

# **Sistema de control de acceso en el parqueadero de las bicicletas de una universidad**

Luisa Fernanda Rubio Marroquin

Universitaria Agustiniana  
Facultad de Ingenierías  
Programa de Ingeniería en Telecomunicaciones  
Bogotá D.C.  
2020

# **Sistema de control de acceso en el parqueadero de las bicicletas de una universidad**

Luisa Fernanda Rubio Marroquin

Director

Guillermo Fernando Valencia Plata

Trabajo de grado para optar al título de Ingeniera en Telecomunicaciones

Universitaria Agustiniana

Facultad de Ingenierías

Programa de Ingeniería en Telecomunicaciones

Bogotá D.C.

2020

## **Dedicatoria**

*Dedico este proyecto de grado primeramente a Dios, quién ha sido el pilar e impulso para sacar adelante este proyecto. A mis padres, quienes me han brindado su apoyo incondicional y sus maravillosos consejos para lograr cumplir cada uno de mis objetivos propuestos. A mis hermanos, Tatiana y Daniel, quienes han sido una de las mayores motivaciones para seguir adelante, brindándome su apoyo y amor en todo momento.*

### **Agradecimientos**

Agradezco a Dios y a mi familia por ser mi pilar de motivación para llevar a cabo este proyecto.

A cada una de las personas que han sido participes de este proceso educativo, y que me han aportado experiencias y enseñanzas valiosas para el desarrollo personal y académico.

Al ingeniero Guillermo Valencia por su ayuda para la elaboración de este proyecto, por sus consejos y apoyo constante en el proceso hasta la culminación del mismo.

## **Resumen**

En este documento se describe el diseño de un sistema de seguridad para el control de acceso a un parqueadero de bicicletas y los diferentes módulos usados. Este sistema está compuesto por un Arduino UNO, un módulo lector RC522, un módulo ENC28J60 y una base de datos relacional, estos componentes son útiles para tomar las lecturas de las etiquetas, procesarlas, transmitir las, y validarlas según la información registrada en la base de datos, para determinar si se autoriza el acceso al parqueadero. Teniendo en cuenta que la universidad tiene una base de datos con la información de los estudiantes, se creó otra base de datos que se relacione con la existente, en la cual se almacena la información de las bicicletas mediante una página web con una interfaz intuitiva, esta permite el registro y consulta de datos correspondientes a la bicicleta, lo que facilita la administración de esta base de datos. Se asignaron etiquetas de tecnología NFC a las bicicletas, ya que su frecuencia de operación es de 13.56MHz igual que el carné institucional, esto permite tomar la lectura de las dos etiquetas con un solo módulo, y así logra la validación de la relación entre el carné institucional y la etiqueta asignada a la bicicleta.

*Palabras clave:* Etiqueta, RFID, NFC, Arduino, bicicleta, parqueadero.

## **Abstract**

This document describes the design of a security system for controlling access to a bicycle parking lot and the different modules used. This system is composed of an Arduino UNO, a RC522 reader module, an ENC28J60 module and a relational database, these components are useful to take the readings of tags, process them, transmit them, and validate them according to the existing registers in the database, to determine if access to the parking lot is authorized. Taking into account that the university has a database with the information of the students, another database was created that is related to the existing one, in which the information of the bicycles is stored through a web page with an intuitive interface. This will allow the registration and consultation of data corresponding to the bicycle, which facilitates the administration of this database. NFC tags were assigned to the bicycles, since their operating frequency is 13.56MHz the same as the institutional card, this allows the reading of the two tags with a single module, and thus the validation of the relationship between the institutional card and the tag assigned to the bicycle.

*Keywords:* Tag, RFID, NFC, Arduino, bicycle, parking lot

## Tabla de contenidos

Introducción .....	13
Planteamiento del problema.....	14
Justificación.....	15
Objetivos .....	16
Objetivo general .....	16
Objetivos específicos .....	16
Marco referencial .....	17
Estado del arte .....	17
Marco teórico .....	19
Etiquetas RFID.....	19
Near Field Communication (NFC).....	20
Frecuencia de operación para RFID.....	20
Arduino. ....	21
Marco legal.....	21
Ley 1581 de 2012.....	21
Ley 1341 de 2009.....	21
Resolución 711 de 2016. ....	22
Metodología de la investigación .....	24
Cronograma.....	25
Presupuesto .....	26
Requerimientos del sistema de control de acceso .....	28
Módulos lectores .....	29
Módulo Lector RC522. ....	29
Módulo RFID/NFC PN532.....	29

Microcontroladores .....	30
Arduino uno. ....	30
PIC18F4550. ....	31
Módulo de conexión a internet.....	31
Módulo Ethernet ENC28J60. ....	32
Ethernet Shield W5100 Arduino.....	32
Conexiones.....	33
Código conexión Arduino – Modulo Lector RFID.....	33
Conexión Lector RC552 – Arduino Uno. ....	34
Conexión del Arduino - Módulo Ethernet ENC28J60.....	35
Diseño del sistema de control de acceso .....	37
Base de datos.....	39
Procesamiento y comparación de la información obtenida.....	46
Resultados y pruebas del prototipo .....	50
Recomendaciones.....	63
Conclusiones .....	64
Referencias.....	65
Anexos.....	68
Código Arduino Uno.....	68
Código conexión Arduino – base de datos.....	73



## Lista de tablas

Tabla 1. Límites generales para dispositivos RFID .....	23
Tabla 2. Cronograma de Actividades.....	25
Tabla 3. Recursos Humanos.....	26
Tabla 4. Presupuesto de Equipos .....	26
Tabla 5. Presupuesto de Software .....	26
Tabla 6. Presupuesto Final .....	27
Tabla 7. Entorno estimado de ocupación del parqueadero de bicicletas.....	28
Tabla 8. Costos de los módulos lectores .....	30
Tabla 9. Características Arduino Uno .....	30
Tabla 10. Características PIC18F4550.....	31
Tabla 11. Costos de los módulos de ethernet .....	33
Tabla 12. Conexión Lector RC522- Arduino UNO.....	34
Tabla 13. Conexión Arduino UNO – ENC28J60.....	36
Tabla 14. Detalles de la solicitud de la página web. ....	50

## Lista de figuras

Figura 1. Módulo Lector RC522.....	29
Figura 2. Módulo Lector RFID/NFC PN532.....	30
Figura 3. Módulo Ethernet ENC28J60.....	32
Figura 4. Arduino Ethernet Shield.....	32
Figura 5. Inclusión Librería en Arduino IDE.....	33
Figura 6. Gestor de librerías en IDE Arduino.....	33
Figura 7. Lectura de la Etiqueta en el Monitor Serie.....	34
Figura 8. Código ID Monitor Serial del IDE.....	35
Figura 9. Circuito PCB Sistema Control de Acceso.....	37
Figura 10. Diagrama de elementos funcionales.....	38
Figura 11. Diagrama de red.....	39
Figura 12. Diagrama de Bases de Datos.....	39
Figura 13. Tabla "Bicicleta".....	40
Figura 14. Interfaz phpMyAdmin.....	40
Figura 15. Página de Inicio Aplicación Web.....	41
Figura 16. Ingreso de datos a la tabla "Bicicleta".....	41
Figura 17. Registro de datos Tabla "Bicicleta".....	41
Figura 18. Consulta en la Base de Datos.....	42
Figura 19. Resultado de la consulta.....	42
Figura 20. Diagrama de flujo código Arduino.....	44
Figura 21. Diagrama de flujo código rfid.php.....	45
Figura 22. Conexión del módulo Ethernet a la página web.....	46
Figura 23. Importación de librerías y definición de pines.....	46
Figura 24. Creación de la instancia para el lector.....	46
Figura 25. Inicialización Módulo Ethernet.....	47
Figura 26. Void setup.....	47
Figura 27. Código del lector.....	48
Figura 28. Envío de información al servidor.....	48
Figura 29. Código GET "rfid.php".....	49
Figura 30. Función "my_callback".....	49

Figura 31. Registro de datos de las bicicletas. ....	51
Figura 32. Registro satisfactorio. ....	51
Figura 33. Validación del registro.....	51
Figura 34. Relación de los UID según ID. ....	52
Figura 35. Montaje del prototipo. ....	52
Figura 36. Lectura carné primer prueba. ....	53
Figura 37. Monitor serial carné primera prueba.....	53
Figura 38. Monitor serial etiqueta bicicleta primera prueba.. ....	53
Figura 39. Lectura etiqueta bicicleta primer prueba.. ....	53
Figura 40. Lectura carné segunda prueba. ....	54
Figura 41. Monitor serial carné segunda prueba. ....	54
Figura 42. Monitor serial etiqueta bicicleta segunda prueba. ....	54
Figura 43. Lectura etiqueta bicicleta segunda prueba.....	55
Figura 44. Lectura carné tercera prueba.....	55
Figura 45. Monitor serial carné tercera prueba. ....	55
Figura 46. Monitor serial etiqueta tercera prueba.. ....	56
Figura 47. Lectura etiqueta bicicleta tercera prueba.. ....	56
Figura 48. Lectura carné cuarta prueba.....	56
Figura 49. Monitor serial carné cuarta prueba. ....	56
Figura 50. Monitor serial etiqueta cuarta prueba. ....	57
Figura 51. Lectura etiqueta bicicleta cuarta prueba. ....	57
Figura 52. Lectura carné quinta prueba.....	57
Figura 53. Monitor serial carné quinta prueba. ....	58
Figura 54. Monitor serial etiqueta quinta prueba.. ....	58
Figura 55. Lectura etiqueta bicicleta quinta prueba.....	58
Figura 56. Lectura carné sexta prueba. ....	59
Figura 57. Monitor serial carné sexta prueba.....	59
Figura 58. Monitor serial etiqueta bicicleta sexta prueba.. ....	59
Figura 59. Lectura etiqueta bicicleta sexta prueba.....	59
Figura 60. Lectura carné séptima prueba.. ....	60
Figura 61. Monitor serial carné séptima prueba.....	60

Figura 62. Monitor serial etiqueta séptima prueba.....	60
Figura 63. Lectura etiqueta bicicleta séptima prueba.....	61
Figura 64. Monitor serial carné octava prueba.....	61
Figura 65. Lectura carné octava prueba. ....	61
Figura 66. Monitor serial carné novena prueba.....	62
Figura 67. Monitor serial etiqueta bicicleta novena prueba.....	62

## **Introducción**

Debido al incremento del uso de las bicicletas por parte de los estudiantes de las universidades para movilizarse, las universidades se han visto en la necesidad de crear parqueaderos para atender dicha demanda. Buscando garantizar que la persona que extraiga del parqueadero una bicicleta sea su dueño, se desarrolló un prototipo de sistema de control de seguridad que permita relacionar cada bicicleta con su respectivo dueño usando el carné de la universidad, etiqueta RFID asignada a la bicicleta y unas bases de datos. Para el desarrollo de este proyecto se aplicaron conocimientos sobre bases de datos, PHP, HTML, funcionamiento de la tecnología RFID, programación del Arduino, y conexión con servidores en la nube.

El control de acceso al parqueadero permitirá registrar y validar que el acceso de los usuarios y las bicicletas correspondan a los datos registrados en la base de datos. El objetivo es realizar una relación entre el carné institucional y la etiqueta asignada a la bicicleta mediante la relación entre bases de datos, y así poder verificar que el propietario extraiga su bicicleta, llevar a cabo esta implementación permitirá fortalecer el sistema de control actual.

En este documento se presenta los requerimientos del sistema de control, en el cual se describen el microcontrolador y los módulos a usar, el diseño del sistema, en el cual se muestra el circuito usado en el prototipo y se describe la relación entre las bases de datos, después se procede con la explicación del procesamiento y comparación de la información capturada y a mostrar el funcionamiento del mismo, mediante algunas pruebas realizadas.

### **Planteamiento del problema**

En la ciudad de Bogotá el número de usuarios de bicicletas va en aumento día a día, según un estudio realizado por la facultad de ingeniería de la Universidad Libre (Revista Semana, 2018), esta cifra supera los 835.000 usuarios. Con el crecimiento de número de usuarios también se ha incrementado el número de hurtos reportados diariamente, en los dos primeros meses del año se ha presentado un incremento de 9% de casos de hurto con respecto al año 2018 (Vargas, 2019).

Este es un problema latente para los bici-usuarios, para solventar esta problemática los entes públicos deben buscar la manera de implementar un espacio seguro para guardar estos vehículos en sus instalaciones y ofrecer así un incentivo para el uso de medios de transporte alternativos. Los organismos públicos deben garantizar el desarrollo de un transporte sostenible en la ciudad como parte de responsabilidad social, para este desarrollo óptimo, las bicicletas son clave fundamental y se debe incentivar el uso de este medio de transporte por medio de la organización de espacios de estacionamiento para bicicletas seguros. (Olmos & Lopez, 2018)

La Universitaria Agustiniana ofrece un parqueadero de bicicletas, en el cual la seguridad depende de las guayas y candados puestos por los usuarios, que en casos proporcionan la seguridad necesaria, y el filtro que realizan los vigilantes en la entrada al parqueadero, filtro que puede resultar en ocasiones poco eficiente, porque al presentar el carné al ingreso, no existe una manera de verificar la veracidad de la información proporcionada por el usuario y mucho menos tener la certeza de que la bicicleta si sea de su propiedad.

Mediante la aplicación de las telecomunicaciones ¿Cuáles son las características que debe tener el sistema de control de acceso para mejorar la seguridad del parqueadero de bicicletas de la Universitaria Agustiniana?

