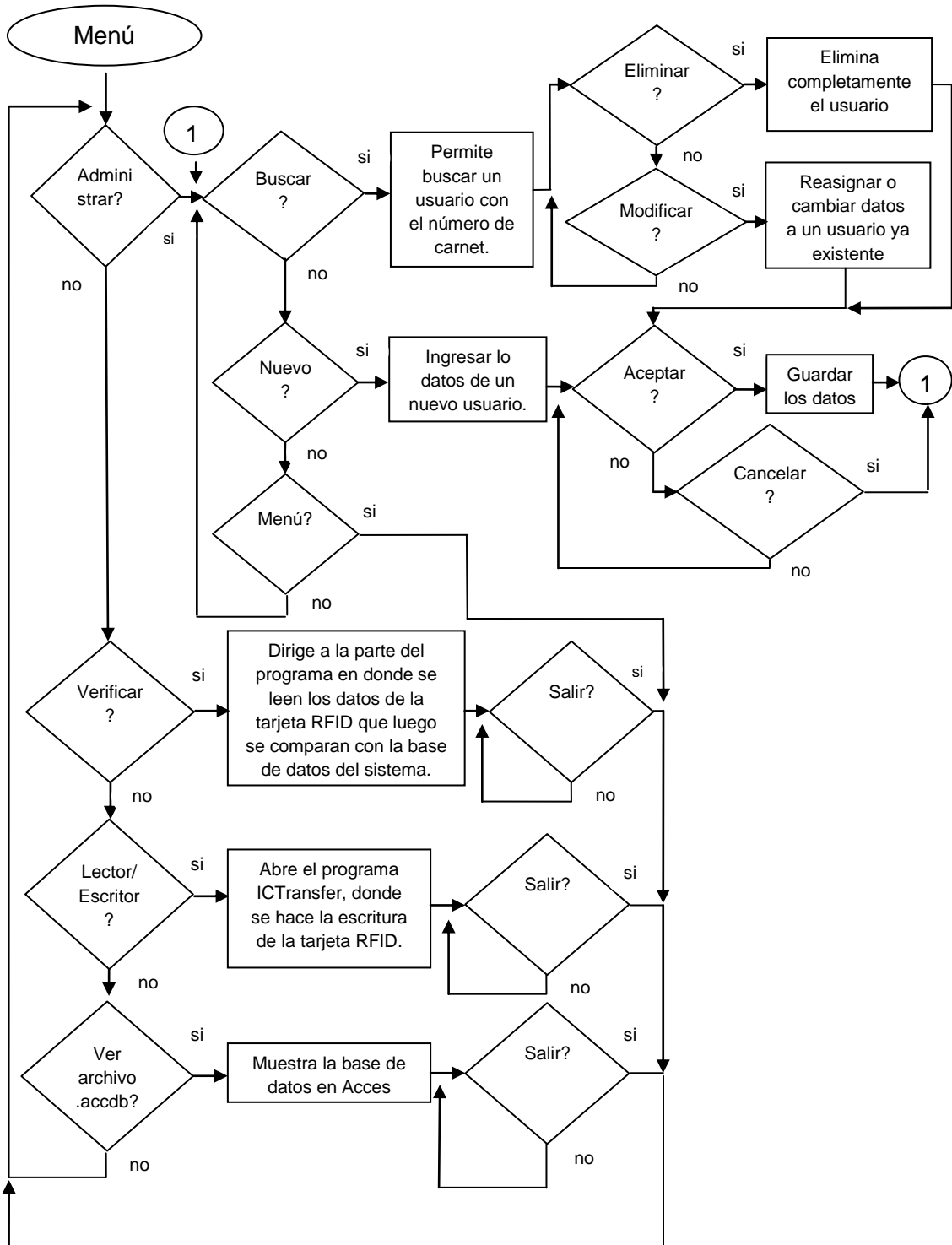
Figura 17. Ventana de Login.



The image shows a standard Windows-style login dialog box. The title bar at the top contains a yellow key icon, the text "Login", and three window control buttons (minimize, maximize, close). The main area of the dialog is light gray and contains two text input fields. The first field is labeled "Usuario:" and the second is labeled "Contraseña:". Below these fields are two buttons: "Ingresar" (Login) and "Salir" (Exit). The "Ingresar" button is gray, and the "Salir" button is light blue.

El usuario es "pascualbravo" y la contraseña es "electronica".

Figura 18. Diagrama de flujo de la interfaz



7. CONCLUSIONES

Se implementó correctamente la tarjeta RFID y el módulo SL500L producido por StrongLink para escribir, leer, modificar y borrar información de los usuarios registrados en la base de datos; las etiquetas RFID se ven como una alternativa que reemplazará los códigos de barras, puesto que tienen un número de ventajas importantes sobre esta.

El acople entre hardware y software del prototipo se hizo correctamente gracias a la utilización de librerías DLL, controladas por un programa creado en Visual Basic; las aplicaciones desarrolladas en lenguajes de programación orientadas a objetos en unión con la utilización adecuada de bases de datos, es muy útil en el desarrollo de sistemas de control de activos, control de acceso de personas, entre muchas más aplicaciones.

Es necesario destacar que el desarrollo del sistema total (integrando la parte de RFID y el desarrollo de la aplicación), provee herramientas que permiten un pequeño acercamiento al desarrollo de tecnologías que impulsen al país hacia la generación de propuestas en lugar de solo adquirirlas; teniendo en cuenta lo anterior, es de suma importancia resaltar que el desarrollo de este trabajo de grado permitió el acercamiento a una nueva tecnología y el conocimiento de sus características.

8. RECOMENDACIONES

En cuanto a la base de datos se puede agregar la foto del usuario para un mejor reconocimiento de la persona que porta la tarjeta, también por medio de código se puede agregar la fecha y la hora en la que se hacen los cambios a los registros.

La tarjeta RFID puede ser utilizada para imprimir el respectivo carnet estudiantil, ya que está hecha del mismo material utilizado por la institución, así la información estaría siempre presente en la tarjeta en caso de que se deteriore la impresión.

Si lo que se desea es identificar al usuario a una mayor distancia se puede modificar el circuito del módulo SL500L cambiando el juego de resistencias y condensadores que forman el circuito.

Se debe recordar a que puerto COM se está conectando el módulo RFID ya que en el programa se debe especificar y trabajar solo con tarjetas MIFARE 1K.

A la base de datos creada en Acces no se le puede hacer ninguna modificación, ya que podría comprometer el funcionamiento del programa.

CIBERGRAFÍA

RFID, <http://es.wikipedia.org/wiki/RFID>, citado 19 de Febrero de 2013

CHIP, http://es.wikipedia.org/wiki/Circuito_integrado, citado 14 de febrero de 2013

VEHÍCULO, <http://es.wikipedia.org/wiki/Veh%C3%ADculo>, citado 15 de febrero de 2013

SOFTWARE, <https://es.wikipedia.org/wiki/Software>, citado 15 de febrero de 2013

AYUDA, SOPORTE Y COMANDOS DE MICROSOFT VISUAL STUDIO, <http://msdn.microsoft.com/es-es/vstudio/default.aspx>, citado 20 de febrero de 2013

ADO, [http://msdn.microsoft.com/es-es/library/e80y5yhx\(v=vs.80\).aspx](http://msdn.microsoft.com/es-es/library/e80y5yhx(v=vs.80).aspx), citado 22 de febrero de 2013

INSTALLSHIELD, http://www.flexerasoftware.com/webdocuments/productdemos/IS2010_VSLE_Demo/IS2010_VSLE_Demo-lite.html, citado 28 de febrero de 2013

CODIGO RFID, <http://www.codigo.org.ar/Codig/RFID.asp>, citado 1 de marzo de 2013

MIFARE 1K, http://www.zero13wireless.net/foro/showthread.php?7900-Mifare-1K-STD_S50

Anexo A. Código del menú

Public Class Form2

Private Sub Button1_Click (sender As Object, e As EventArgs) Handles
Button1.Click, Permite ingresar a la ventana de administración

Hide ()

Form1.Show ()

End Sub

Private Sub Button3_Clic (sender As Object, e As EventArgs) Handles
Button3.Clic, permite ingresar a la ventana Lector/Escritor

Hide()

frm_main.Show()

End Sub

Private Sub Button4_Click(sender As Object, e As EventArgs) Handles
Button4.Click, permite abrir la base de datos original
System.Diagnostics.Process.Start("C:\Users\Familiar\Desktop\usuarios.accdb")

End Sub

Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click ,permite ingresar a la ventana verificar

Hide()

frm_main2.Show()

End Sub

Private Sub Button5_Click (sender As Object, e As EventArgs) Handles
Button5.Click, permite salir del programa

Close()

Form3.Close()

End Sub

Anexo B. Código de la ventana administrar

```
Public Class Form1
```

```
    Dim N As Integer = 0
```

```
    Dim CN As New
```

```
OleDb.OleDbConnection("PROVIDER=MICROSOFT.ACE.OLEDB.12.0;DATA  
SOURCE=C:\Users\Familiar\Desktop\usuarios.accdb"), Proceso para enlazar la  
base de datos con el programa
```

```
Sub MOSTRAR() , Proceso para mostrar la base de datos
```

```
    Dim DA As New OleDb.OleDbDataAdapter("SELECT * FROM [user]", CN)
```

```
    Dim DS As New DataSet
```

```
    DA.Fill(DS, "[user]")
```

```
    DataGridView1.DataSource = DS.Tables("[user]")
```

```
End Sub
```

```
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles
```

```
MyBase.Load
```

```
    Info.Text = "Primero encuentre el usuario deseado a partir del respectivo  
número de carnet pulsando el botón buscar, luego puede reasignar y/o cambiar sus  
datos con el botón modificar; pero si lo que desea es borrar el usuario  
completamente de la base de datos, pulse el botón eliminar."
```

```
    Call MOSTRAR()
```

```
End Sub
```

```
Private Sub Button8_Click (sender As Object, e As EventArgs) Handles
```

```
Button8.Click, Primer proceso para el botón "Modificar"
```

```
    If TextBox1.Text = "" Then MessageBox.Show("Usuario no especificado, por  
favor busque el usuario primero", "Alerta") : Exit Sub
```

```
    TextBox2.Enabled = True
```

```
TextBox3.Enabled = True
TextBox4.Enabled = True
TextBox5.Enabled = True
TextBox6.Enabled = True
TextBox7.Enabled = True
Button2.Enabled = True
Button10.Enabled = True
Button11.Enabled = False
Button1.Enabled = False
Button3.Enabled = False
Button8.Enabled = False
Button9.Enabled = False
N = 2
End Sub
```