### **Justificación**

Uno de los medios de transporte alternativos que más se ha popularizado a lo largo de los últimos años en Bogotá ha sido la bicicleta, razón por la cual el parqueadero de bicicletas se ha convertido en uno de los lugares más importantes para la planeación de movilidad realizada por cada organización. Mejorar este espacio estimularía el uso de medios de transporte alternativos, y el desarrollo de actividad física por parte de la comunidad de la Universitaria Agustiniana.

El motivo por el cual se inició con el planteamiento de este proyecto fue para apoyar y mejorar la seguridad que se ofrece en el parqueadero institucional de bicicletas, mediante la implementación de un sistema de control de acceso que compare y valide la información transmitida por las etiquetas (carné y la etiqueta ubicada en la bicicleta), verificando que la bicicleta sea de propiedad del usuario que presente las etiquetas. Este proyecto se desarrollará con la finalidad de aplicar los conocimientos adquiridos durante la carrera en beneficio de la comunidad universitaria, y de adquirir competencias tales como el manejo de bases de datos, implementación y uso de sistemas RFID, programación HTML y PHP, y gestión de proyectos, útiles para un buen desempeño profesional.

## **Objetivos**

### **Objetivo general**

Diseñar un sistema de control de seguridad en el parqueadero de bicicletas de una universidad, para prevenir que la bicicleta sea extraída por personas diferentes al propietario.

### **Objetivos específicos**

Identificar los requerimientos del sistema de control de seguridad para determinar cuál es la tecnología adecuada a usar.

Diseñar un sistema de control de seguridad que valide y compare la información obtenida de tags y la información registrada en la base de datos.

Realizar pruebas de funcionamiento del sistema de control de seguridad, para validar la efectividad del mismo.



## Marco referencial

### Estado del arte

A medida que evolucionan las tecnologías, también van evolucionando las formas de identificación en las entradas de lugares públicos o privados, con la finalidad de tener control de acceso por cuestiones de seguridad y para garantizar el acceso oportuno a la información. Debido a la evolución de las necesidades, las instituciones educativas se deben adaptar a ello, en el caso de la Universidad distrital Francisco José de Caldas, en una tesis de grado se plantea la implementación de un sistema de control de acceso en la facultad de ingeniería mediante el uso de torniquetes controlados por carnet con tecnología NFC. (Balsero & Vargas, 2016)

En este sistema de control usaron tarjetas RFID que operan a 13.56MHz, las tarjetas son los elementos que proporcionarán la identificación, esta está basada en el estándar ISO 15693, el lector utilizado fue el Wall Plate Reader, el cual permite leer hasta 10 tags de manera simultaneas. El lector de RFID opera en dos modos: automático, en el cual el lector realiza lecturas de manera periódica, y en el caso de modo por comando, el lector se encuentra en un estado pasivo, y solo realiza las lecturas cuando recibe un comando del host. (Balsero & Vargas, 2016)

En la Universidad Distrital, se planteó el diseño de un sistema de seguridad y registro por medio de acceso RFID para el parqueadero de bicicletas, con la finalidad de brindar al usuario un servicio de seguridad, mediante el diseño de un dispositivo para contener el registro de cada persona, realizando una interacción entre el usuario y máquina, para verificar la propiedad de cada usuario (Garzon & Rodriguez, 2017)

Los modos de funcionamiento pueden ser de dos modos: pasivo y activo. En el modo pasivo, el dispositivo destino se comunica con este modulando la señal recibida, en el que el dispositivo destino obtiene la energía necesaria para funcionar del campo electromagnético generado por el iniciador. En el caso del modo activo, ambos dispositivos necesitan de una fuente de alimentación para funcionar, en este modo los nodos negocian el intercambio de datos (Chulde & Méndez, 2014)

Al realizar la comparación entre las etiquetas NFC y RFID de estos, se puede evidenciar que las frecuencias en las que se trabajan son las mismas en las cuales en este caso se podría manejar un mismo lector para las dos etiquetas, en dichos proyectos. En el caso de la Universitaria Agustiniiana, los lectores ubicados en el parqueadero detectan señales de 135KHz, los cuales solo detectarían los carnés, en el caso de implementar una etiqueta para identificar cada bicicleta sería

necesario implementar un lector diferente ya que las etiquetas necesarias para este sistema de control trabajan a una frecuencia de 13.56MHz.

Para la implementación de las etiquetas NFC es necesario tener en cuenta la información a guardar, para seleccionar la etiqueta adecuada y según las especificaciones de estas, saber cuáles son las más convenientes en cuanto a velocidad de transmisión. También, hay que tener en cuenta las características necesarias, como su capacidad de memoria y la seguridad con relación al proyecto a ejecutar para que el sistema de control sea efectivo. (Niño, 2016)

Los sistemas de asistencia deben evolucionar a sistemas sofisticados, inteligentes y automatizados. El arduino uno digitaliza las señales analógicas obtenidas de los sensores utilizados para vigilar la recepción de las señales de las etiquetas y tarjetas, esta información registrada se almacena en las bases de datos, para ser gestionados posteriormente a través de un programa informático, para su consulta futura. Este sistema usa para la base de datos el software Eclipse Oxygen.3a, esta estará interconectada al IDE de arduino. Eclipse Oxygen.3a estará interconectado con el arduino, el software se conectará a Microsoft Excel para obtener información de los datos entregados para llenar la hoja de asistencia. El lector RFID actúa como una entrada, la cual recibe información de la etiqueta y luego pasa la información al arduino. (Bakht, Ud Din, Shehzadi, & Aftab, 2019)

En la universidad Hangzhou Dianzi se planteó un proyecto de investigación enfocado en el desarrollo de un sistema de control de acceso para oficinas aplicando la tecnología RFID, en el cual proponen el diseño de un sistema con control del cierre automático de la cerradura electromagnética, el cual está conformado por un lector de las etiquetas, un magnetismo de la puerta, una cerradura electromagnética y el sistema de alarma. Cuando se acerca la etiqueta al rango de lectura-escritura, la tarjeta RF transmite la información al controlador de acceso a través del lector de tarjetas, este verificará la validez y veracidad de la información para posteriormente habilitar o denegar la apertura de la puerta. Los datos resultantes del procesamiento de datos registrados se guardarán en una base de datos SQL serve para su futura consulta, y para tener disponibilidad en cualquier momento. (Wang & Wang, 2018)

Durante el desarrollo de este sistema de control de acceso para las oficinas fue necesario definir los parámetros de configuración del sistema, esto sirvió para tener en cuenta la información que se debe almacenar en las bases de datos para realizar la validación de identidad, todo lo anterior con

el fin de incrementar la seguridad del sistema de control de acceso de la oficina. (Wang & Wang, 2018)

Desarrollar un sistema de control de acceso NFC es muy similar al sistema RFID, ya que la tecnología NFC es compatible con la infraestructura RFID. Utilizando un arduino permite la conectividad ethernet con otros equipos y la conexión a módulos NFC, el arduino utiliza los puertos digitales para mantener cerrados los electroimanes ubicados en la puerta, cuando se valide el acceso estos actuadores habilitarán la apertura de la puerta. En caso de presentar una falla en el suministro eléctrico, el sistema tendrá un sistema de respaldo de batería, lo que permitirá desactivar el sistema y habilitar el sistema manual. Este sistema de control mejorará la seguridad, y la integración de los sistemas de gestión y de control del personal, mediante automatización del procedimiento de control de asistencia. (Díaz, 2015)

### **Marco teórico**

#### **Etiquetas RFID.**

Es una tecnología basada en ondas de radio para identificar y detectar objetos, mediante el uso de etiquetas que permiten la lectura y escritura de información, compuestas por un microchip y una antena. Un sistema RFID consiste en tres componentes que son: etiquetas RFID, lectores o readers y software de enlace. Las etiquetas RFID, son elementos capaces de almacenar y transmitir información, compuestos por un microchip y una antena en el cual posee un código único de información, poseen una variedad de tamaños y formas diseñadas según el uso, según el tipo de etiqueta varía el rango de distancia de lectura, la capacidad de almacenamiento de datos, velocidad de transferencia de datos y el tiempo de vida útil, las etiquetas más conocidas son: las pasivas, activas y semi-pasivas.

Las pasivas son las etiquetas que adquieren mediante la potencia de transición emitida por las antenas de lectura, su activación se realiza mediante la aproximación a la cobertura de la antena lectora. A diferencia de las pasivas, las etiquetas activas poseen una alimentación propia mediante una pequeña batería, que permite tener una comunicación con la antena lectora a mayores distancias permitiendo el proceso de mayor cantidad de datos. En el caso de las etiquetas semi-pasivas, poseen una batería, pero esta es usada solo para la alimentación para el microchip. (Romero, 2019)

### **Near Field Communication (NFC).**

Esta tecnología ofrece nuevas funcionalidades, ya que se puede combinar una etiqueta NFC con un lector RFID, lo que permite una comunicación bidireccional entre dos dispositivos, rompiendo así la separación funcional entre el lector y la etiqueta de este tipo. La comunicación de campo cercano (NFC), que utiliza los principios de la tecnología RFID, la cual ofrece una comunicación inalámbrica a corto alcance y elevada frecuencia 13.56 MHz, permitiendo el intercambio entre periféricas hasta una distancia de aproximadamente de 110cm. La velocidad de transmisión de datos puede variar entre 106kbps, 202 kbps, y 424kbps, este protocolo NFC cubre los modos de operación activo y pasivo. Un dispositivo NFC puede comunicarse con cualquier tarjeta inteligente y lector. (Chulde & Méndez, 2014)

### **Frecuencia de operación para RFID.**

Para que el sistema RFID logre tener una comunicación, es necesario que la frecuencia de trabajo de las etiquetas y de los lectores sean las mismas, esto condiciona las características físicas de propagación del campo electromagnético, de transmisión de datos y sensibilidad de los materiales. Según la frecuencia de trabajo varían las aplicaciones usuales, existen cuatro frecuencias de trabajo que son: frecuencia baja (LF) que trabaja en una frecuencia de 125KHz, frecuencia alta (HF) que opera en una frecuencia de 13.56MHz, ultra alta frecuencia (UHF) que trabaja en un rango de frecuencia de 860MHz y 960MHz, y las microondas que operan en un rango de frecuencias de 2.4GHz y 5.8GHz. Para una aplicación de sistema de control las frecuencias de trabajo LF y HF son las que se acoplan a este requerimiento.

El acoplamiento entre el lector y la etiqueta depende del rango de frecuencias utilizadas en la transmisión. Este acoplamiento puede ser inductivo que se usa para comunicaciones de baja frecuencia y alta frecuencia, este funciona de tal modo que la corriente eléctrica que circula por la antena del lector genera un campo electromagnético y al acercarse a la antena de la etiqueta induce la corriente para alimentar la etiqueta, que posteriormente modula la corriente de carga para que se permita la transmisión efectiva de los datos. El otro tipo de acoplamiento es el capacitivo, que es utilizado para la comunicación de frecuencias UHF (Ultra High Frequency) y microondas. Su funcionamiento es mediante la emisión por parte del lector de ondas de radio que la etiqueta recibe, modula y refleja al lector de nuevo. (Tami, Ballesteros, & Fino, 2015)

### **Arduino.**

La placa arduino se basa en wiring y el entorno de desarrollo Arduino basado en processing. Wiring es una plataforma abierta para la elaboración de dispositivos o prototipos electrónicos compuestos en el entorno de programación. El software que permite enseñar los fundamentos de programación basándose en un lenguaje de código abierto, orientado para los usuarios con la necesidad de generar trabajos en el ámbito profesional, este entorno llamado processing. Una de las principales características de las placas arduino son que poseen una memoria RAM de 256MB, una velocidad de reloj de 16MHz, cuenta con hasta 54 terminales de entrada/salida, número de terminales que varía según el módulo a escoger, para decidir cuál es el módulo adecuado a usar, se debe tener en cuenta las características del proyecto a desarrollar. (Haro Flores, 2016)

### **Marco legal**

#### **Ley 1581 de 2012.**

En este proyecto será necesario la recolección y comparación de información personal de cada usuario del parqueadero de bicicletas, por lo que es necesario tener en cuenta la ley 1581 de 2012, ley en la que se constituye el marco general de la protección de datos personales en Colombia.

*Artículo 1º. Objeto.* La presente ley tiene por objeto desarrollar el derecho constitucional que tienen todas las personas a conocer, actualizar y rectificar las informaciones que se hayan recogido sobre ellas en bases de datos o archivos, y los demás derechos, libertades y garantías constitucionales a que se refiere el artículo 15 de la Constitución Política; así como el derecho a la información consagrado en el artículo 20 de la misma. (Congreso de la República, 2012, pág. 1).

*Artículo 9º. Autorización del Titular.* Sin perjuicio de las excepciones previstas en la ley, en el Tratamiento se requiere la autorización previa e informada del Titular, la cual deberá ser obtenida por cualquier medio que pueda ser objeto de consulta posterior. (Congreso de la República, 2012, pág. 5).

#### **Ley 1341 de 2009.**

El artículo 11 de la ley 1341 de 2009, está relacionado con el permiso de uso del espectro radioeléctrico y del establecimiento de la banda de frecuencia de uso libre. En la resolución 711 de 2016, se establece las bandas de frecuencia de uso libre, en el cual se menciona los dispositivos de identificación por radiofrecuencia.