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click, Primer proceso para el botón "Nuevo usuario"

LIMPIARU()

Info.Text = "Por favor no dejar espacios en blanco, para guardar guardar los
datos del nuevo usuario pulse el botón aceptar, de lo contrario pulse el botón
cancelar."

```
TextBox1.Focus()
TextBox1.Enabled = True
TextBox2.Enabled = True
TextBox3.Enabled = True
TextBox4.Enabled = True
TextBox5.Enabled = True
TextBox6.Enabled = True
TextBox7.Enabled = True
Button2.Enabled = True
Button10.Enabled = True
```



```

Button11.Enabled = False
Button1.Enabled = False
Button3.Enabled = False
Button8.Enabled = False
Button9.Enabled = False
N = 1
End Sub

```

```

Private Sub Button3_Click (sender As Object, e As EventArgs) Handles
Button3.Click, Proceso para el botón "Eliminar"

```

```

    Dim n As MsgBoxResult
    Dim CMD As New OleDb.OleDbCommand("DELETE FROM [user] WHERE
Codigo="" & TextBox1.Text & """, CN)

```

```

    If TextBox1.Text = "" Then MessageBox.Show("Usuario no especificado, por
favor busque el usuario primero", "Alerta") : Exit Sub

```

```

    n = MessageBox.Show("¿Está seguro que desea eliminar este usuario?",
"Confirmación", MessageBoxButtons.YesNo, MessageBoxIcon.Question,
MessageBoxDefaultButton.Button2)

```

```

    If n = MsgBoxResult.Yes Then
        CN.Open()
        CMD.ExecuteNonQuery()
        CN.Close()
        LIMPIARU()
        MOSTRAR()
        MessageBox.Show("Usuario eliminado satisfactoriamente")
    Else
        MessageBox.Show("Eliminación cancelada")
    End If
End Sub

```

Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click, Proceso para el botón "Aceptar"

Info.Text = "Primero encuentre el usuario deseado a partir del respectivo
número de carnet pulsando el botón buscar, luego puede reasignar y/o cambiar sus
datos con el botón modificar; pero si lo que desea es borrar el usuario
completamente de la base de datos, pulse el botón eliminar."

If TextBox1.Text = "" Then MessageBox.Show("Código no puede estar vacío",
"Alerta") : Exit Sub

If TextBox2.Text = "" Then MessageBox.Show("Nombre no puede estar vacío",
"Alerta") : Exit Sub

If TextBox3.Text = "" Then MessageBox.Show("Tipo no puede estar vacío",
"Alerta") : Exit Sub

If TextBox4.Text = "" Then MessageBox.Show("Documento no puede estar
vacio", "Alerta") : Exit Sub

If TextBox5.Text = "" Then MessageBox.Show("Carnet no puede estar vacío",
"Alerta") : Exit Sub

If TextBox6.Text = "" Then MessageBox.Show("vehículo no puede estar
vacio", "Alerta") : Exit Sub

If TextBox7.Text = "" Then MessageBox.Show("Placa no puede estar vacío",
"Alerta") : Exit Sub

If N = 1 Then

VALIDAR()

Elseif N = 2 Then

ACTUALIZARU()

End If

Button1.Enabled = True

Button3.Enabled = True

Button8.Enabled = True

Button2.Enabled = False

```
Button10.Enabled = False
Button11.Enabled = True
Button9.Enabled = True
TextBox1.Enabled = False
TextBox2.Enabled = False
TextBox3.Enabled = False
TextBox4.Enabled = False
TextBox5.Enabled = False
TextBox6.Enabled = False
TextBox7.Enabled = False
LIMPIARU()
End Sub
```

Private Sub Button10_Click (sender As Object, e As EventArgs) Handles
Button10.Click, Proceso para el botón "Cancelar"

Info.Text = "Primero encuentre el usuario deseado a partir del respectivo
número de carnet pulsando el botón buscar, luego puede reasignar y/o cambiar sus
datos con el botón modificar; pero si lo que desea es borrar el usuario
completamente de la base de datos, pulse el botón eliminar."

```
Button9.Enabled = True
Button1.Enabled = True
Button3.Enabled = True
Button8.Enabled = True
Button2.Enabled = False
Button10.Enabled = False
Button11.Enabled = True
Button9.Enabled = True
TextBox1.Enabled = False
TextBox2.Enabled = False
TextBox3.Enabled = False
TextBox4.Enabled = False
```

```
TextBox5.Enabled = False
TextBox6.Enabled = False
TextBox7.Enabled = False
LIMPIARU()
End Sub
```

```
Sub BUSCARU(), Segundo proceso para el botón "Buscar"
    Dim x As Object
    x = InputBox("Por favor ingrese el número de carnet", "Buscar")
    Dim CMD As New OleDb.OleDbCommand("SELECT * FROM [user] WHERE
Carnet=" & x & "'", CN)
    Dim DR As OleDb.OleDbDataReader

    CN.Open()
    DR = CMD.ExecuteReader

    If DR.Read Then
        TextBox1.Text = DR(0)
        TextBox2.Text = DR(1)
        TextBox3.Text = DR(2)
        TextBox4.Text = DR(3)
        TextBox5.Text = DR(4)
        TextBox6.Text = DR(5)
        TextBox7.Text = DR(6)
    Else
        MessageBox.Show("Usuario no registrado", "Alerta")
    End If
    CN.Close()
End Sub
```

Private Sub Button11_Click(sender As Object, e As EventArgs) Handles
Button11.Click, Primer proceso para el botón "Buscar"

BUSCARU()

End Sub

Sub VALIDAR(), Proceso para verificar datos iguales

Dim CMD As New OleDb.OleDbCommand("SELECT * FROM [user] WHERE
Carnet=" & TextBox5.Text & "'", CN)

Dim DR As OleDb.OleDbDataReader

CN.Open()

DR = CMD.ExecuteReader

If DR.Read Then

CN.Close()

MessageBox.Show("Este usuario ya se encuentra registrado", "Alerta") :

Exit Sub

Else

NUEVOU()

End If

End Sub

Sub NUEVOU(), Segundo proceso para el botón "Nuevo usuario"

Dim CMD As New OleDb.OleDbCommand("INSERT INTO [user] VALUES('" &
TextBox1.Text & "','" & TextBox2.Text & "','" & TextBox3.Text & "','" & TextBox4.Text
& "','" & TextBox5.Text & "','" & TextBox6.Text & "','" & TextBox7.Text & "')", CN)

CMD.ExecuteNonQuery()

CN.Close()

MOSTRAR()

MessageBox.Show("Usuario registrado satisfactoriamente")

End Sub

```

Sub ACTUALIZARU(), Segundo proceso del botón "Modificar"
    Dim CMD As New OleDb.OleDbCommand("UPDATE [user] SET Codigo=" &
    TextBox1.Text & ",Nombre=" & TextBox2.Text & ",Tipo=" & TextBox3.Text &
    ",Documento=" & TextBox4.Text & ",Carnet=" & TextBox5.Text & ",Vehiculo=" &
    TextBox6.Text & ",Matricula=" & TextBox7.Text & " WHERE Codigo=" &
    TextBox1.Text & """, CN)
        CN.Open()
        CMD.ExecuteNonQuery()
        CN.Close()
        MOSTRAR()
        MessageBox.Show("Los datos del usuario han sido actualizados")
End Sub

Sub LIMPIARU()
    TextBox1.Clear()
    TextBox2.Clear()
    TextBox3.Clear()
    TextBox4.Clear()
    TextBox5.Clear()
    TextBox6.Clear()
    TextBox7.Clear()
End Sub

Private Sub Button9_Click_1(sender As Object, e As EventArgs) Handles
Button9.Click, Proceso para volver al menú
    Hide()
    Form2.Show()
End Sub
End Class