**Artículo 11º- Acceso al uso del espectro radioeléctrico.** El uso del espectro radioeléctrico requiere permiso previo, expreso y otorgado por el Ministerio de Tecnologías de la Información y las Comunicaciones.

El permiso de uso del espectro respetará la neutralidad en la tecnología siempre y cuando esté coordinado con las políticas del Ministerio Comunicaciones, no generen interferencias sobre otros servicios, sean compatibles con las tendencias internacionales del mercado, no afecten la seguridad nacional, y contribuyan al desarrollo sostenible. El Ministerio de Tecnologías de la Información y las Comunicaciones adelantará mecanismos de selección objetiva, previa convocatoria pública, para el otorgamiento del permiso para el uso del espectro radioeléctrico y exigirá las garantías correspondientes. En aquellos casos, en los que el nivel de ocupación de la banda y la suficiencia del recurso lo permitan, así como cuando prime la continuidad del servicio o la ampliación de la cobertura, el Ministerio podrá otorgar los permisos de uso del espectro de manera directa.

En la asignación de las frecuencias necesarias para la defensa y seguridad nacional, el Ministerio de Tecnologías de la Información y las Comunicaciones tendrá en cuenta las necesidades de los organismos de seguridad del Estado. El trámite, resultado e información relativa a la asignación de este tipo de frecuencias tiene carácter reservado. El Gobierno Nacional podrá establecer bandas de frecuencias de uso libre de acuerdo con las recomendaciones de la UIT, y bandas exentas del pago de contraprestaciones entre otras para programas sociales del Estado. (Congreso de Colombia, 2009)

#### **Resolución 711 de 2016.**

**Artículo 2º- Bandas de frecuencia de libre utilización.** Serán de libre utilización en todo el territorio nacional las bandas de frecuencia establecidas en el Anexo de la presente resolución bajo las condiciones determinadas en el mismo. (Agencia Nacional del Espectro, 2016)

**Anexo 2.2. Excepciones o exclusiones de los límites generales.** En esta sección se han establecido disposiciones especiales para ciertos tipos de transmisores que requieren intensidades de señal superiores a las proporcionadas por los límites generales de emisión radiada establecidos en la Tabla de Límites generales para cualquier radiador intencional. (Agencia Nacional del Espectro, 2016)

Tabla 1.

*Límites generales para dispositivos RFID*

Dispositivos de identificación por radiofrecuencia - RFID 60 dB $\mu$ A/m a 10 m		
Banda (MHz)	Aplicación	Límite de potencia o intensidad de campo
13.567-13.71	Otros dispositivos no específicos de corto alcance	334 $\mu$ V/m a 30 m
13.71-14.01	Otros dispositivos no específicos de corto alcance	106 $\mu$ V/m a 30 m
26.957-27.283	Otros dispositivos no específicos de corto alcance	42 dB $\mu$ A/m a 10 m

*Nota.* Agencia Nacional de Espectro (2016).

### **Metodología de la investigación**

La investigación cuantitativa es un conjunto de procesos sistemáticos y empíricos, aplicados al estudio de un problema o planteamiento especificado y delimitado desde un principio, basado en investigaciones previas. Este tipo de investigación es un proceso secuencial y probatorio, en el cual cada etapa precede a la siguiente, por lo que no es posible saltar pasos. Utiliza la recolección de datos para probar hipótesis, mediante la medición matemática y el análisis estadístico, para formular y demostrar teorías. Este tipo de investigación debe ser objetiva, para que los factores personales del investigador no influyan en el procedimiento o conclusiones de la investigación. (Hernández Sampieri, 2014)

Este proyecto se realizará desde un enfoque cuantitativo, ya que, desde el origen de la idea de proyecto, se delimitó y concretó que la problemática es la seguridad de las bicicletas en el parqueadero de la Universitaria Agustiniense sede Tagaste. En base de la idea de proyecto se definirán las variables tales como los requerimientos del sistema de control de acceso, el método de procesamiento y comparación de información de los usuarios, entre otras variables que se tendrán en cuenta para la implementación del sistema de control de acceso al parqueadero. Después se procederá con la realización de pruebas y recolección de información, para posteriormente analizar los resultados, y mediante evidencia de las pruebas realizadas determinar si se cumple con los factores determinados en un principio.



## Cronograma

Tabla 2.

*Cronograma de Actividades*

Actividades	Mes 1				Mes 2				Mes 3				Mes 4			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Investigación de parámetros del sistema actual de control de acceso de los parqueaderos																
Identificación de parámetros para el sistema de control de acceso																
Consulta de características de diversos métodos de procesamiento de información																
Definición del modo de procesamiento y comparación de información																
Elaboración de prototipo del sistema de control de acceso																
Recolección de información de los usuarios																
Creación de base de datos																
Desarrollo de pruebas y revisión del cumplimiento de los parámetros																

*Nota.* Autoría propia

## Presupuesto

Tabla 3.

### *Recursos Humanos*

Perfil	Justificación	Cantidad	Valor
Investigador	Persona encargada del desarrollo del trabajo	1	\$ 4.880.000
TOTAL			\$ 4.880.000

*Nota.* Autoría propia

Tabla 4.

### *Presupuesto de Equipos*

Equipo	Justificación	Cantidad	Valor
Modulo Lector Arduino	Dispositivo encargado de recibir la información transmitida por las etiquetas	1	\$15.000
Etiquetas RFID	Elementos capaces de almacenar y transmitir información	50	\$1.500 x Unidad
Arduino	Microcontrolador	1	\$85.000
Módulo ENC28J60	Permite la conexión del Arduino a la red Ethernet	1	\$15.000
PC	Desarrollo de pruebas y visualización de datos	1	\$1'600.000
TOTAL			\$1'790.000

*Nota.* Autoría propia

Tabla 5.

### *Presupuesto de Software*

Software	Justificación	Valor
Base de datos	Almacenamiento de datos de los usuarios y sus bicicletas	\$0
Arduino IDE	Entorno de programación para crear aplicaciones a las placas Arduino	\$0
Servidor en la nube	Infraestructura virtual que almacena el procesamiento de información	\$0
TOTAL		\$0

*Nota.* Autoría propia

Tabla 6.

*Presupuesto Final*

ITEM	TOTAL
Recursos Humanos	\$ 4'880.000
Equipos	\$ 1'790.000
Software	\$ 0
TOTAL	\$ 6'670.000

*Nota.* Autoría propia

### Requerimientos del sistema de control de acceso

Para el funcionamiento del sistema de control de acceso es necesario cumplir con ciertos requerimientos. Se determina el módulo lector a usar, según la frecuencia de operación usada por parte de los carnés institucionales, para procesar las lecturas tomadas, será necesario contar con un microcontrolador que tenga la capacidad de tener comunicación con el lector. Para que el microcontrolador pueda enviar los datos de lectura al servidor web, se tendrá que usar un módulo que permita la conexión a internet y así realizar las consultas en la base de datos alojadas en dicho servidor.

Se deben relacionar los datos del usuario con los datos de las bicicletas mediante una base de datos, por lo que en la entrada del parqueadero se deberá contar con una talanquera para autorizar la entrada al parqueadero, y un dispositivo con acceso a internet para realizar el registro de los usuarios nuevos y la consulta de los datos registrados en caso de ser necesario, en la página web que permite la consulta en la base de datos.

Se asume que la ocupación máxima del parqueadero de la Universitaria Agustiniiana es de doscientas bicicletas, por lo cual se hizo una aproximación de los posibles usuarios que usan el parqueadero durante un día de alto flujo como sucede entre semana (lunes a viernes). En la tabla 7, se muestra el número de bicicletas que entran y salen durante diferentes rangos de horas durante el día, y el número de bicicletas que ocupan el parqueadero durante estos rangos.

Tabla 7.

*Entorno estimado de ocupación del parqueadero de bicicletas*

Horario	Entrada Bicicletas	Salida Bicicletas	Bicicletas en el parqueadero
06:30 a.m. - 09:30 a.m.	200	0	200
09:30 a.m. - 11:30 a.m.	30	150	80
11:30 a.m. - 2:30 p.m.	20	70	30
02:30 p.m. - 05:30 p.m.	20	30	20
05:30 p.m. - 08:30 p.m.	200	90	130
08:30 p.m. - 10:00 p.m.	30	160	0

*Nota.* Autoría propia

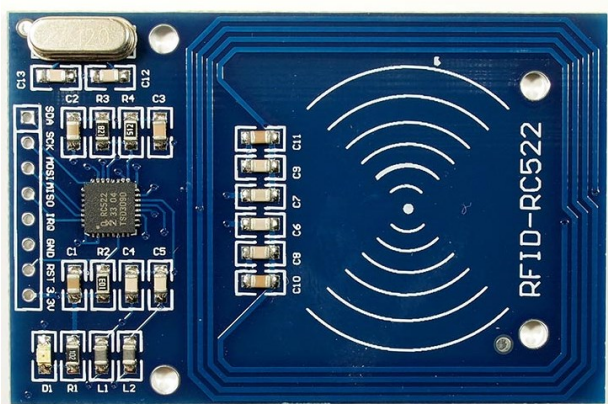
Teniendo en cuenta la estimación anterior, se determinará que para la creación de la base de datos se deberá registrar aproximadamente quinientos datos de bicicletas, debidamente relacionados con el ID correspondiente a cada carné institucional.

## Módulos lectores

Debido a que los carnés de identificación personal de la Universitaria Agustiniana tienen una frecuencia de operación de 13.56MHz, es necesario tener en cuenta que los lectores RFID cuenten con esta misma frecuencia de trabajo, por lo cual se muestran a continuación, dos módulos lectores que cumplen esta característica.

### Módulo Lector RC522.

El módulo RC522 es un chip de tarjeta de comunicación sin contacto de lectura y escritura integrado, trabaja bajo el concepto avanzado de modulación y demodulación completamente integrado con todos los tipos de métodos y protocolos de comunicación sin contacto pasivos de métodos y protocolos de comunicación sin contacto pasivo de 13.56MHz. Soporta la serie de comunicación sin contacto de alta velocidad, con una velocidad de transmisión bidireccional de datos de hasta 424Kb/s. El módulo RC522 mostrado en la figura 1, tiene como características un rango de lectura de 0 a 0.6cm, y comunicación por SPI, lo que posibilita la implementación con cualquier microcontrolador con interfaz SPI (All Electronics, 2020).



**Figura 1.** Módulo Lector RC522. T Haoyu Electronics. (2020)

### Módulo RFID/NFC PN532.

El módulo lector PN532 opera en una frecuencia de trabajo de 13.56 MHz, permite la comunicación SPI (bus de interfaz de periféricos), I2C y UART (Transmisor-Receptor Asíncrono Universal), tiene una transferencia de datos máxima de 10Mbits/s. Este módulo permite la comunicación bidireccional con los teléfonos celulares que cuentan con la tecnología NFC, dando la posibilidad de realizar implementaciones diversas (Electronilab, 2020). En la figura 2 se muestra el módulo descrito anteriormente.



**Figura 2.** Módulo Lector RFID/NFC PN532. Electronilab. (2020)

En la tabla 8 se muestran los costos de los dos módulos lectores cotizados en la tienda Electronilab, en la cual se muestra que el módulo PN532 es más costoso en comparación al lector RC522. Se escoge el módulo más económico, aunque este cuente con una velocidad menor de transmisión respecto al PN532, soporta la necesaria para la lectura y transmisión de la misma hacia el microcontrolador usadas en el prototipo.

Tabla 8.

*Costos de los módulos lectores*

Módulo Lector	Costo
Lector RFID RC522	COP \$9,600
RFID/NFC PN532	COP \$69,700

*Nota.* Electronilab (2020).

## Microcontroladores

### Arduino uno.

El Arduino Uno es un microcontrolador basado en ATmega328P, este tiene comunicación SPI, lo que posibilita la comunicación con el módulo lector mencionado anteriormente. Las características de este microcontrolador se muestran en la tabla 9.

Tabla 9.

*Características Arduino Uno*

Microcontrolador	ATmega328P
Voltaje de funcionamiento	5V

Memoria flash	32 KB
SRAM	2 KB
EEPROM	1 KB
Velocidad del reloj	16 MHz

*Nota.* Arduino (2020).

### **PIC18F4550.**

El microcontrolador PIC18F4550 recibe los datos del lector por medio del protocolo SPI, este protocolo es un bus de tres líneas, que transmite los paquetes de información de 8 bits, dos líneas transfieren datos y la otra línea le corresponde al reloj. Estas características se muestran en la tabla 10.

Tabla 10.

*Características PIC18F4550*

Microcontrolador	PIC18F4550
Voltaje de funcionamiento	5V
Memoria flash	32KB
SRAM	2048
EEPROM	256 bytes
Velocidad del reloj	48MHz

*Nota.* Microchip (2020)

Se realiza la comparativa de estos dos microcontroladores, con la finalidad de conocer las características, y conocer su capacidad para los suplir los requerimientos necesarios para llevar a cabo la implementación del sistema de acceso. Para el diseño del prototipo se usará el microcontrolador Arduino, en caso de necesitar un mayor uso de memoria, será necesario analizar el uso de otro microcontrolador.

### **Módulo de conexión a internet**

Para determinar cuál es el módulo a usar, se determinó que la conexión a internet debe ser por medio guiado, debido a que los parqueaderos de bicicletas están ubicados en zonas en las cuales puede existir interferencia por su infraestructura, tales como los sótanos. Por lo cual se determinó

que debía ser por medio de un conector ethernet, a continuación, se muestran dos módulos con dichas características.