```

Anexo C. Código de la ventana Lector/Escritor

```
Option Strict Off
Option Explicit On
Imports VB = Microsoft.VisualBasic
Friend Class frm_main
    Inherits System.Windows.Forms.Form
    'Dim CN As New
OleDb.OleDbConnection("PROVIDER=MICROSOFT.ACE.OLEDB.12.0;DATA
SOURCE=C:\Users\FAMIGOM\Desktop\usuarios.accdb")
    Dim CN As New
OleDb.OleDbConnection("PROVIDER=MICROSOFT.ACE.OLEDB.12.0;DATA
SOURCE=C:\Users\Familiar\Desktop\usuarios.accdb")

    Private Sub Command1_Click(ByVal eventSender As System.Object, ByVal
eventArgs As System.EventArgs) Handles Command1.Click
        Dim port, i, baud As Integer
            Dim j As Short
            Dim buf1(200) As Byte
            Dim b1 As Byte
            Dim s1 As String
            port = cb_ckh.SelectedIndex + 1
            If (port = 0) Then
                lb_info.Text = "Por favor seleccione el puerto COM!"
            End If
            baud = CInt(cb_btl.Text)
            If (baud = 0) Then
                lb_info.Text = "Por favor seleccione el numero de bits por segundo!"
            End If
        'Abrir puerto
            i = rf_init_com(port, baud)
```

```

        If (i <> 0) Then
            lb_info.Text = "El módulo RFID no esta conectado!"
            Exit Sub
        End If
'Solicitar
        i = rf_request(0, &H52s, j)
        If (i <> 0) Then
            lb_info.Text = "La solicitud ha fallado! por favor coloque la tarjeta
nuevamente en el lector y pulse el botón de Solicitud de ingreso."
            Command2.Enabled = False
            Exit Sub
        End If
'anticolisión
        i = rf_anticoll(0, 4, buf1(0), b1)
        If (i <> 0) Then
            lb_info.Text = "El proceso de anticolisión fallo!"
            Exit Sub
        End If
        s1 = ""
        For i = 0 To b1 - 1
            s1 = s1 & VB.Right("00" & Hex(buf1(i)), 2)
        Next i
        tx_kh.Text = s1
'Seleccion de tarjeta
        i = rf_select(0, buf1(0), 4, b1)
        If (i <> 0) Then
            lb_info.Text = "La seleccion de la tarjeta ha fallado!"
            Exit Sub
        End If
lb_info.Text = "Tarjeta seleccionada exitosamente!"
Command2.Enabled = True

```


End Sub

Private Sub Command2_Click(ByVal eventSender As System.Object, ByVal
eventArgs As System.EventArgs) Handles Command2.Click

TextBox1.Clear()

TextBox2.Clear()

TextBox3.Clear()

TextBox4.Clear()

TextBox5.Clear()

TextBox6.Clear()

Dim i, m As Integer

Dim buf1(200) As Byte

Dim buf2(200) As Byte

Dim s1 As String

Dim b2, b1, b3 As Byte

s1 = Trim(tx_key.Text)

If (Len(s1) <> 12) Then

lb_info.Text = "La longitud de la clave es incorrecta!"

tx_key.Focus()

Exit Sub

End If

For i = 0 To 5

buf1(i) = Val("&H" & Mid(s1, i * 2 + 1, 2))

Next i

m = cb_kh.SelectedIndex

If (m = -1) Then

lb_info.Text = "Seleccione el bloque por favor!"

Exit Sub

End If

If (op_a.Checked) Then

```

        b1 = &H60s
    End If
    If (op_b.Checked) Then
        b1 = &H61s
    End If
    b3 = CByte(m)
'Autenticación
    i = rf_M1_authentication2(0, b1, b3, buf1(0))
    If (i <> 0) Then
        lb_info.Text = "El proceso de autenticación ha fallado, por favor no retire la
tarjeta! Coloque la tarjeta en el lector y pulse el botón Solicitar ingreso
nuevamente."
        Exit Sub
    End If
'Lectura de la tarjeta
    i = rf_M1_read(0, b3, buf2(0), b2)
    If (i <> 0) Then
        lb_info.Text = "La lectura de la tarjeta ha fallado!"
        Exit Sub
    End If
    s1 = ""
    For i = 0 To b2 - 1
        s1 = s1 & VB.Right("00" & Hex(buf2(i)), 2)
    Next i
    tx_sj.Text = s1
lb_info.Text = "Lectura exitosa!"
cb_ckh.Enabled = False
cb_kh.Enabled = False
Command1.Enabled = False
Command2.Enabled = False
Button1.Enabled = True

```

```
Command3.Enabled = True
tx_sj.Enabled = True
TextBox1.Enabled = True
TextBox2.Enabled = True
TextBox3.Enabled = True
TextBox4.Enabled = True
TextBox5.Enabled = True
TextBox6.Enabled = True
End Sub
```

```
Private Sub Command3_Click(ByVal eventSender As System.Object, ByVal
eventArgs As System.EventArgs) Handles Command3.Click
    If tx_sj.Text = "" Then MessageBox.Show("Código no puede estar vacío",
"Alerta") : Exit Sub
    If TextBox1.Text = "" Then MessageBox.Show("Nombre no puede estar vacío",
"Alerta") : Exit Sub
    If TextBox2.Text = "" Then MessageBox.Show("Tipo no puede estar vacío",
"Alerta") : Exit Sub
    If TextBox3.Text = "" Then MessageBox.Show("Documento no puede estar
vacío", "Alerta") : Exit Sub
    If TextBox4.Text = "" Then MessageBox.Show("Carnet no puede estar vacío",
"Alerta") : Exit Sub
    If TextBox5.Text = "" Then MessageBox.Show("Vehiculo no puede estar
vacío", "Alerta") : Exit Sub
    If TextBox6.Text = "" Then MessageBox.Show("Placa no puede estar vacío",
"Alerta") : Exit Sub
```

```
Dim i, m As Integer
Dim buf1(200) As Byte
Dim buf2(200) As Byte
Dim s1 As String
```

```

Dim b2, b1, b3 As Byte
s1 = Trim(tx_key.Text)
If (Len(s1) <> 12) Then
    lb_info.Text = "La longitud de la clave es incorrecta!"
    tx_key.Focus()
    Exit Sub
End If
For i = 0 To 5
    buf1(i) = Val("&H" & Mid(s1, i * 2 + 1, 2))
Next i
m = cb_kh.SelectedIndex
If (m = -1) Then
    lb_info.Text = "Seleccione un bloque porfavor!"
    Exit Sub
End If
If (op_a.Checked) Then
    b1 = &H60S
End If
If (op_b.Checked) Then
    b1 = &H61S
End If

s1 = Trim(tx_sj.Text)
If (Len(s1) <> 32) Then
    lb_info.Text = "La longitud de los datos es incorrecta "
    tx_sj.Focus()
    Exit Sub
End If
For i = 0 To 15
    buf2(i) = Val("&H" & Mid(s1, i * 2 + 1, 2))
Next i

```

```

'Autenticación
b3 = CByte(m)
i = rf_M1_authentication2(0, b1, b3, buf1(0))
If (i <> 0) Then
    lb_info.Text = "La autenticación ha fallado, por favor no retire la tarjeta hasta
terminado todo el proceso! pulse el botón cancelar, coloque la tarjeta en el lector y
pulse el botón Solicitar ingreso nuevamente. "
    Exit Sub
End If
'Escritura de la tarjeta
i = rf_M1_write(0, b3, buf2(0))
If (i <> 0) Then
    lb_info.Text = "La escritura de la tarjeta ha fallado!"
    Exit Sub
End If
NUEVOU()
End Sub

```

```

Private Sub Command4_Click(ByVal eventSender As System.Object, ByVal
eventArgs As System.EventArgs) Handles Command4.Click
    Dim i As Integer
    i = rf_halt(0)
    If (i <> 0) Then
        lb_info.Text = "Detención fallida!"
        Exit Sub
    End If
    lb_info.Text = "Detención exitosa"
End Sub

```

```

Private Sub frm_main_Load(ByVal eventSender As System.Object, ByVal
eventArgs As System.EventArgs) Handles MyBase.Load

```

Info.Text = "Coloque la tarjeta sobre el lector, seleccione el puerto COM donde esta conectado el módulo, luego de clic izquierdo en el botón Solicitar ingreso, despues elija el bloque 1 y de clic izquierdo en el botón Leer, para finalizar ingrese la información solicitada en el panel de datos. Para asignar el codigo rempleemplaze los ultimos valores de 0s por el numero de documento ya sea CC o TI, dejando los otros valores intactos."

```
Dim i As Integer
```

```
For i = 0 To 63
```

```
    cb_kh.Items.Insert(i, CStr(i))
```

```
Next i
```

```
cb_ckh.SelectedIndex = 0 : cb_btl.SelectedIndex = 1 : cb_kh.SelectedIndex =
```

4

```
End Sub
```

```
Private Sub lb_info_Click(sender As Object, e As EventArgs) Handles  
lb_info.Click
```

```
End Sub
```

```
Sub NUEVOU() 'procedimiento para agregar un nuevo usuario
```

```
    Dim CMD As New OleDb.OleDbCommand("INSERT INTO [user] VALUES('" &  
tx_sj.Text & "','" & TextBox1.Text & "','" & TextBox6.Text & "','" & TextBox2.Text &  
"',"' & TextBox3.Text & "','" & TextBox5.Text & "','" & TextBox4.Text & "')", CN)
```

```
    CN.Open()
```

```
    CMD.ExecuteNonQuery()
```

```
    CN.Close()
```

```
    tx_sj.Clear()
```

```
    TextBox1.Clear()
```

```
    TextBox2.Clear()
```

```
    TextBox3.Clear()
```

```
    TextBox4.Clear()
```

```
    TextBox5.Clear()
```

```

    TextBox6.Clear()
    Button1.Enabled = False
    tx_sj.Enabled = False
    TextBox1.Enabled = False
    TextBox2.Enabled = False
    TextBox3.Enabled = False
    TextBox4.Enabled = False
    TextBox5.Enabled = False
    TextBox6.Enabled = False
    Command3.Enabled = False
    Command2.Enabled = True
    lb_info.Text = "Escritura exitosa!"
End Sub

Private Sub Button9_Click(sender As Object, e As EventArgs) Handles
Button9.Click
    Close()
    Form2.Show()
End Sub

Private Sub cb_kh_SelectedIndexChanged(sender As Object, e As EventArgs)
Handles cb_kh.SelectedIndexChanged

End Sub

Private Sub Label3_Click(sender As Object, e As EventArgs) Handles
Label3.Click

End Sub

Private Sub tx_key_TextChanged(sender As Object, e As EventArgs) Handles
tx_key.TextChanged