### **Módulo Ethernet ENC28J60.**

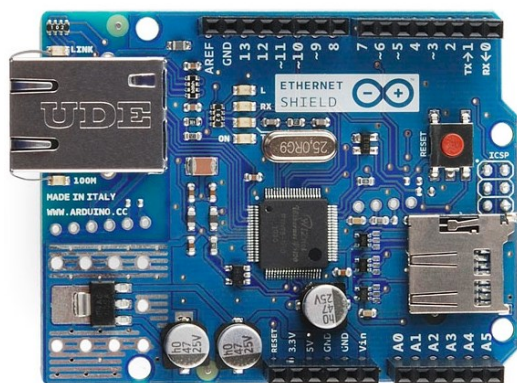
El módulo ENC28J60 proporciona un conector ethernet estándar RJ45 como se muestra en la figura 3, dando así la capacidad de conectar el Arduino a una red ethernet, suministrando una pila de red IP capaz de soportar TCP y UDP, usando la librería EtherCard para escribir los flujos de datos. La interfaz de comunicación SPI es compatible con sistemas digitales de 3.3V o 5V como Arduino, Teensy, Pic o Raspberry Pi (NaylampMechatronics, 2020).



**Figura 3.** Módulo Ethernet ENC28J60. NaylampMechatronic (2020).

### **Ethernet Shield W5100 Arduino.**

El Arduino Ethernet Shield proporciona un conector ethernet estándar RJ45, dando así la capacidad de conectar el Arduino a una red ethernet, suministrando una pila de red IP capaz de soportar TCP y UDP, usando la librería Ethernet para escribir los flujos de datos, el módulo se muestra en la figura 4. Arduino Uno usa los pines digitales 10, 11, 12 y 13 (SPI) para comunicarse con el Ethernet Shield (Aprendiendo Arduino, 2020)



**Figura 4.** Arduino Ethernet Shield (Aprendiendo Arduino, 2020).



Se escogió el módulo ENC28J60 debido a que es más económico respecto al W5100 como se evidencia en la tabla 11. Este módulo tiene la ventaja de permitir la conexión con microcontroladores diferentes al Arduino.

Tabla 11.

*Costos de los módulos de ethernet*

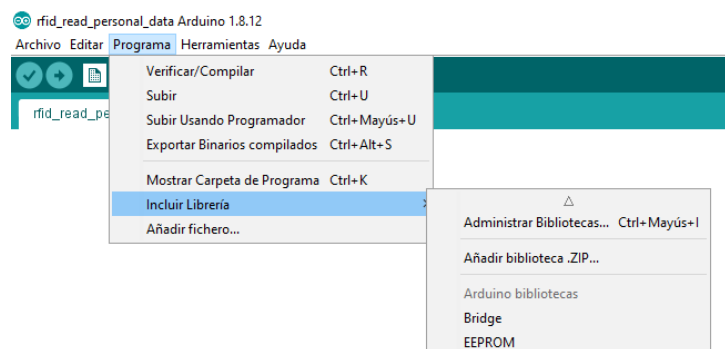
Módulo Ethernet	Costo
ENC28J60	COP \$16,000
Ethernet Shield W5100	COP \$23,000

*Nota.* Mercado Libre (2020).

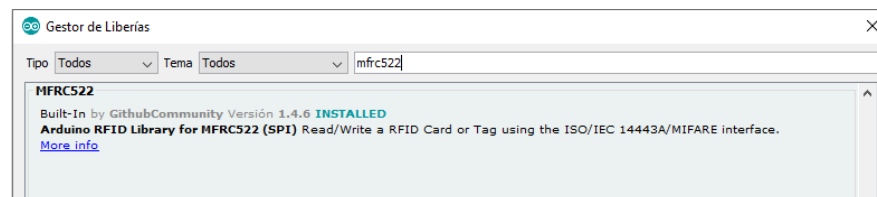
## Conexiones

### Código conexión Arduino – Modulo Lector RFID.

Para programar las instrucciones necesarias para la lectura de las etiquetas RFID es necesario incluir primero la librería “MFRC522”, dicha librería se puede encontrar en el IDE de Arduino en la opción Programa > Incluir Librería > Administrar Bibliotecas, después de seleccionar esta opción se desplegará el “Gestor de Bibliotecas” en la cual se buscará el nombre de la librería del módulo lector para proceder con la descarga de la librería como se muestra en las figuras 5 y 6.



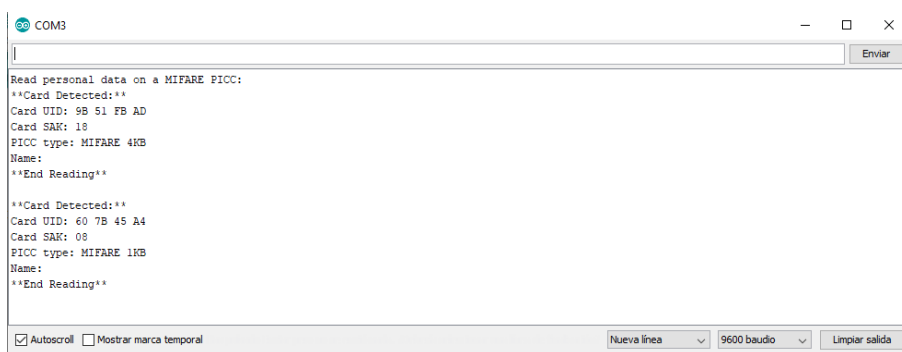
**Figura 5.** Inclusión Librería en Arduino IDE. Autoría propia.



**Figura 6.** Gestor de librerías en IDE Arduino. Autoría propia.

La librería “MFRC522” tiene ejemplos con las posibles aplicaciones del módulo lector, se puede encontrar el ejemplo “rfid\_read\_personal\_data”, con este ejemplo se puede dar las instrucciones al Arduino para realizar la lectura de la etiqueta y mostrar posteriormente en el “Monitor Serie” del IDE de Arduino los datos de la lectura.

Como se muestra en el monitor serie de la figura 7, se toma la lectura de dos etiquetas en la cual se muestra: la identificación única de la etiqueta (Card UID), el SAK que identifica el tipo de circuito integrado y la capacidad de la etiqueta, y el tipo de PICC (Proximity Integrated Circuit Chip) el cual muestra el tipo de MIFARE de la etiqueta. El MIFARE es la tecnología de tarjetas inteligentes sin contacto (TISC), son tarjetas de memoria protegida, divididas en sectores, bloques y mecanismos simples de seguridad para el control de acceso (NXP, 2020).



```

COM3
Read personal data on a MIFARE PICC:
**Card Detected:**
Card UID: 9B 51 FB AD
Card SAK: 18
PICC type: MIFARE 4KB
Name:
**End Reading**

**Card Detected:**
Card UID: 60 7B 45 A4
Card SAK: 08
PICC type: MIFARE 1KB
Name:
**End Reading**

Autoscroll [x]  Mostrar marca temporal [ ]  Nueva línea [v]  9600 baudio [v]  Limpiar salida [v]

```

**Figura 7.** Lectura de la Etiqueta en el Monitor Serie. Autoría propia.

En la figura 7 se muestra la lectura de dos etiquetas, la primera lectura es del carné de la universidad en la que se muestra que es de PICC tipo MIFARE 4K, que ofrece una memoria de 32 sectores con cuatro bloques cada uno y 8 sectores de 16 bloques, en el caso de la lectura de la segunda tarjeta, se muestra que es de tipo MIFARE 1K, que tiene una memoria de 4 bytes dividida en 16 sectores de 4 bloques de datos cada uno.

### **Conexión Lector RC552 – Arduino Uno.**

Como se mencionó anteriormente, el módulo lector se comunica por medio del protocolo SPI, protocolo usado también por arduino uno, en la tabla 12 se muestran las conexiones para que se logre la comunicación entre el módulo RC522 y el arduino uno, para la lectura de la etiqueta RFID y el procesamiento de la información.

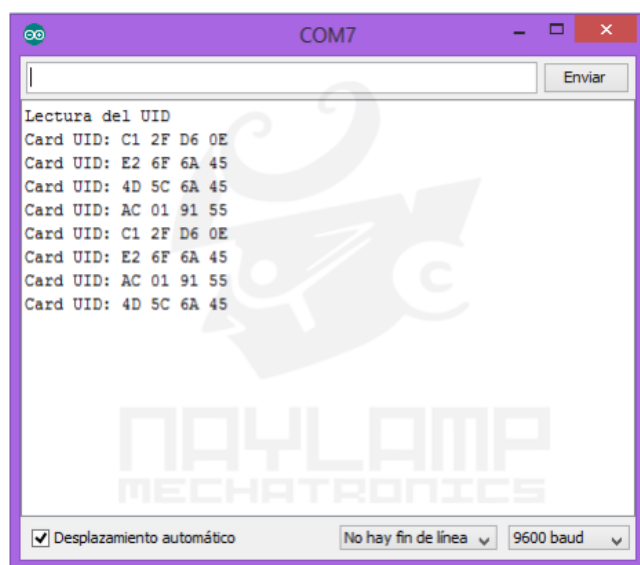
Tabla 12.

*Conexión Lector RC522- Arduino UNO.*

Módulo RC522	Arduino Uno
SDA	Pin 6
SCK	Pin 13
MOSI	Pin 11
MISO	Pin 12
IRQ	No conectado
GND	GND
RST	Pin 9
3,3V	3,3 V

*Nota.* NaylampMechatronics (2020)

En la figura 8 se muestra el código de identificación de la lectura a las etiquetas RFID, es enviado por medio del puerto serial del lector al arduino, este código se puede visualizar en el monitor serial del IDE de arduino, con este código de identificación se puede validar las etiquetas, verificando si pertenece a la base de datos a usar (NaylampMechatronics, 2020).



**Figura 8.** Código ID Monitor Serial del IDE. NaylampMechatronic (2020).

### **Conexión del Arduino - Módulo Ethernet ENC28J60.**

La conexión de pines entre el módulo ENC28J60 y el Arduino Uno se muestra en la tabla 10, se debe tomar una consideración especial con el pin de comunicación SPI, por lo cual solo se debe

usar este pin para la comunicación del módulo ENC28J60, en este caso el pin 13 es el usado para la comunicación SPI.

Tabla 13.

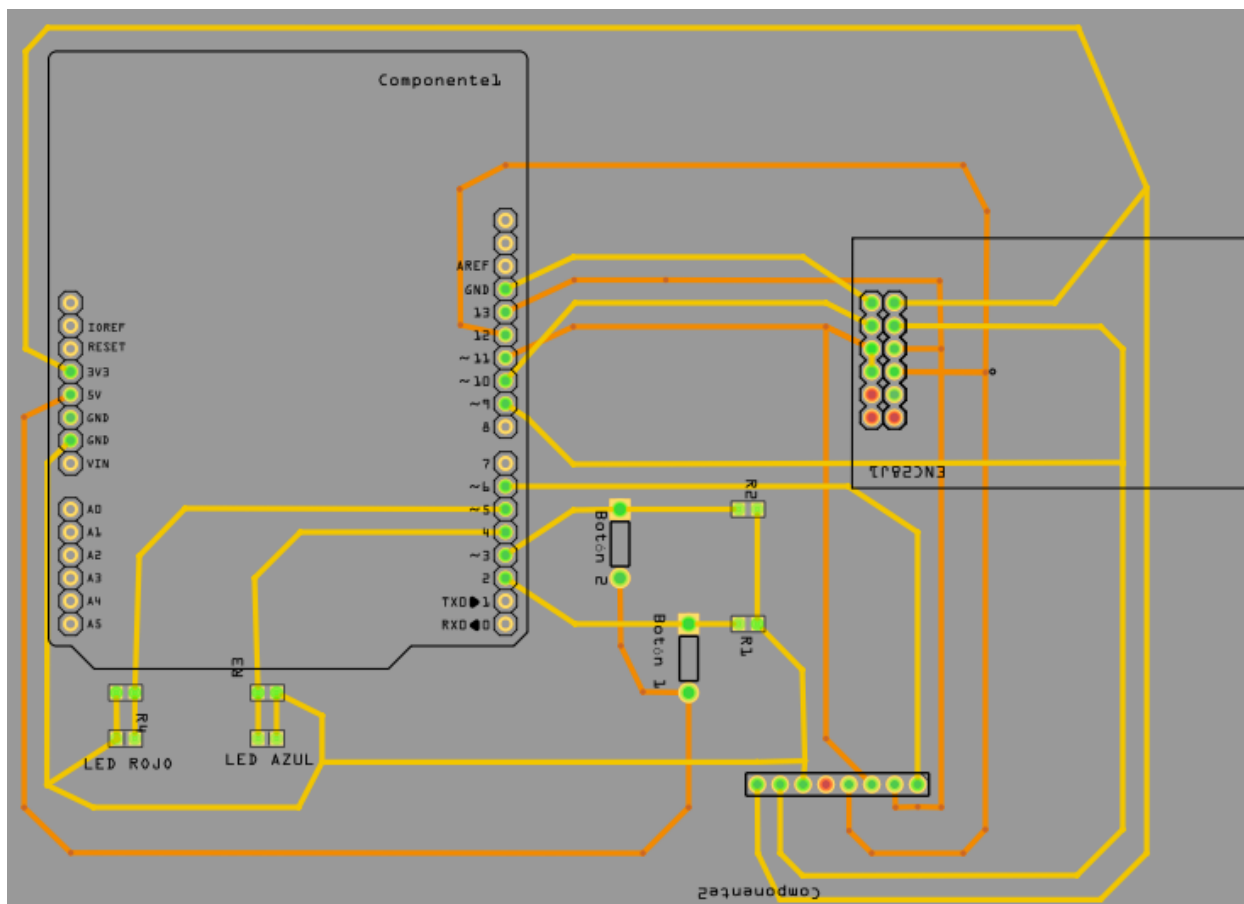
*Conexión Arduino UNO – ENC28J60.*

Módulo Ethernet ENC28J60	Arduino Uno
VCC	3.3V
GND	GND
SS (CS)	10
MOSI (SI)	11
MISO (SO)	12
SCK	13
RESET	RESET / 9

*Nota.* NaylampMechatronics (2020)

### Diseño del sistema de control de acceso

Para el sistema de control de acceso se usó el circuito PCB mostrado en la figura 8, en el cual se muestra que se usa dos botones, estos tienen la finalidad de diferenciar la lectura de la etiqueta institucional (carné) y de la etiqueta asignada a la bicicleta.