```

End Sub

```
Private Sub cb_btl_SelectedIndexChanged(sender As Object, e As EventArgs)  
Handles cb_btl.SelectedIndexChanged
```

End Sub

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles  
Button1.Click
```

```
    tx_sj.Clear()  
    TextBox1.Clear()  
    TextBox2.Clear()  
    TextBox3.Clear()  
    TextBox4.Clear()  
    TextBox5.Clear()  
    TextBox6.Clear()  
    cb_ckh.Enabled = True  
    cb_kh.Enabled = True  
    Command1.Enabled = True  
    Button1.Enabled = False  
    Command2.Enabled = True  
    Command3.Enabled = False  
    tx_sj.Enabled = False  
    TextBox1.Enabled = False  
    TextBox2.Enabled = False  
    TextBox3.Enabled = False  
    TextBox4.Enabled = False  
    TextBox5.Enabled = False  
    TextBox6.Enabled = False
```

End Sub

End Class

Anexo D. Código de la ventana verificar

```
Option Strict Off
Option Explicit On
Imports VB = Microsoft.VisualBasic
Friend Class frm_main2
    Inherits System.Windows.Forms.Form
    'Dim CN As New
    OleDb.OleDbConnection("PROVIDER=MICROSOFT.ACE.OLEDB.12.0;DATA
SOURCE=C:\Users\FAMIGOM\Desktop\usuarios.accdb")
    Dim CN As New
    OleDb.OleDbConnection("PROVIDER=MICROSOFT.ACE.OLEDB.12.0;DATA
SOURCE=C:\Users\Familiar\Desktop\usuarios.accdb")

    Private Sub Command1_Click(ByVal eventSender As System.Object, ByVal
eventArgs As System.EventArgs) Handles Command1.Click
        Dim port, i, baud As Integer
        Dim j As Short
        Dim buf1(200) As Byte
        Dim b1 As Byte
        Dim s1 As String
        port = cb_ckh.SelectedIndex + 1
        If (port = 0) Then
            lb_info.Text = "Por favor seleccione el puerto COM!"
        End If
        baud = CInt(cb_btl.Text)
        If (baud = 0) Then
            lb_info.Text = "Por favor seleccione el numero de bits por segundo!"
        End If
        'Abrir puerto
        i = rf_init_com(port, baud)
```

```

If (i <> 0) Then
    lb_info.Text = "El módulo RFID no está conectado!"
    Exit Sub
End If
'Solicitar
i = rf_request(0, &H52S, j)
If (i <> 0) Then
    lb_info.Text = "La solicitud ha fallado!"
    Exit Sub
End If
'anticolisión
i = rf_anticoll(0, 4, buf1(0), b1)
If (i <> 0) Then
    lb_info.Text = "El proceso de anticolisión fallo!"
    Exit Sub
End If
s1 = ""
For i = 0 To b1 - 1
    s1 = s1 & VB.Right("00" & Hex(buf1(i)), 2)
Next i
tx_kh.Text = s1
'Seleccion de tarjeta
i = rf_select(0, buf1(0), 4, b1)
If (i <> 0) Then
    lb_info.Text = "La seleccion de la tarjeta ha fallado!"
    Exit Sub
End If
lb_info.Text = "Tarjeta seleccionada exitosamente!"
Command2.Enabled = True
End Sub

```

```
Private Sub Command2_Click(ByVal eventSender As System.Object, ByVal  
eventArgs As System.EventArgs) Handles Command2.Click
```

```
    Dim i, m As Integer
```

```
    Dim buf1(200) As Byte
```

```
    Dim buf2(200) As Byte
```

```
    Dim s1 As String
```

```
    Dim b2, b1, b3 As Byte
```

```
    s1 = Trim(tx_key.Text)
```

```
    If (Len(s1) <> 12) Then
```

```
        lb_info.Text = "La longitud de la clave es incorrecta!"
```

```
        tx_key.Focus()
```

```
        Exit Sub
```

```
    End If
```

```
    For i = 0 To 5
```

```
        buf1(i) = Val("&H" & Mid(s1, i * 2 + 1, 2))
```

```
    Next i
```

```
    m = cb_kh.SelectedIndex
```

```
    If (m = -1) Then
```

```
        lb_info.Text = "Seleccione el bloque por favor!"
```

```
        Exit Sub
```

```
    End If
```

```
    If (op_a.Checked) Then
```

```
        b1 = &H60S
```

```
    End If
```

```
    If (op_b.Checked) Then
```

```
        b1 = &H61S
```

```
    End If
```

```
    b3 = CByte(m)
```

```
    'Autenticación
```

```
    i = rf_M1_authentication2(0, b1, b3, buf1(0))
```

```
    If (i <> 0) Then
```

```

        lb_info.Text = "El proceso de autenticación ha fallado!"
    Exit Sub
End If
'Lectura de la tarjeta
i = rf_M1_read(0, b3, buf2(0), b2)
If (i <> 0) Then
    lb_info.Text = "La lectura de la tarjeta ha fallado!"
    Exit Sub
End If
s1 = ""
For i = 0 To b2 - 1
    s1 = s1 & VB.Right("00" & Hex(buf2(i)), 2)
Next i
tx_sj.Text = s1
lb_info.Text = "Lectura exitosa!"
End Sub

```

```

Private Sub Command3_Click(ByVal eventSender As System.Object, ByVal
eventArgs As System.EventArgs) Handles Command3.Click

```

```

    Dim i, m As Integer
    Dim buf1(200) As Byte
    Dim buf2(200) As Byte
    Dim s1 As String
    Dim b2, b1, b3 As Byte
    s1 = Trim(tx_key.Text)
    If (Len(s1) <> 12) Then
        lb_info.Text = "La longitud de la clave es incorrecta!"
        tx_key.Focus()
        Exit Sub
    End If
    For i = 0 To 5

```

```

        buf1(i) = Val("&H" & Mid(s1, i * 2 + 1, 2))
    Next i
    m = cb_kh.SelectedIndex
    If (m = -1) Then
        lb_info.Text = "Seleccione un bloque porfavor!"
        Exit Sub
    End If
    If (op_a.Checked) Then
        b1 = &H60S
    End If
    If (op_b.Checked) Then
        b1 = &H61S
    End If

    s1 = Trim(tx_sj.Text)
    If (Len(s1) <> 32) Then
        lb_info.Text = "La longitud de los datos es incorrecta "
        tx_sj.Focus()
        Exit Sub
    End If
    For i = 0 To 15
        buf2(i) = Val("&H" & Mid(s1, i * 2 + 1, 2))
    Next i
    'Autenticación
    b3 = CByte(m)
    i = rf_M1_authentication2(0, b1, b3, buf1(0))
    If (i <> 0) Then
        lb_info.Text = "La autenticación ha fallado!"
        Exit Sub
    End If
    'Escritura de la tarjeta

```

```

i = rf_M1_write(0, b3, buf2(0))
If (i <> 0) Then
    lb_info.Text = "La escritura de la tarjeta ha fallado!"
    Exit Sub
End If

```

```

    lb_info.Text = "Escritura exitosa!"
End Sub

```

```

Private Sub Command4_Click(ByVal eventSender As System.Object, ByVal
eventArgs As System.EventArgs) Handles Command4.Click

```

```

    Dim i As Integer
    i = rf_halt(0)
    If (i <> 0) Then
        lb_info.Text = "Detención fallida!"
        Exit Sub
    End If
    lb_info.Text = "Detención exitosa"
End Sub

```

```

Private Sub frm_main_Load(ByVal eventSender As System.Object, ByVal
eventArgs As System.EventArgs) Handles MyBase.Load

```

Info.Text = "Coloque la tarjeta sobre el lector, seleccione el puerto COM donde esta conectado el módulo, luego de clic derecho en el boton Solicitar ingreso, despues elija el bloque 1 y para finalizar de clic izquierdo en el botón Leer."

```

    Dim i As Integer
    For i = 0 To 63
        cb_kh.Items.Insert(i, CStr(i))
    Next i
    cb_ckh.SelectedIndex = 0 : cb_btl.SelectedIndex = 1 : cb_kh.SelectedIndex =

```

End Sub

```
Private Sub lb_info_Click(sender As Object, e As EventArgs) Handles  
lb_info.Click
```

End Sub

```
Private Sub Button9_Click(sender As Object, e As EventArgs) Handles  
Button9.Click
```

```
    Close()
```

```
    Form2.Show()
```

```
End Sub
```

```
Private Sub cb_kh_SelectedIndexChanged(sender As Object, e As EventArgs)  
Handles cb_kh.SelectedIndexChanged
```

End Sub

```
Private Sub Label3_Click(sender As Object, e As EventArgs) Handles  
Label3.Click
```

End Sub

```
Private Sub tx_key_TextChanged(sender As Object, e As EventArgs) Handles  
tx_key.TextChanged
```

End Sub

```
Private Sub cb_btl_SelectedIndexChanged(sender As Object, e As EventArgs)  
Handles cb_btl.SelectedIndexChanged
```

```

End Sub

Private Sub tx_sj_TextChanged(sender As Object, e As EventArgs) Handles
tx_sj.TextChanged
    Dim CMD As New OleDb.OleDbCommand("SELECT * FROM [user] WHERE
Codigo=" & tx_sj.Text & "", CN)
    Dim DR As OleDb.OleDbDataReader

    CN.Open()