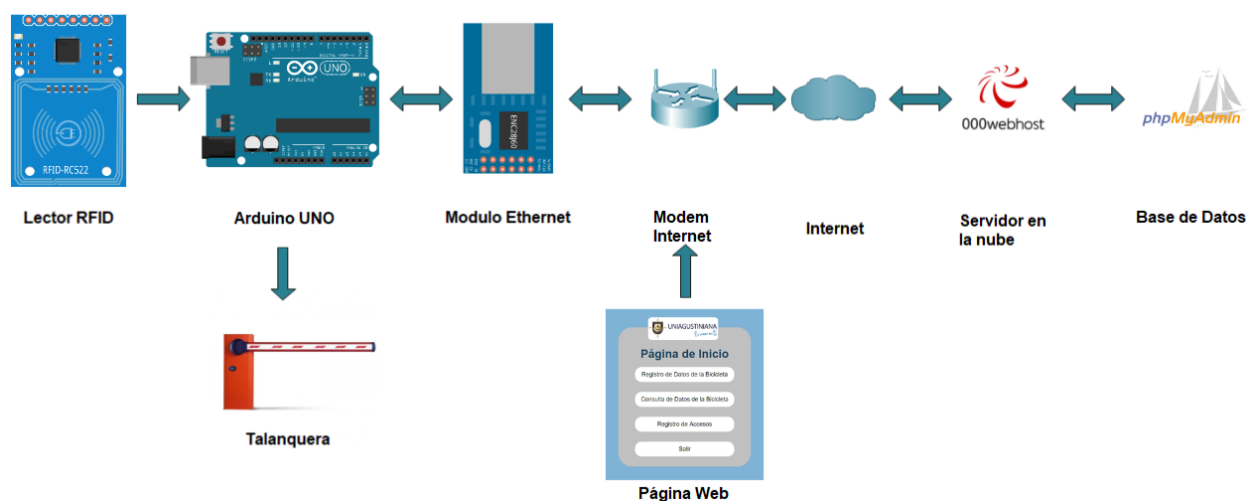
**Figura 9.** Circuito PCB Sistema Control de Acceso. Autoría propia.

Luego de tomar las lecturas correspondientes, se realiza la validación y comparación de las dos etiquetas, en la cual se determina si se autoriza la entrada mediante la activación de un LED color azul, y en el caso de que no sea autorizado se activará el LED color rojo.

En la figura 10, se muestra como el diagrama de elementos funcionales, en el cual se muestra la relación de cada elemento. El funcionamiento de esta red RFID inicia en el momento en el que se acerca la etiqueta RFID y el lector toma y envía la lectura al Arduino UNO, el cual se encarga de guardar dicha lectura en una variable y enviar la misma mediante una petición web al servidor 000webhost, para enviar dicha información, el Arduino debe tener conexión a internet, por lo que es necesario usar un módulo que lo permita.

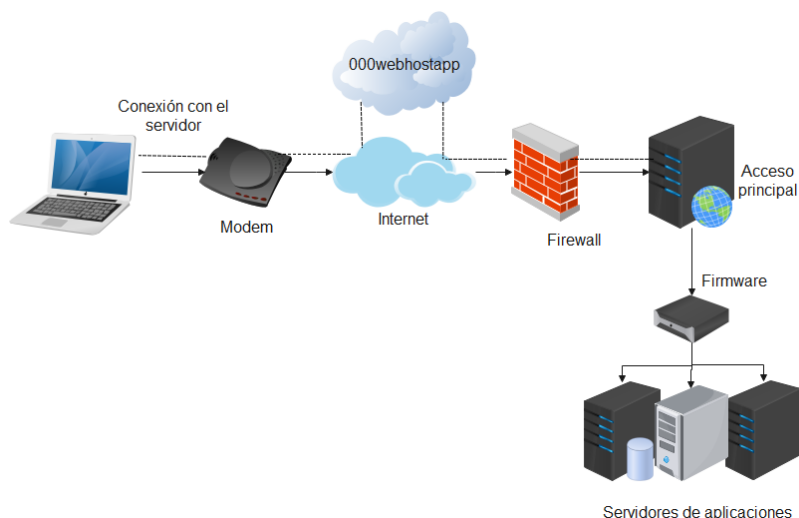
El Arduino envía el paquete de la petición web mediante el módulo de Ethernet gracias a la conexión a internet, el cual establece conexión con el servidor y entrega dicho paquete, el cual se recibe en el archivo PHP que realiza la consulta y validación de los datos enviados en la base de datos por el método GET. Luego de recibir la respuesta, los datos resultantes se reciben en el archivo PHP y se envían al Arduino para que realice la validación necesaria y determine si el ingreso se autoriza. En la figura 8 se indica que la respuesta se mostrará en la talanquera, en el caso de este prototipo, la talanquera se representará mediante leds, en el caso de que el ingreso no se autorice, se encenderá un LED rojo y un zumbador indicando que no coincide la etiqueta con las registradas en la base de datos, en el caso contrario se encenderá un LED azul, indicando que se autoriza el ingreso.

Para el registro y consultas de datos de las bicicletas, se debe realizar mediante la página web, la cual se conecta mediante código PHP al servidor en la nube y por consiguiente a la base de datos alojada en el servidor 000webhost.



**Figura 10.** Diagrama de elementos funcionales. Autoría propia.

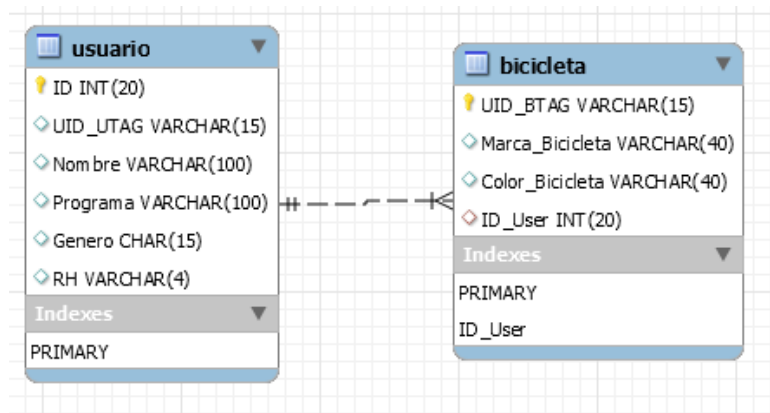
En la figura 11, se muestra el diagrama de red, en el que se evidencia las capas de acceso a la red, las cuales proporcionan una conexión segura al servidor ubicado en la nube (000webhostapp), en las que se ven involucradas las capas del modelo OSI, desde la capa física en los servidores de aplicaciones, y hasta en la capa de aplicación evidenciada en el dispositivo desde el que se realice la conexión al servidor.



**Figura 11.** Diagrama de red. Autoría propia.

### Base de datos

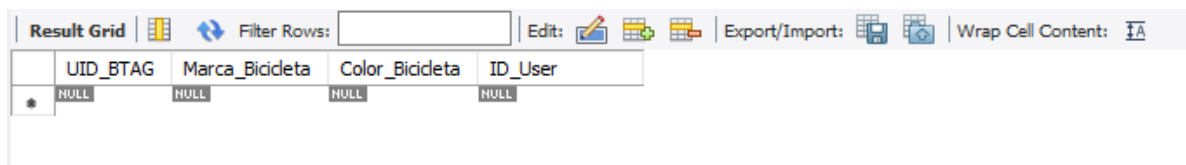
La base de datos será necesaria para la comparación y validación de datos obtenidos de las etiquetas (Carné de la institución y etiqueta ubicada en la bicicleta), esta se realizará en MYSQL. La tabla “Usuario” es de consulta, en la cual la llave primaria es el ID del carné institucional. En la tabla “Bicicleta” se guardará la información de la lectura de la etiqueta ubicada en la bicicleta, esta información será tomada como llave primaria, esta relación entre ambas tablas se muestra en la figura 12.



**Figura 12.** Diagrama de Bases de Datos. Autoría propia.

La base de datos del parqueadero de bicicletas se conecta con la base de datos de la universidad por medio de la llave primaria “ID” de la tabla “Usuario” y la llave secundaria “ID\_User” de la tabla “Bicicleta”, formando de esta manera una relación de uno a muchos, lo que permite el registro de más de una bicicleta para un mismo usuario del parqueadero en la base de datos.

Será necesario recolectar información relevante de la bicicleta, para llenar la tabla “Bicicleta” mostrada en la figura 13, en la que se incluirá el UID de la etiqueta que se le ubicara a la bicicleta, esta información es fundamental para la identificación de las etiquetas cuando se realicen las lecturas que tomará el módulo.



UID_BTAG	Marca_Bicicleta	Color_Bicicleta	ID_User
NULL	NULL	NULL	NULL

**Figura 13.** Tabla "Bicicleta". Autoría propia.

En el caso de la columna “UID\_BTAG” de la tabla “Bicicleta”, la información se debe importar del arduino, ya que este dato se obtiene por medio del lector, por lo que será necesario realizar una conexión entre MYSQL y el arduino. Para dicha conexión es necesario realizar peticiones HTTP a un servidor, la página a la que se realizará las peticiones estará escrita en PHP, este código recibirá las variables de consulta desde la URL y las enviará mediante una consulta SQL. Se debe realizar un código de conexión entre PHP y SQL, después de realizar dicha conexión, se enlaza y envía los datos almacenados en la memoria del arduino a MYSQL mediante un cliente web, que realice peticiones HTTP POST contra el servidor, para el manejo de los datos que se van a ingresar a la base de datos se puede hacer mediante phpMyAdmin (Rincon Ingenieril, 2020). En la figura 14 se muestra la interfaz de ingreso al base de datos.



**Figura 14.** Interfaz phpMyAdmin. Autoría propia.

Se desarrolla una aplicación web en PHP para tener acceso al registro, consulta de datos y registro de accesos en la base de datos “Bicicleta” como se muestra en la figura 15.





**Figura 15.** Página de Inicio Aplicación Web. Autoría propia.

Para añadir datos en la base de datos del parqueadero de bicicletas, se realizará por medio del acceso de “Registro de Datos de la bicicleta” el cual se dirigirá al formulario mostrado en la Figura 16. Ingreso de datos a la tabla “Bicicleta”

**Figura 16.** Ingreso de datos a la tabla “Bicicleta”. Autoría propia.

En la figura 17 se muestra los datos registrados desde la aplicación web y guardados en la tabla “Bicicleta” mediante una conexión PHP – MYSQL.

	UID_BTAG	Marca_Bicicleta	Color_Bicicleta	ID_User
	D5 86 DB D9	Shimano	Roja	1920111039
	F5 81 DB D9	GW	Negro	1920161021
	NULL	NULL	NULL	NULL

**Figura 17.** Registro de datos Tabla “Bicicleta”. Autoría propia.

Como se muestra en la pagina de inicio de la aplicación web, hay un boton de “Consulta de Datos de la Bicicleta” como se puede observar en la figura 18, el cual se podrá consultar mediante

el ingreso del ID del usuario, mostrando como resultado el UID de la etiqueta, la marca de la bicicleta, el color de la bicicleta y el ID del usuario del propietario de la bicicleta.

**Figura 18.** Consulta en la Base de Datos. Autoría propia.

En la figura 19 se muestra el resultado de una consulta realizada en la base de datos, en la cual se visualiza los datos de la bicicleta registrada.

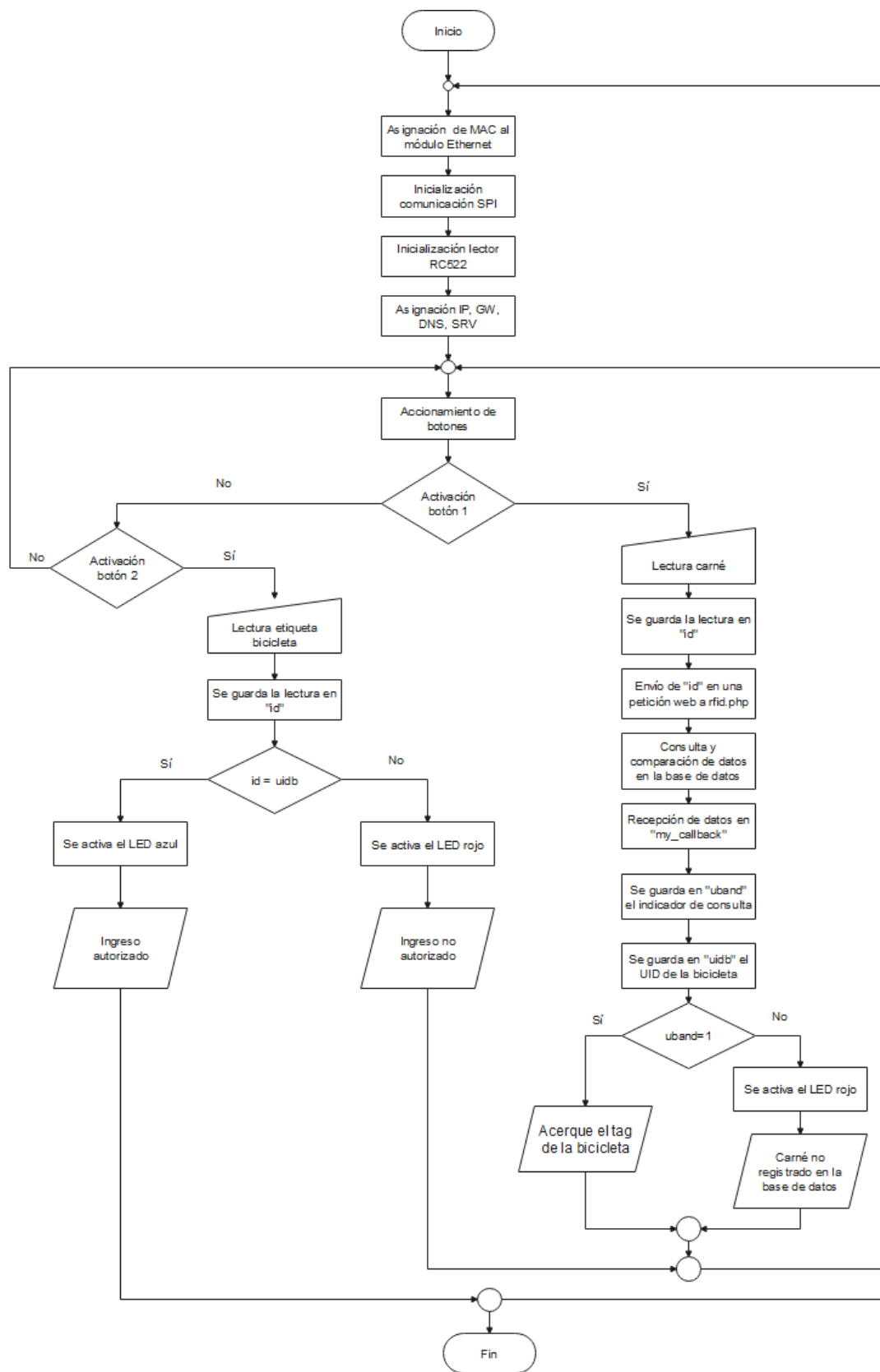
**Figura 19.** Resultado de la consulta. Autoría propia.

En la figura 20, se muestra el diagrama de flujo el cual explica el funcionamiento del sistema de control de acceso, en el cual primero se inicializan los módulos de ethernet y el lector, se inicia la comunicación SPI, que va a permitir la comunicación con el Arduino. Se debe asignar la IP, la puerta de enlace predeterminada, el DNS y el server, esto mediante DHCP. Luego de la asignación anterior, el usuario del sistema de control de seguridad se encargará de accionar un botón de lectura, en caso de que el botón accionado sea el botón uno, se procede con la toma y almacenamiento de la lectura del UID en la variable “id”, esto con el fin de enviar la variable posteriormente por medio de una petición web tipo GET hacía el servidor web, se recibe la variable

en el archivo “rfid.php” y se realiza la consulta en la base de datos, en dicha consulta se compara que el UID esté registrado en la base de datos y se consulta el UID de la etiqueta de la bicicleta relacionado con el carné institucional.

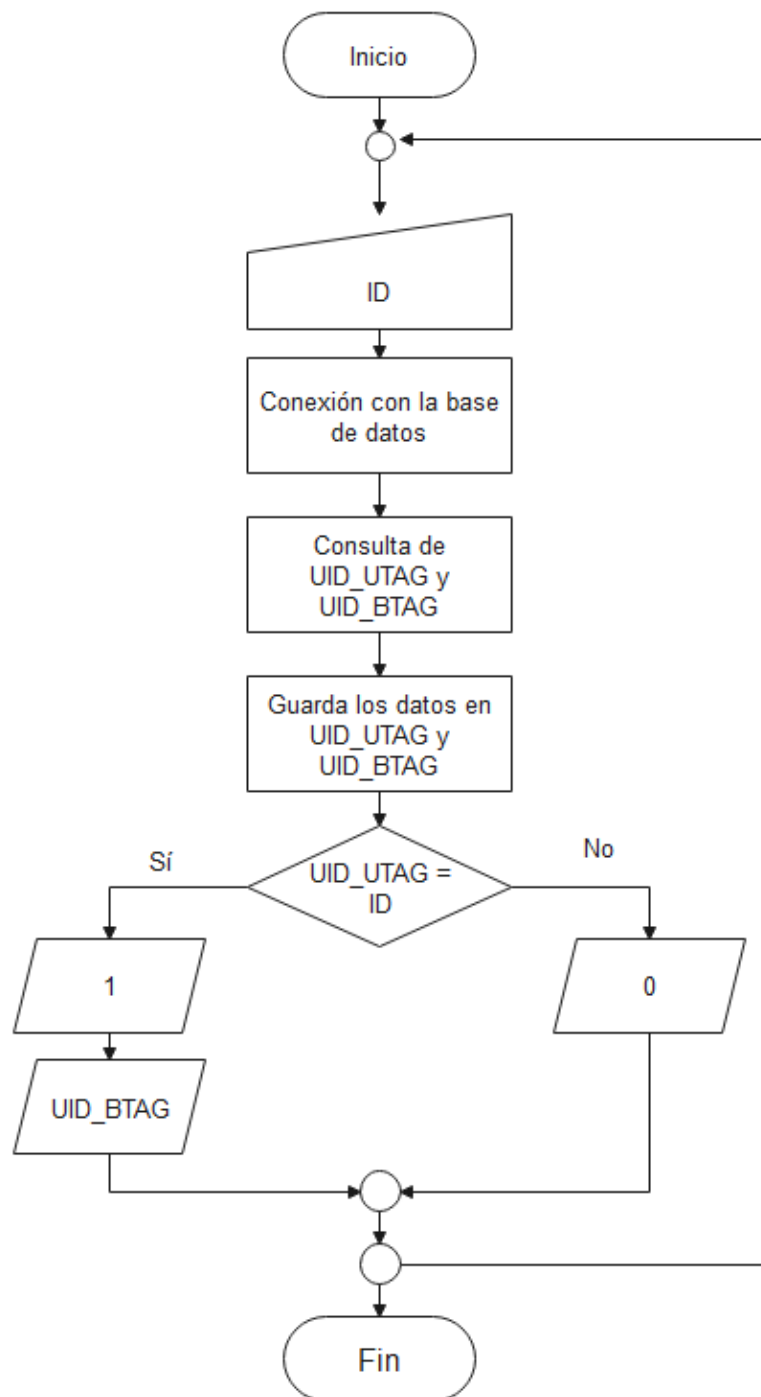
Estos datos generados en la consulta, se reciben en el Arduino por medio de la función “my\_callback”, esta se encarga de recibir la respuesta dada por el servidor en la nube, por lo cual se procede a guardar en la variable “uband”, la validación del registro del UID del carné, y en la variable “uidb” se guarda el UID correspondiente a la etiqueta de la bicicleta. Ya teniendo estas variables, se valida que la variable “uband” sea “1”, esto significa que la lectura tomada si está registrada en la base de datos, por lo cual se indica al usuario que acerque la etiqueta de la bicicleta, en caso de que el “uband” sea “0”, se activa el LED rojo y se indica por medio del monitor serial que el carné no está registrado en la base de datos.

Después de esto, se toma la lectura de la etiqueta asignada a la bicicleta mediante la activación del botón 2, esta lectura se guarda en la variable “id”, para posteriormente compararla con la variable “uidb”, en caso de que sean iguales se activa el LED azul y se indica que el ingreso se autoriza, en el caso contrario, se activa el LED rojo y el zumbador indicando que no se autoriza el ingreso.



**Figura 20.** Diagrama de flujo código Arduino. Autoría propia.

En la figura 21 se muestra el diagrama de flujo de la explicación del funcionamiento del código rfid.php, que se encarga de realizar la consulta en la base de datos. Esta consulta retorna a la función “my\_callback” y con dicha información se puede hacer la validación de la lectura del carné y de la lectura de la etiqueta asignada a la bicicleta que se tomará al accionar el botón dos, para verificar así, la relación existente entre el carné y la etiqueta de la bicicleta.



**Figura 21.** Diagrama de flujo código rfid.php. Autoría propia.

## Procesamiento y comparación de la información obtenida

Para realizar la comparación y validación de la información obtenida de las etiquetas RFID, es necesario validar primero que el módulo de Ethernet ENC28J60 tenga conexión a internet, esta validación se realiza mediante una solicitud HTTP como se muestra en el sketch de ejemplo “WebClient” correspondiente a la librería EtherCard, esta solicitud permite la comunicación con el hosting 000webhost, en la cual se envía un paquete y posteriormente se recibe un paquete de respuesta “HTTP/1.1 200 OK” en el que se indica que la comunicación con el servidor es exitosa como se muestra en la figura 22.

```
[webClient]
IP: 192.168.0.16
GW: 192.168.0.1
DNS: 190.157.8.101
SRV: 145.14.144.114

<<< REQ >>>
HTTP/1.1 200 OK
Date: Mon, 05 Oct 2020 00:03:33 GMT
Content-Type: text/html; charset=UTF-8
Connection: close
Server: awex
X-Xss-Protection: 1; mode=block
X-Content-Type-Options: nosniff
X-Request-ID: df5a5aacfafe426de0a319e1be39a84a
```

**Figura 22.** Conexión del módulo Ethernet a la página web. Autoría propia.

A continuación, se explicará el código usado para la programación del Arduino en el cual se validará la relación existente entre el carné institucional y la etiqueta RFID asignada a la bicicleta relacionada en la base de datos. El paso inicial en el código es importar las librerías a usar y definir los pines de comunicación SPI del lector RC522, como se muestra en la figura 23, la asignación de diferentes pines de comunicación SPI al módulo lector y al módulo Ethernet, permitirán el funcionamiento al mismo tiempo de los dos módulos. Después de la definición de los pines de comunicación, se le asignan estos pines mediante la creación de una instancia para el módulo lector RFID como en la figura 24, esto con el fin de que el módulo comience a funcionar.

```
#include <SPI.h>
#include <MFRC522.h>
#include <EtherCard.h>
#define SS_PIN 6
#define RST_PIN 9
```

**Figura 23.** Importación de librerías y definición de pines. Autoría propia.

```
MFRC522 mfrc522(SS_PIN, RST_PIN); // Create MFRC522 instance.
```

**Figura 24.** Creación de la instancia para el lector. Autoría propia.

Luego de la creación de la instancia para el funcionamiento del módulo lector RFID, se procede con el inicio del funcionamiento del módulo ethernet, en el cual se le asigna la dirección MAC al módulo y se define el sitio web con el que se va a generar la conexión con el servidor alojado en la dirección: “pruebacontrolacceso.000webhostapp.com”, como se muestra en la figura 25.

```
static byte mymac[] = { 0x00,0x25,0xAB,0x5C,0x86,0x5F };
byte Ethernet::buffer[700];
static uint32_t timer;
const char website[] PROGMEM = "pruebacontrolacceso.000webhostapp.com";
```

**Figura 25.** Inicialización Módulo Ethernet. Autoría propia.

En el setup del código se establece la velocidad del puerto serial, el inicio del protocolo SPI síncrono e iniciar el funcionamiento del módulo lector. Para iniciar el funcionamiento del módulo ethernet, se realiza la validación de la correcta asignación de la dirección IP, la dirección de la puerta de enlace predeterminada, y el DNS de la ruta de conexión a internet, como se muestra en la figura 26.

```
void setup()
{
  pinMode(led_aut, OUTPUT);
  pinMode(led_na, OUTPUT);
  Serial.begin(57600);
  SPI.begin(); // Initiate SPI bus
  lector.PCD_Init(); // Initiate MFRC522
  Serial.println("\n[webClient]");
  if (ether.begin(sizeof Ethernet::buffer, mymac) == 0)
    Serial.println( "Failed to access Ethernet controller");

  if (!ether.dhcpSetup())
    Serial.println("DHCP failed");
  ether.printIp("IP: ", ether.myip);
  ether.printIp("GW: ", ether.gwip);
  ether.printIp("DNS: ", ether.dnsip);
  if (!ether.dnsLookup(website))
    Serial.println("DNS failed");

  ether.printIp("SRV: ", ether.hisip);
  attachInterrupt(digitalPinToInterrupt(2), ISRA, RISING);
  attachInterrupt(digitalPinToInterrupt(3), ISRB, RISING);
  Serial.println("Oprima un botón");
}
```

**Figura 26.** Void setup. Autoría propia.

En el loop del código mostrado en la figura 27, se realiza la lectura de las etiquetas RFID, en esta parte del código se verifica si hay una tarjeta presente, en caso de que se devuelva un valor verdadero realiza la lectura “id”.

```

id = "";
// Si hay tarjeta nueva presente la leemos
if (lector.PICC_IsNewCardPresent() && lector.PICC_ReadCardSerial())
{
    // Mostramos el ID de la tarjeta leída
    //Serial.print("ID tarjeta: ");

    for (byte i = 0; i < lector.uid.size; i++)
    {
        if(lector.uid.uidByte[i] < 0x10) {
            Serial.print("0");
        }

        id += String(lector.uid.uidByte[i], HEX);
    }
    Serial.println(id);
    Serial.println();
    data0=id;
    lector.PICC_HaltA();
    delay(500);
}