    DR = CMD.ExecuteReader

    If DR.Read Then
        TextBox1.Text = DR(1)
        TextBox2.Text = DR(2)
        TextBox3.Text = DR(3)
        TextBox4.Text = DR(4)
        TextBox5.Text = DR(5)
        TextBox6.Text = DR(6)
    Else
        TextBox1.Clear()
        TextBox2.Clear()
        TextBox3.Clear()
        TextBox4.Clear()
        TextBox5.Clear()
        TextBox6.Clear()
        MessageBox.Show("Usuario no registrado", "Alerta")
    End If
    CN.Close()
End Sub
End Class

```


Anexo E. Código de enlace con DLL

Option Strict Off

Option Explicit On

Module mo_declare

'Para poder acceder al puerto COM

Public Declare Function rf_init_com Lib "MasterRD.dll" (ByVal port As Integer, ByVal baud As Integer) As Integer

'Para cerrar el puerto COM una vez este en uso

Public Declare Function rf_ClosePort Lib "MasterRD.dll" () As Integer

'Para reconocer el tipo de tarjeta RFID y su número de serieint WINAPI

Public Declare Function rf_request Lib "MasterRD.dll" (ByVal icdev As Short, ByVal model As Byte, ByRef TagType As Short) As Integer

'Anticolisión que permite leer varias tarjetas a la vez

Public Declare Function rf_anticoll Lib "MasterRD.dll" (ByVal icdev As Short, ByVal bcnt As Byte, ByRef ppsnr As Byte, ByRef pRLength As Byte) As Integer

'Para seleccionar el puerto COM, el numero de bits por segundo y el bloque

Public Declare Function rf_select Lib "MasterRD.dll" (ByVal icdev As Short, ByRef pSnr As Byte, ByVal srclen As Byte, ByRef size As Byte) As Integer

'Para la autenticación de las claves A y B

Public Declare Function rf_M1_authentication2 Lib "MasterRD.dll" (ByVal icdev As Short, ByVal model As Byte, ByVal block As Byte, ByRef key As Byte) As Integer

'Para leer los datos de la tarjeta

Public Declare Function rf_M1_read Lib "MasterRD.dll" (ByVal icdev As Short, ByVal block As Byte, ByRef buff As Byte, ByRef pLen As Byte) As Integer

```
'Para escribir los datos en las tarjetas
Public Declare Function rf_M1_write Lib "MasterRD.dll" (ByVal icdev As Short,
ByVal block As Byte, ByRef buff As Byte) As Integer

'para detener la tarjeta
Public Declare Function rf_halt Lib "MasterRD.dll" (ByVal icdev As Short) As
Integer
End Module
```

Anexo F. Código de la ventana Login

```
Public Class Form3
```

```
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles  
Button1.Click, botón salir
```

```
        Close()
```

```
    End Sub
```

```
    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles  
Button2.Click, Se define el usuario y la contraseña
```

```
        If usuario.Text = "pascualbravo" And contraseña.Text = "electronica" Then
```

```
            Timer1.Start()
```

```
            ProgressBar1.Show()
```

```
        Else
```

```
            MessageBox.Show("Ingrese los datos correctos por favor", "Alerta")
```

```
        End If
```

```
    End Sub
```

```
    Private Sub Form3_Load(sender As Object, e As EventArgs) Handles  
MyBase.Load, representación del proceso de carga
```

```
        ProgressBar1.Hide()
```

```
    End Sub
```

```
    Private Sub Timer1_Tick(sender As Object, e As EventArgs) Handles  
Timer1.Tick, Fijamos el tiempo de carga de la ventana menú
```