```

**Figura 27.** Código del lector. Autoría propia.

Después de la toma de la lectura, se guarda el valor del UID de la etiqueta en la variable “data0”, esta será la variable de consulta que se enviará a la base de datos, mediante el método GET al archivo “rfid.php” del sitio web definido en el principio del código, como se observa en la última línea mostrada en la figura 28.

```

ether.packetLoop(ether.packetReceive());
data = data0;
char charBuf[70];
data.toCharArray(charBuf, 50);
if (millis() > timer) {
    timer = millis() + 5000;
    Serial.println();
    Serial.print("<<<< REQ ");
    ether.hisport = 80;//to access local host
    ether.browseUrl(PSTR("/rfid.php?id="), charBuf, website, my_callback);
}

```

**Figura 28.** Envío de información al servidor. Autoría propia.

El código PHP mostrado en la figura 29, recibe el valor enviado desde el Arduino por el método GET, en este código se realiza la conexión con la base de datos, para posteriormente ejecutar la consulta en las bases de datos “Usuarios” y “Bicicletas”, en el que se retornarán los valores del UID del carné y el UID de la etiqueta de la bicicleta. Este valor retorna al monitor serial del IDE del Arduino Uno.



```

/public_html/rfid.php
1 <?php
2 $id=$_GET["id"];
3
4 $mysql_server = "localhost";
5 $mysql_db = "id14881111_control_acceso";
6 $mysql_user = "id14881111_localhost";
7 $mysql_pass = "2u\>YP[|<E-NQ-%F";
8
9 $mysqli = new mysqli("localhost", "id14881111_localhost", "2u\>YP[|<E-NQ-%F", "id14881111_control_acceso");
10 $result = $mysqli->query("select usu.UID_UTAG, bici.UID_BTAG from usuario as usu join bicicleta as bici on usu.id=bici.id_user WHERE usu.UID_UTAG='".$id'");
11
12 if ($result) {
13 while ($row = $result->fetch_array()) {
14     $UID_UTAG = $row['UID_UTAG'];
15     $UID_BTAG = $row['UID_BTAG'];
16     echo $UID_BTAG;
17 }
18 }
19 $contar = mysqli_num_rows($result);
20 if($contar == 0){
21     echo "0";
22 }
23 else
24 {
25     echo "1";
26     $result1 = $mysqli->query("insert into RegistroUsuario (UID_USER, FECHA) values ('$id',NOW())");
27 }
28 ?>

```

**Figura 29.** Código GET “rfid.php”. Autoría propia.

La función “my\_callback” mostrada en la figura 30, se encarga el paquete de respuesta del web hosting, en el cual se muestra la conexión con el mismo y la respuesta dada a la consulta en la base de datos. Se guarda esta respuesta en la variable tipo cadena z, para posteriormente hacer un arreglo para guardar en variables globales los datos de la consulta para las posteriores comparaciones con las lecturas tomadas.

```

static void my_callback (byte status, word off, word len)
{
    Serial.println(">>>");
    String z = (const char*) Ethernet::buffer + off;
    Serial.println(">>>");
    Ethernet::buffer[off+300] = 0;
    uidb = z.substring(243, 251);
    Serial.println(uidb);
    uband = z.substring(len - 1);
    Serial.println(uband);
    delay(500);
    if(uband==check)
    {
        Serial.println("Acerque el tag de la bicicleta");
    }
    else{
        Serial.println("Carné no registrado en la base de datos");
        Serial.println("Ingreso no autorizado");
        digitalWrite(led_na, HIGH);
        delay(500);
        digitalWrite(led_na, LOW);
        Serial.println();
    }
    uband="";
    pboton=0;
    pboton2=0;
}

```

**Figura 30.** Función “my\_callback”. Autoría propia.

Con la finalidad de conocer cuál es el tiempo de conexión con el servidor, se realizaron algunas pruebas del funcionamiento del servidor usado para el desarrollo de la página web y de la base de

datos, para esto se realizaron estas pruebas en la página web “Web Page Performance”, en la que se puede visualizar el tipo de contenido, el tiempo de solicitud de inicio de 0.367 s, de conexión del DNS Lookup de 34 ms, de conexión inicial de 153 ms, de negociación SSL 172 ms y de la descarga de contenido 1 ms. En el código de estado, se obtiene como respuesta el código 200, esto quiere decir que la solicitud HTTP ha sido exitosa. En la tabla 14, se puede visualizar estos valores, según el recurso a usar.

Tabla 14.

*Detalles de la solicitud de la página web.*

Recursos	Tipo de contenido	Solicitud de inicio	DNS Lookup	Conexión inicial	Negociación SSL	Tiempo al primer byte	Descarga de contenido	Bytes descargados	Código de error / estado	IP
https://pruebacontrolacceso.000webhostapp.com/index.php	text/html	0.367 s	34 ms	153 ms	172 ms	166 ms	1 ms	0.4 KB	200	145.14.145.84
https://pruebacontrolacceso.000webhostapp.com/master-inicio.css	text/css	0.613 s	-	-	-	158 ms	1 ms	0.8 KB	200	145.14.145.84
https://pruebacontrolacceso.com/img/Captura.PNG	image/png	0.617 s	-	-	-	162 ms	9 ms	9.1 KB	200	145.14.145.84

*Nota.* Web Page Performance (2020)

### Resultados y pruebas del prototipo

Se realizó el registro de los datos de las bicicletas mediante la página web, como se muestra en la figura 31.

UNIAGUSTINIANA  
*Es crear en ti*

### Registro de Datos de Bicicletas

UID TAG Bicicleta  
49a957b3

Marca de la bicicleta  
Gw

Color de la bicicleta  
Rojo

ID del usuario  
6

Enviar

Borrar

Página Principal

**Figura 31.** Registro de datos de las bicicletas. Autoría propia.

En la figura 32 se muestra el mensaje en el cual se valida que el registro se realizó satisfactoriamente.



**Figura 32.** Registro satisfactorio. Autoría propia.

En la figura 33, se muestra la validación del registro de los datos de la bicicleta, mediante una consulta a la base de datos, ingresando el ID del usuario.

UNIAGUSTINIANA  
*Es crear en ti*

**ID: 6**

UID\_BTAG: 49a957b3

Marca\_Bicicleta: Gw

Color\_Bicicleta: Rojo

Página Principal

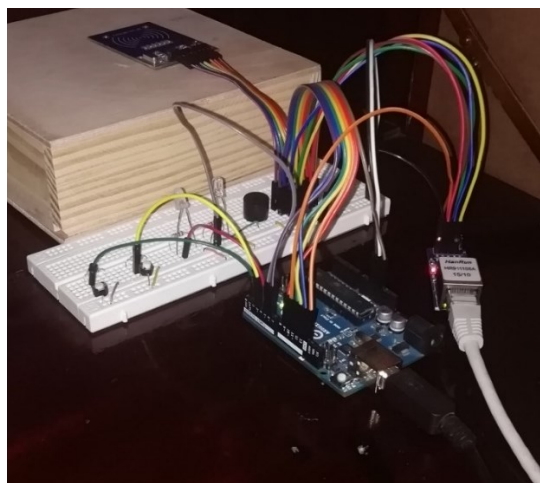
**Figura 33.** Validación del registro. Autoría propia.

En la figura 34 se muestran los UID del carné institucional y de la etiqueta de la bicicleta, relacionados con el ID del usuario, para las pruebas de funcionamiento se registraron siete relaciones de etiquetas RFID, las cuales se validarán posteriormente.

ID_User	UID_UTAG	UID_BTAG
1	b6bea2b	f22b4cd4
2	551bf2a	723260d4
3	b378a940	b2d8c3d4
4	d056e732	52a7a9d4
5	c3711f40	a2e1d8d4
6	607b45a4	49a957b3
1920161021	9b51fbad	f581dbd9

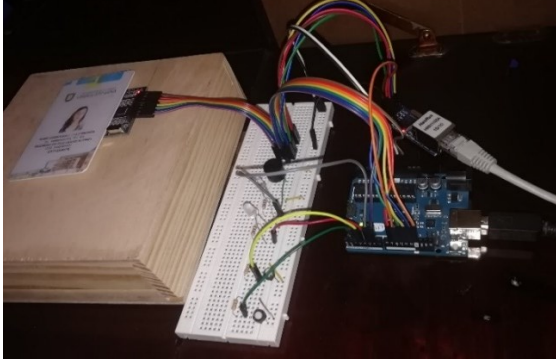
**Figura 34.** Relación de los UID según ID. Autoría propia.

Este procedimiento se realizó con las etiquetas usadas en las pruebas que se realizarán a continuación. Se realiza el montaje del prototipo para realizar las pruebas de funcionamiento, en la figura 35 se muestra dicho montaje. Para evidenciar el éxito de la comparación y validación se mostrará a continuación, algunas pruebas de funcionamiento.



**Figura 35.** Montaje del prototipo. Autoría propia.

En la figura 36 se muestra la primera toma de lectura de prueba en la cual se acciono el primer botón para tomar y guardar esta lectura, este carné institucional está registrado en la base de datos por lo cual permite la lectura de la etiqueta de la bicicleta como se muestra en la figura 37.



**Figura 36.** Lectura carné primera prueba. Autoría propia.

```
Oprima el botón 1
9b51fbad
```

```
<<< REQ 9b51fbad
>>>
>>>
f581dbd9
1
Acerque el tag de la bicicleta
```

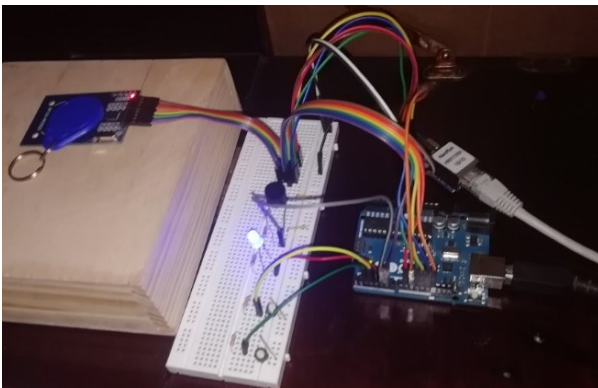
**Figura 37.** Monitor serial carné primera prueba. Autoría propia.

Se oprime el segundo botón para tomar la lectura de la etiqueta asignada a la bicicleta, esta se compara con la variable en la que se guardo el UID de la bicicleta extraido de la consulta anterior a la base de datos realizada en la lectura del carné como se observa en la figura 38. Se valida que la lectura y la variable sea la misma, y se indica que el ingreso se autoriza mediante la activación de un LED como se muestra en la figura 39.

```
f581dbd9
f581dbd9

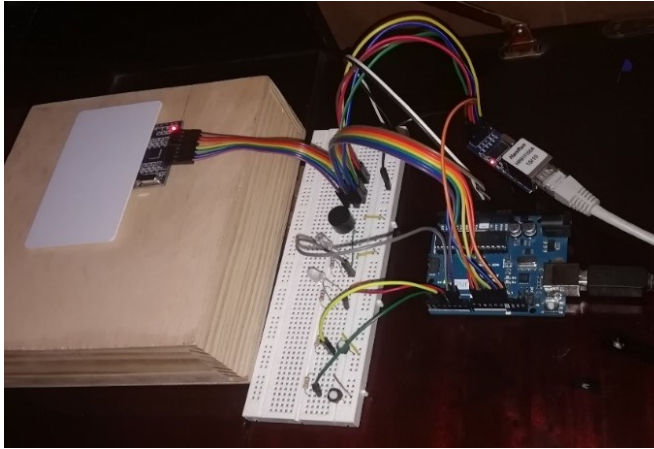
Ingreso autorizado
```

**Figura 38.** Monitor serial etiqueta bicicleta primera prueba. Autoría propia.



**Figura 39.** Lectura etiqueta bicicleta primera prueba. Autoría propia.

Se verifica el funcionamiento mediante una segunda prueba, en la figura 40 se muestra la segunda toma de lectura de prueba, esta etiqueta está registrada en la base de datos por lo cual permite la lectura de la etiqueta de la bicicleta como se muestra en la figura 41.