```
        ProgressBar1.Increment(35)
```

```
        If ProgressBar1.Value = ProgressBar1.Maximum Then
```

```
            Timer1.Stop()
```

```
            MessageBox.Show("Bienvenido!", "usuario")
```

```
        Hide()  
        Form2.Show()  
    End If  
End Sub  
End Class
```

Anexo G. Manual de usuario

1. REQUERIMIENTOS DE HARDWARE Y SOFTWARE

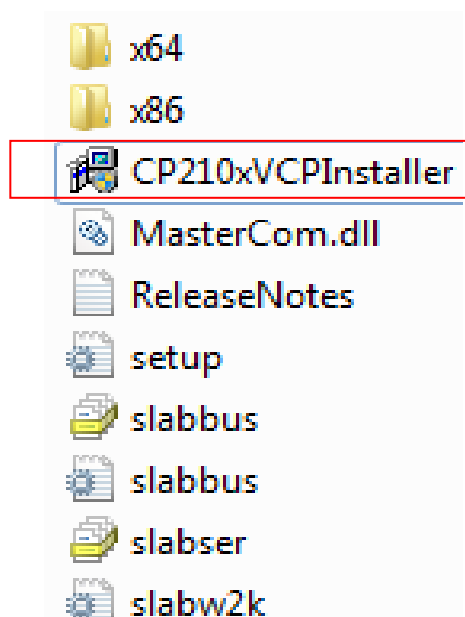
Los requerimientos recomendados de hardware y software son:

Hardware: procesador Pentium IV o superior, Memoria RAM de 256 Mb o superior.

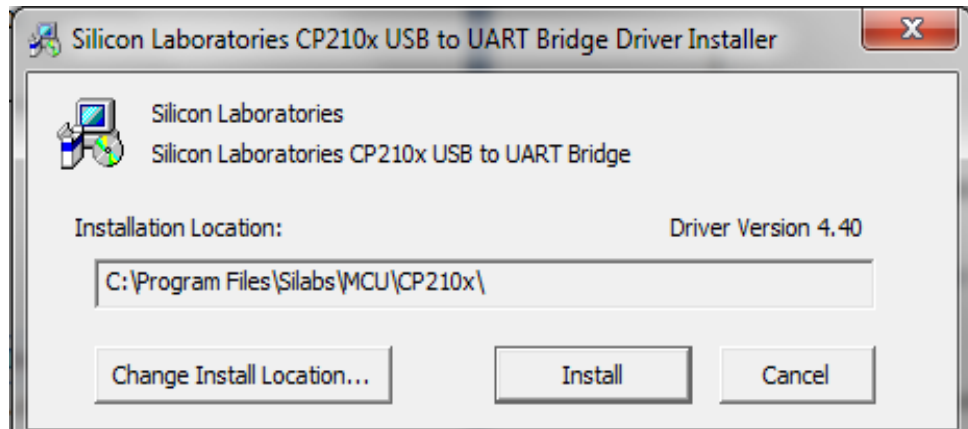
Software: sistema operativo Windows XP o superior, Microsoft Acces 2007, .NET Framework 4.5.

2. INTALACIÓN DEL SOFTWARE Y CONEXIÓN DEL MÓDULO RFID

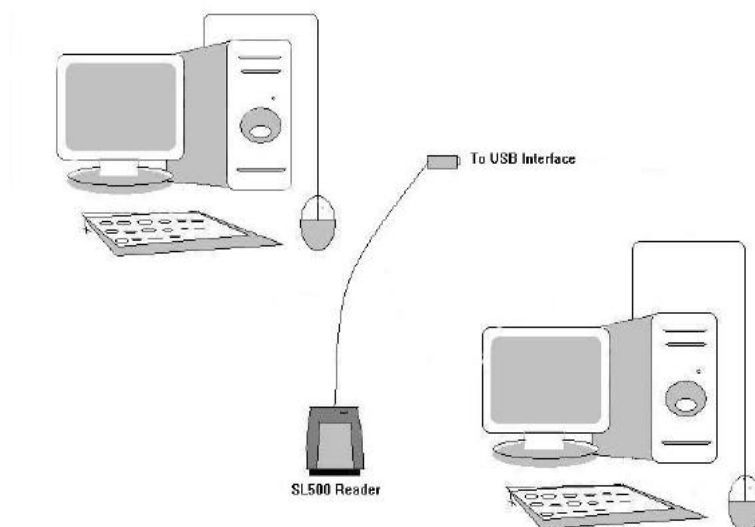
1. instale el driver "CP210xVCPInstaller" contenido en el CD, para ello haga doble clic izquierdo en el setup.



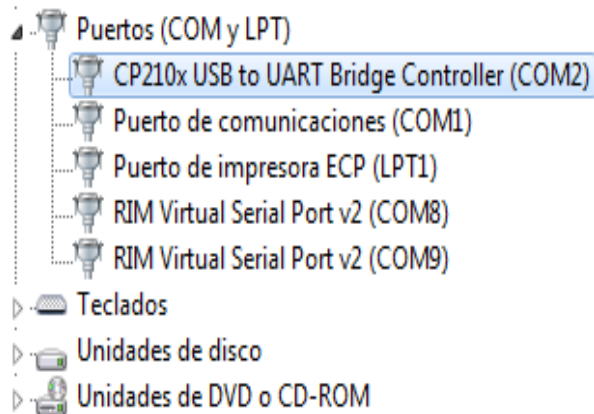
Luego de clic izquierdo en Install y espere a que se complete la instalación.



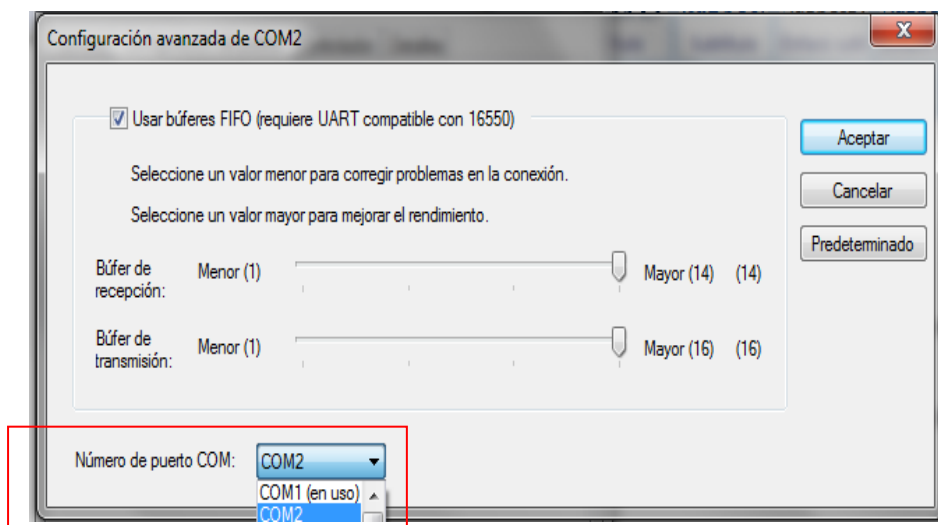
2. Conecte el módulo SL500L a un puerto USB del computador



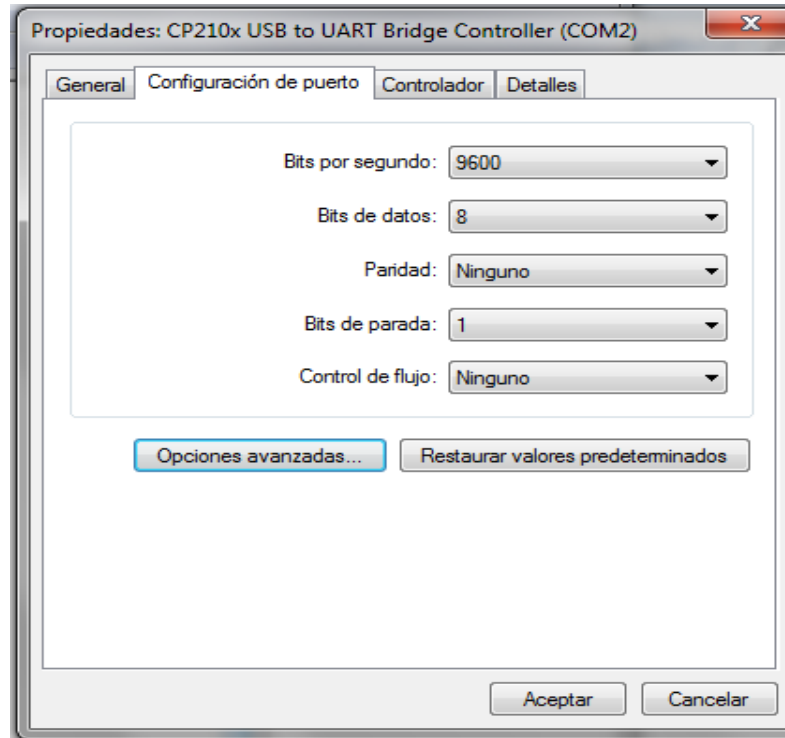
3. Asigne un puerto virtual COM del 1 al 9 y configúrelo, para hacerlo vaya al administrador de dispositivos y en el apartado Puertos (COM y LPT) encontrará el dispositivo.



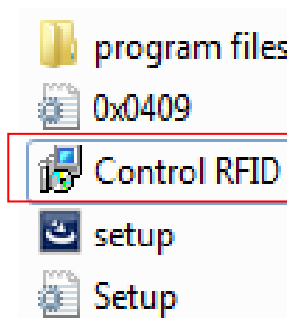
Ahora de clic derecho sobre él y elija la opción propiedades, a continuación de clic izquierdo en configuración de puerto y elija opciones avanzadas, luego seleccione el número de puerto COM del 1 al 9.



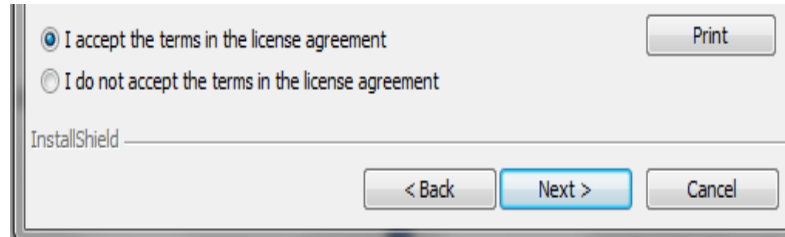
Ahora de clic izquierdo en aceptar y configure el puerto de acuerdo a la siguiente imagen.



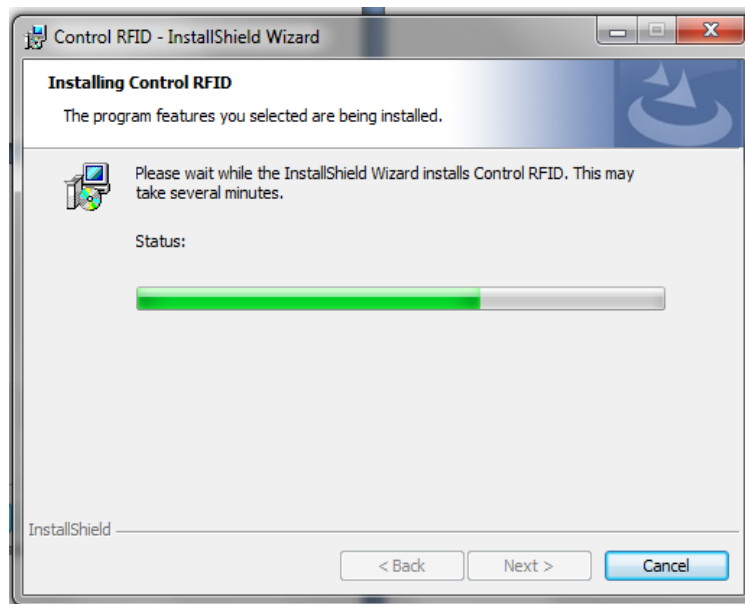
4. Ahora instale el software de control RFID contenido en el CD, la carpeta tiene por nombre Control RFID al igual que el ejecutable.



Luego de dar doble clic izquierdo aparecerá la pantalla de presentación de la instalación, de clic izquierdo en Next (siguiente) y después acepte los términos para poder continuar con la instalación.



En la siguiente pantalla de clic izquierdo en Next, igualmente de clic derecho en Install para que comience la instalación.



Ahora el programa encargado de la interacción con el usuario final ha sido instalado correctamente.

5. Para finalizar el CD contiene un archivo llamado usuarios.accdb, este archivo debe ser copiado y pegado en el escritorio ya que es la base de datos principal del sistema.

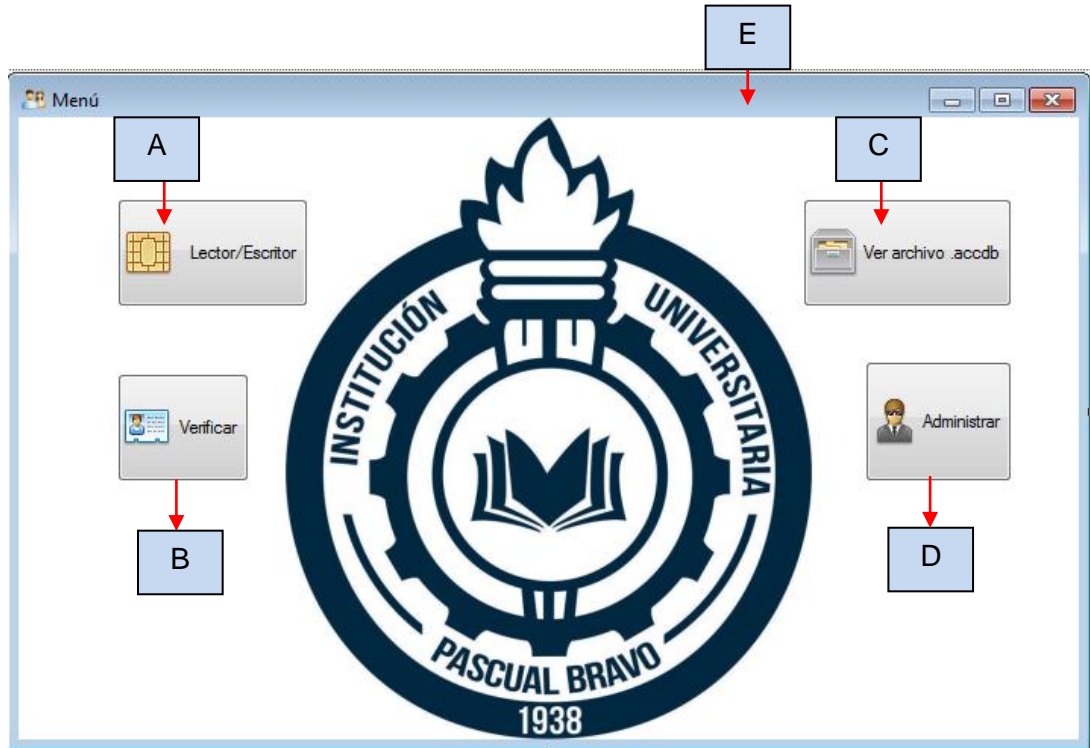
3. INGRESO AL SISTEMA

En esta parte se detalla brevemente la forma en que el usuario debe acceder al sistema:

1. En el escritorio se muestra el icono de acceso directo llamado Control RFID.



2. Dar doble clic izquierdo sobre el icono mostrado en el paso anterior.
3. A continuación aparece la ventana de menú donde se podrán visualizar las diferentes opciones.

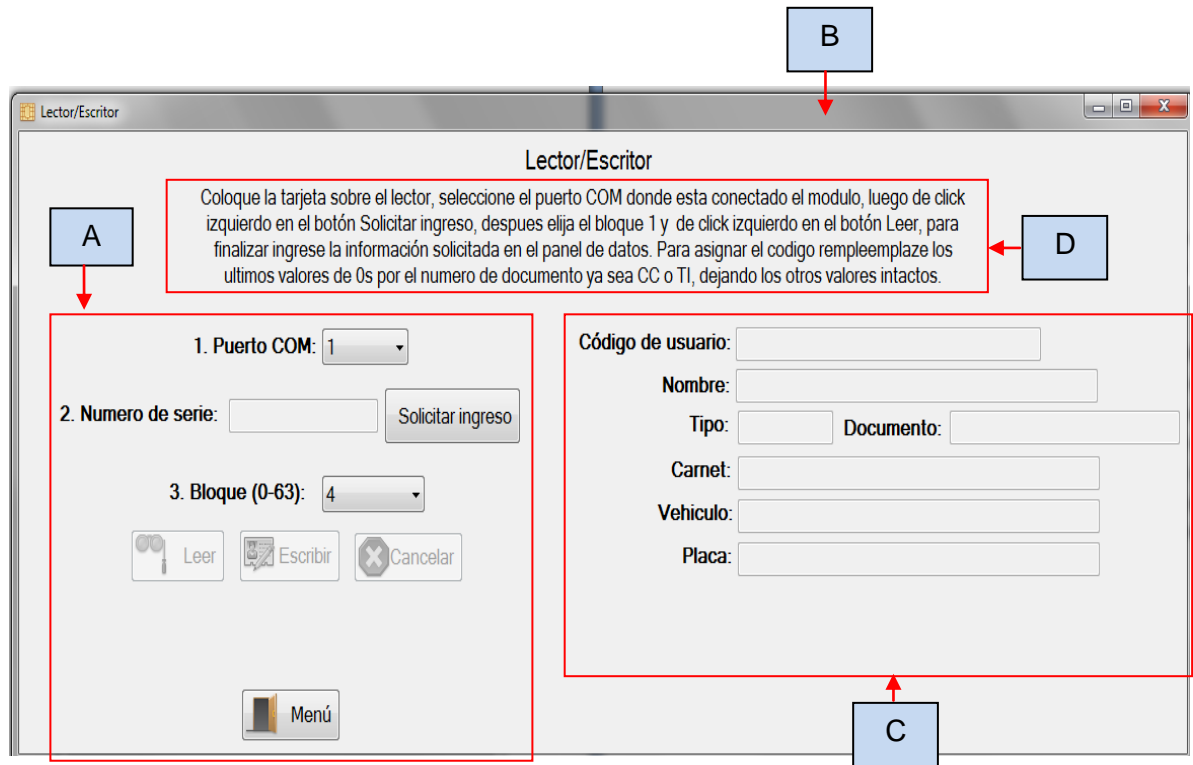


Dónde:

	Sección	Descripción
A	Botón Lector/Escritor	Accede a la ventana en donde se escriben los datos en la tarjeta RFID.
B	Botón Verificar	Permite ingresar a la ventana en donde se hace la verificación y lectura de datos proporcionados por el módulo RFID.
C	Botón Ver archivo .acddb	Permite abrir la base de datos original del sistema.
D	Botón Administrar	Permite ingresar a la ventana en donde se administra la base de datos.
E	Título de la ventana	Indica el nombre de la ventana.

3.1 VENTANA LECTOR/ESCRITOR

En esta sección se explica cómo está compuesta la ventana Lector/Escritor


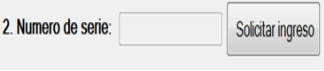
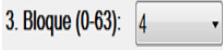

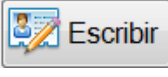




Dónde:

	Sección	Descripción
A	Panel de botones	Estos botones realizan operaciones necesarias para hacer una correcta asignación de datos
B	Título de la ventana	Indica el nombre de la pantalla, este título permite tener una idea referente a los datos que contienen dicha pantalla.
C	Panel de datos	En esta parte el usuario ingresa la información solicita por el programa.

	Sección	Descripción
D	Panel de comentarios	Acá se muestra información que guía al usuario final en el sistema.

3.1.1 Botones de la ventana Lector/Escritor: en esta sección se explica la función de los botones.

Botón	Descripción
 1. Puerto COM: 1	Permite elegir el puerto COM al cual está conectado el Módulo RFID
 2. Numero de serie: <input type="text"/> Solicitar ingreso	Verifica que la tarjeta esté al alcance del módulo.
 3. Bloque (0-63): 4	Permite elegir el bloque en el cual se escribirá el código de usuario.
 Leer	Permite consultar los datos de la tarjeta y habilita el panel de datos para ingresar la información requerida.
 Escribir	Escribe los datos en la tarjeta y registra el usuario en la base de datos.
 Cancelar	Cancela la operación en progreso.

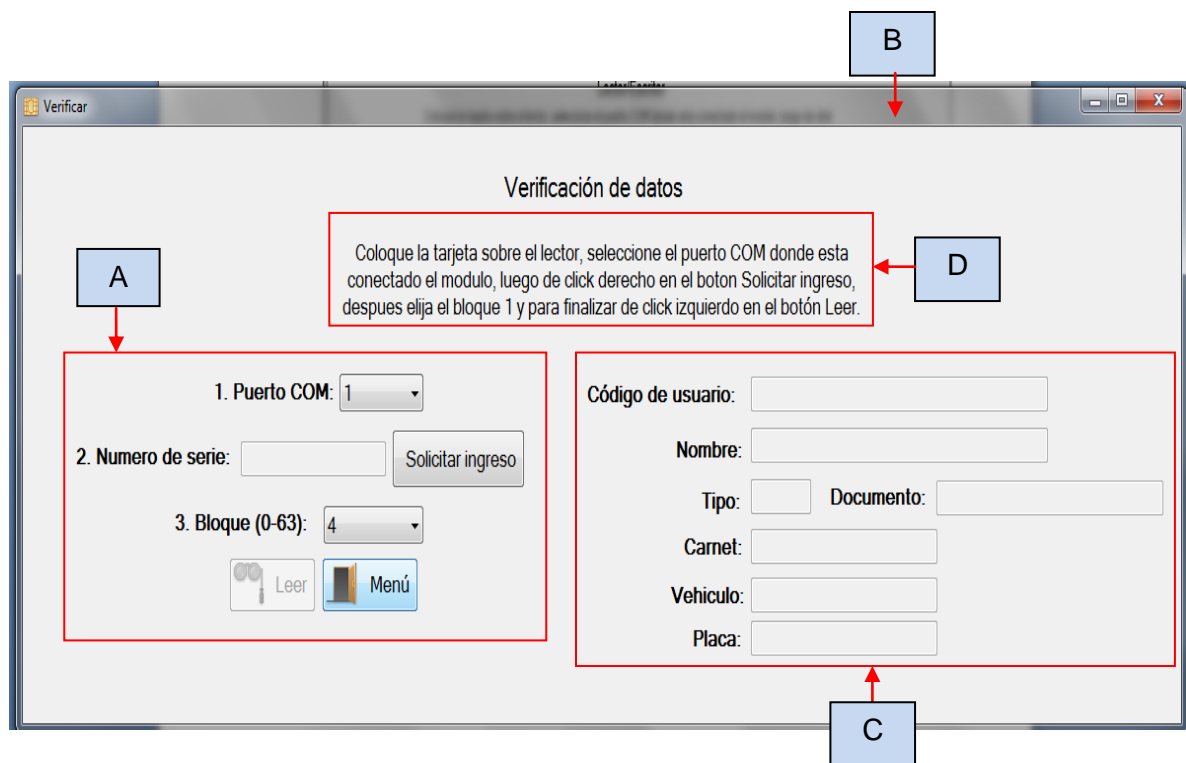
Botón	Descripción
	Permite volver al menú.

Para realizar el proceso de registro de un usuario:

1. coloque la tarjeta sobre el lector.
2. seleccione el puerto COM donde está conectado el módulo.
3. luego de clic izquierdo en el botón Solicitar ingreso.
4. después elija el bloque 1.
5. de clic izquierdo en el botón Leer
6. para finalizar ingrese la información solicitada en el panel de datos
7. Para asignar el código reemplace los últimos valores de 0s por el número de documento ya sea CC o TI, dejando los otros valores intactos.

3.2 VENTANA DE VERIFICACIÓN DE DATOS






En esta sección se explica cómo está compuesta la ventana de verificación.



Dónde:

	Sección	Descripción
A	Panel de botones	Estos botones realizan operaciones necesarias para hacer una correcta asignación de datos
B	Título de la ventana	Indica el nombre de la pantalla, este título permite tener una idea referente a los datos que contienen dicha pantalla.
C	Panel de datos	En esta parte se ven los datos del usuario al dar clic izquierdo en el botón leer.
D	Panel de comentarios	Acá se muestra información que guía al usuario final en el sistema.

3.2.1 Botones de la ventana Verificar: en esta sección se explica la función de los botones.

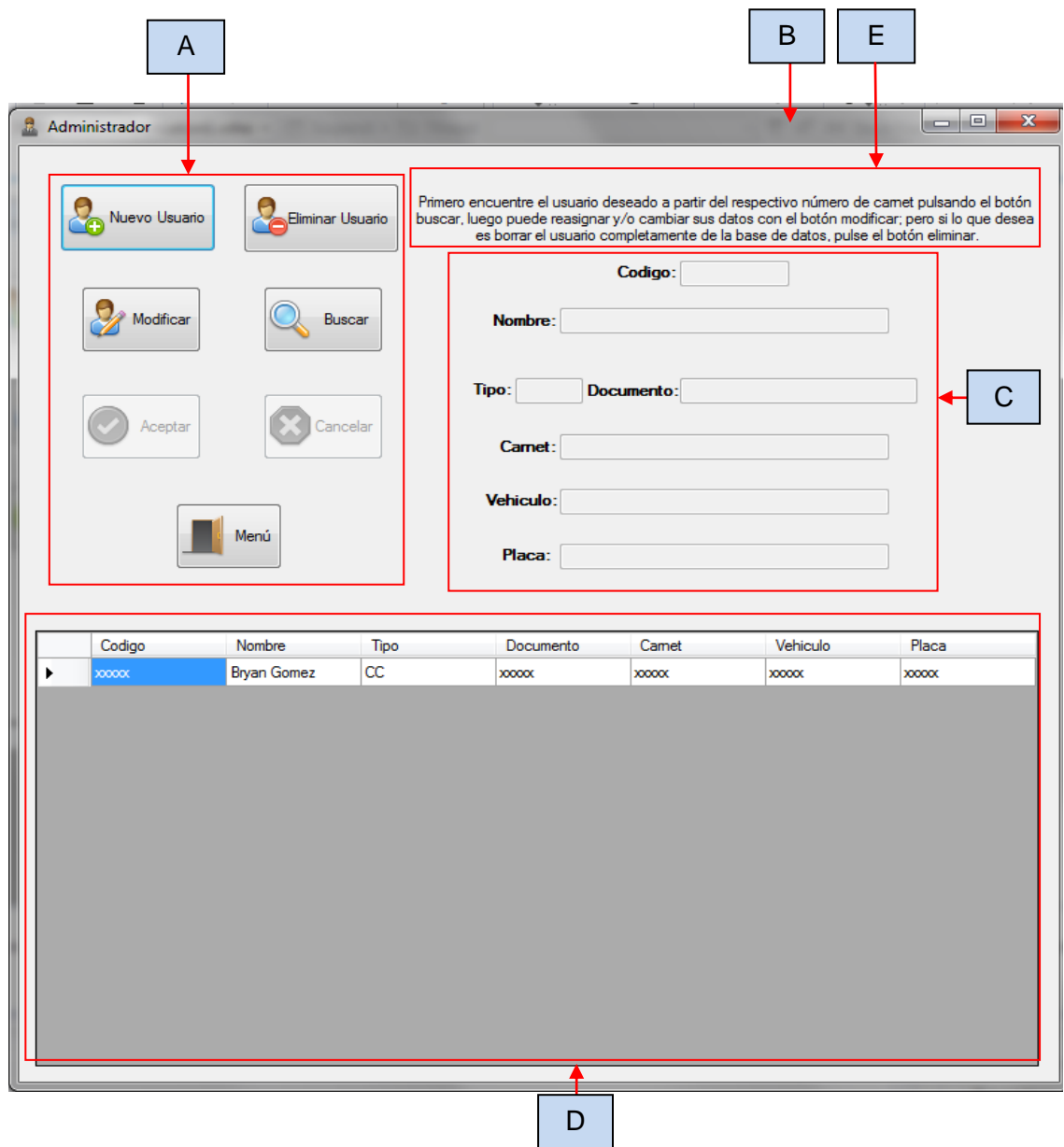
Botón	Descripción
	Permite elegir el puerto COM al cual está conectado el Módulo RFID
	Verifica que la tarjeta esté al alcance del módulo.
	Permite elegir el bloque el cual será leído.
	Permite consultar los datos de la tarjeta.
	Permite volver al menú.

Para realizar el proceso de lectura de un usuario:

1. coloque la tarjeta sobre el lector.
2. seleccione el puerto COM donde está conectado el módulo.
3. luego de clic izquierdo en el botón Solicitar ingreso.
4. después elija el bloque 1.
5. de clic izquierdo en el botón Leer, a continuación aparecerán los respectivos datos del usuario.

3.3 VENTANA ADMINISTRAR

En esta sección se explica cómo está compuesta la ventana administrar.



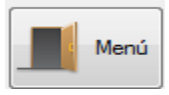




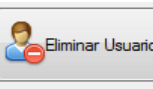
Dónde:

	Sección	Descripción
A	Panel de botones	Estos botones realizan operaciones necesarias para manejar los datos que se ingresarán.

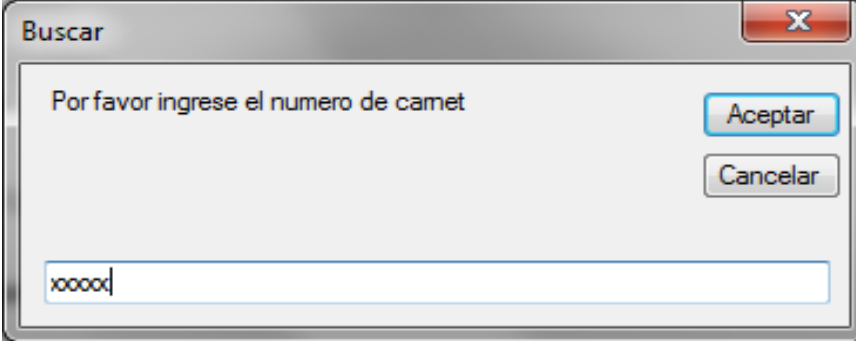
B	Título de la ventana	Indica el nombre de la pantalla, este título permite tener una idea referente a los datos que contienen dicha pantalla.
C	Panel de datos	En esta parte el usuario efectúa los tratamientos de los datos; es decir ingresa, consulta, modifica, y elimina los datos.
D	Datagrid view	Acá se muestra la base de datos de Acces que gracias al enlace por código se pueden ver las modificaciones en tiempo real.
E	Panel de comentarios	Acá se muestra información que guía al usuario final en el sistema.

3.3.1 Botones de la ventana administrar: en esta sección se describe la función principal de los botones que se encuentran en el panel; cada botón tiene una etiqueta que indica la función del mismo.

Botón	Descripción
	Finaliza la operación que se está efectuando guardando los cambios o datos nuevos.
Botón	Descripción
	Cancela la operación que se está efectuando en ese momento, dejando la ventana en su forma inicial.
	Permite volver al menú.

	Permite consultar los datos de un usuario ingresando su respectivo número de carnet.
	Habilita el panel de datos después de haber realizado una consulta para modificar la información existente.
	Elimina el registro actual en pantalla luego de haberlo consultado.

3.3.2 Buscar: para agregar buscar un usuario en específico presione el botón "buscar", a continuación aparecerá una ventana en donde deberá escribir el número de carnet del usuario.



Luego de ingresar el número de carnet y presionar el botón aceptar, aparecerán los datos del usuario en el panel de datos.

Codigo:

Nombre:

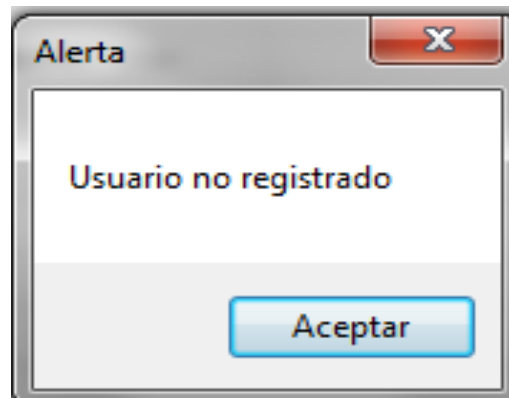
Tipo: **Documento:**

Camet:

Vehiculo:

Placa:

Pero si el usuario no se encuentra en la base de datos aparecerá un mensaje indicando que el usuario no está registrado.



3.3.3 Modificar: la opción modificar solo se habilitará si anteriormente se ha buscado un usuario, al pulsar este botón se habilitará el panel de datos para modificar la información existente.

Para confirmar y guardar los cambios pulse el botón aceptar, de lo contrario pulse el botón cancelar.

Codigo:

Nombre:

Tipo: **Documento:**

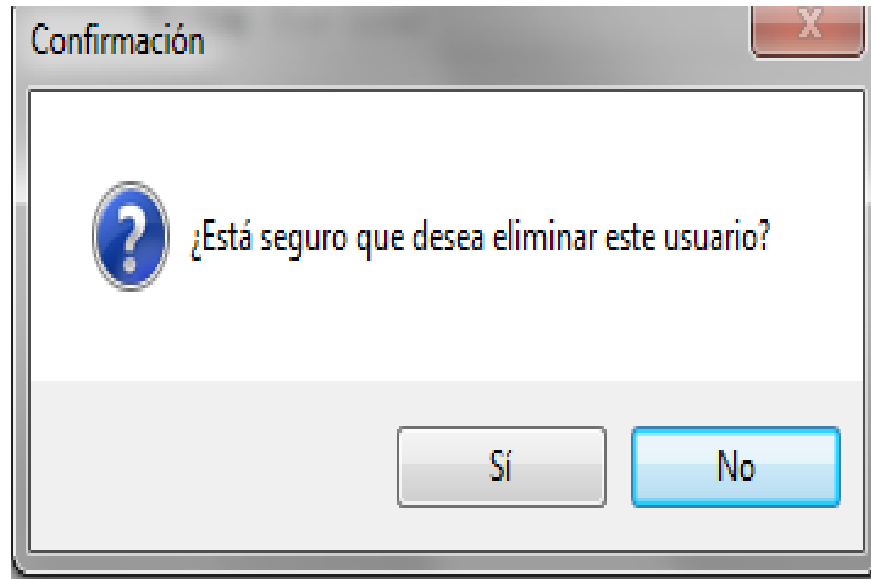
Carnet:

Vehiculo:

Placa:

Recuerde que ningún espacio debe estar sin diligenciar, de lo contrario al pulsar el botón aceptar sucederá lo mismo que con la opción de nuevo usuario y aparecerá un mensaje informando cual espacio está en blanco.

3.3.4 Eliminar usuario: la opción de eliminar usuario solo se habilitará si anteriormente se ha buscado un usuario, al pulsar este botón aparecerá un mensaje para confirmar la decisión.



Si su elección es si, el usuario quedará totalmente eliminado de la base de datos pero si su elección es no, se regresará a la ventana en su forma inicial.

3.4 VER ARCHIVO .ACCDB

Este botón permite ingresar al archivo fuente de la base de datos con formato Microsoft Acces 2007 llamado usuarios.accdb, este archivo no puede ser modificado ya que cualquier alteración podría comprometer el correcto funcionamiento del software.