**Figura 40.** Lectura carné segunda prueba. Autoría propia.

607b45a4

```
<<< REQ 607b45a4
```

```
<<< REQ 607b45a4
```

```
>>>
```

```
>>>
```

```
49a957b3
```

```
1
```

```
Acerque el tag de la bicicleta
```

**Figura 41.** Monitor serial carné segunda prueba. Autoría propia.

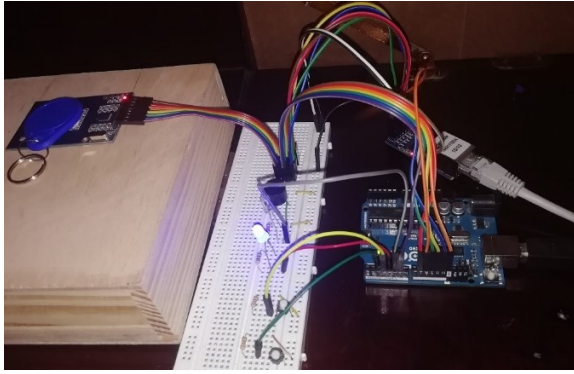
Como se realizó en la primera prueba, se toma la lectura de la etiqueta asignada a la bicicleta, esta se compara con la variable en la que se guardó el UID de la bicicleta como se observa en la figura 42, se valida la relación entre la variable y la lectura, y se autoriza mediante la activación de un LED como se muestra en la figura 43.

```
49a957b3
```

```
49a957b3
```

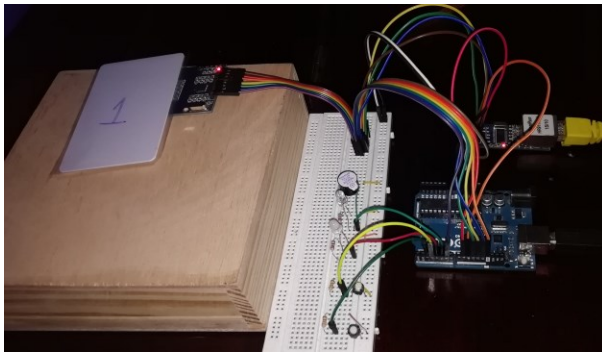
```
Ingreso autorizado
```

**Figura 42.** Monitor serial etiqueta bicicleta segunda prueba. Autoría propia.



**Figura 43.** Lectura etiqueta bicicleta segunda prueba. Autoría propia.

Como se muestra en la figura 44, se validó el funcionamiento del sistema de control de acceso, se tomó la lectura de la etiqueta registrada en la base de datos de usuarios, en la figura 45 se muestra la indicación al usuario para acercar la etiqueta asignada a la bicicleta.



**Figura 44.** Lectura carné tercera prueba. Autoría propia.

```
-> Oprima el botón 1
-> 551bfb2a
->
->
-> <<< REQ 551bfb2a
-> >>>
-> >>>
-> 723260d4
-> 1
-> Acerque el tag de la bicicleta
```

**Figura 45.** Monitor serial carné tercera prueba. Autoría propia.

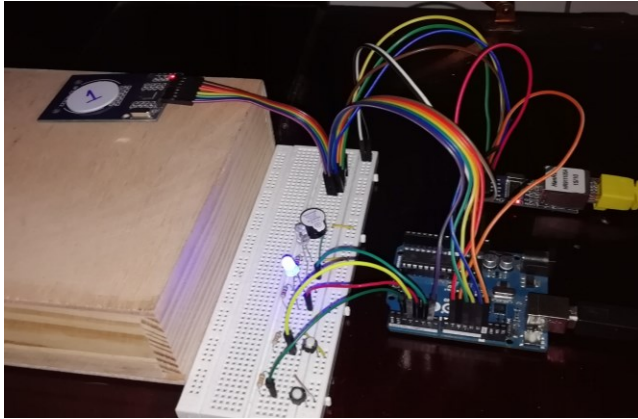
Se procede a tomar la lectura de la etiqueta asignada a la bicicleta, esta se compara con la variable en la que se guardó el UID de la bicicleta como se observa en la figura 46, se valida la relación entre la variable y la lectura, y se autoriza mediante la activación de un LED como se muestra en la figura 47.

```

-> Acerque el tag de la bicicleta
-> 723260d4
-> 723260d4
->
-> Ingreso autorizado

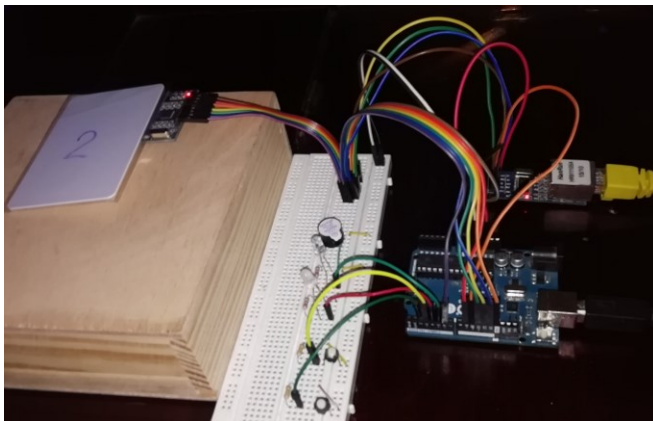
```

**Figura 46.** Monitor serial etiqueta tercera prueba. Autoría propia.



**Figura 47.** Lectura etiqueta bicicleta tercera prueba. Autoría propia.

En la figura 48, se toma la lectura de la etiqueta registrada en la base de datos de usuarios bajo el “ID = 2”, en el monitor serial se muestra la indicación al usuario para acercar la etiqueta asignada a la bicicleta como se muestra en la figura 49.



**Figura 48.** Lectura carné cuarta prueba. Autoría propia.

```

-> 0b6bea2b
->
->
-> <<< REQ b6bea2b
->
-> <<< REQ b6bea2b
-> >>>
-> >>>
-> f22b4cd4
-> 1
-> Acerque el tag de la bicicleta

```

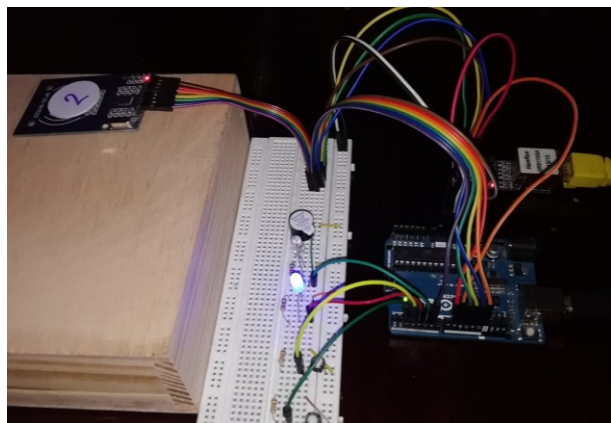
**Figura 49.** Monitor serial carné cuarta prueba. Autoría propia.



Se toma la lectura de la etiqueta asignada a la bicicleta, esta se compara con la variable en la que se guardó el UID de la bicicleta como se observa en la figura 50, se valida la relación entre la variable y la lectura, y se autoriza mediante la activación de un LED como se muestra en la figura 51.

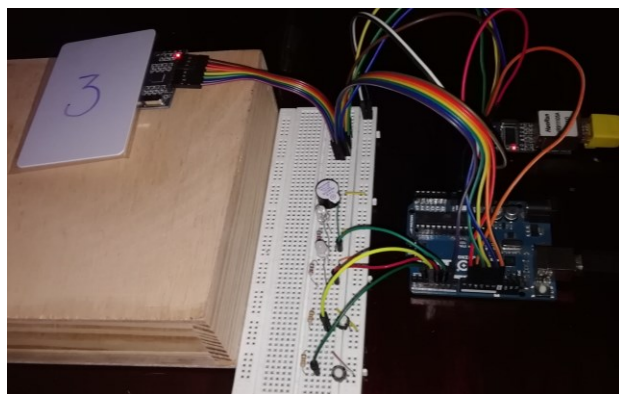
```
-> Acerque el tag de la bicicleta  
-> f22b4cd4  
-> f22b4cd4  
->  
-> Ingreso autorizado
```

**Figura 50.** Monitor serial etiqueta cuarta prueba. Autoría propia.



**Figura 51.** Lectura etiqueta bicicleta cuarta prueba. Autoría propia.

Como se muestra en la figura 52, se tomó la lectura de la etiqueta registrada en la base de datos de usuarios bajo el “ID = 3”, en la figura 53 se muestra la indicación al usuario para acercar la etiqueta asignada a la bicicleta.



**Figura 52.** Lectura carné quinta prueba. Autoría propia.

```

-> b378a940
->
->
-> <<< REQ b378a940
->
-> <<< REQ b378a940
->
-> <<< REQ b378a940
-> >>>
-> >>>
-> b2d8c3d4
-> 1
-> Acerque el tag de la bicicleta

```

**Figura 53.** Monitor serial carné quinta prueba. Autoría propia.

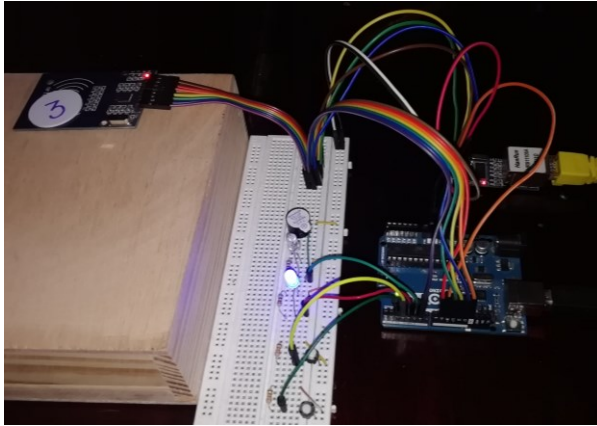
Se procede a tomar la lectura de la etiqueta asignada a la bicicleta, esta se compara con la variable en la que se guardo el UID de la bicicleta como se observa en la figura 54, se valida la relación entre la variable y la lectura, y se autoriza mediante la activación de un LED como se muestra en la figura 55.

```

-> Acerque el tag de la bicicleta
-> b2d8c3d4
-> b2d8c3d4
->
-> Ingreso autorizado

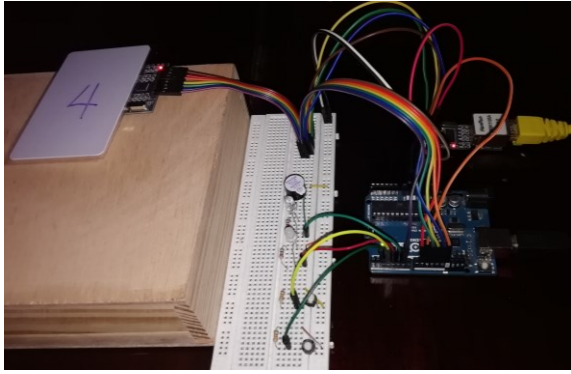
```

**Figura 54.** Monitor serial etiqueta quinta prueba. Autoría propia.



**Figura 55.** Lectura etiqueta bicicleta quinta prueba. Autoría propia.

Como se muestra en la figura 56, se tomó la lectura de la etiqueta registrada en la base de datos de usuarios con el “ID=4”, en la figura 57 se muestra la indicación al usuario para acercar la etiqueta asignada a la bicicleta.



**Figura 56.** Lectura carné sexta prueba. Autoría propia.

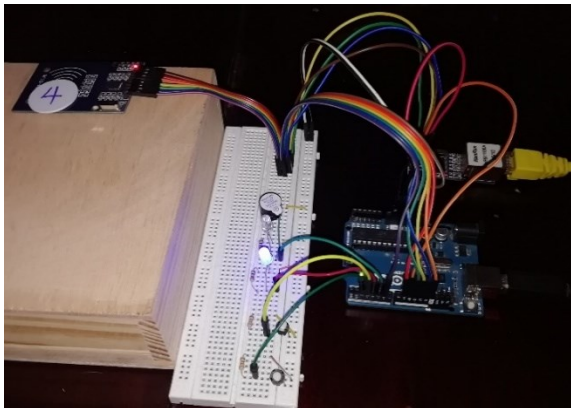
```
-> d056e732
->
->
-> <<< REQ d056e732
->
-> <<< REQ d056e732
-> >>>
-> >>>
-> 52a7a9d4
-> 1
-> Acerque el tag de la bicicleta
```

**Figura 57.** Monitor serial carné sexta prueba. Autoría propia.

Se procede a tomar la lectura de la etiqueta asignada a la bicicleta, esta se compara con la variable en la que se guardó el UID de la bicicleta como se observa en la figura 58, se valida la relación entre la variable y la lectura, y se autoriza mediante la activación de un LED como se muestra en la figura 59.

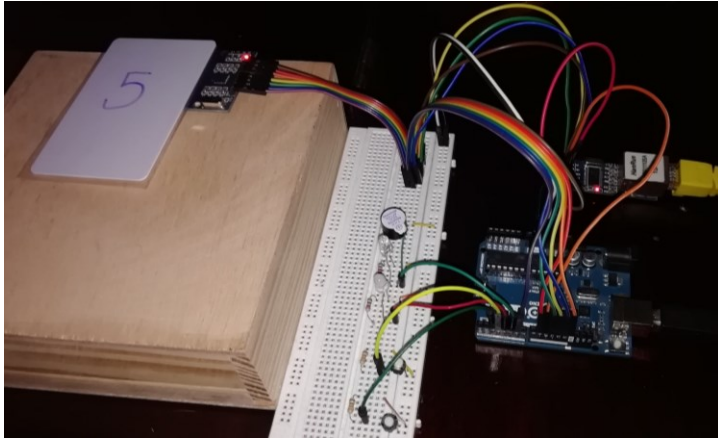
```
-> Acerque el tag de la bicicleta
-> 52a7a9d4
-> 52a7a9d4
->
-> Ingreso autorizado
```

**Figura 58.** Monitor serial etiqueta bicicleta sexta prueba. Autoría propia.



**Figura 59.** Lectura etiqueta bicicleta sexta prueba. Autoría propia.

En la figura 60, se toma la lectura de la etiqueta registrada en la base de datos de usuarios bajo el “ID = 5”, en el monitor serial se muestra la indicación al usuario para acercar la etiqueta asignada a la bicicleta como se muestra en la figura 61.



**Figura 60.** Lectura carné séptima prueba. Autoría propia.

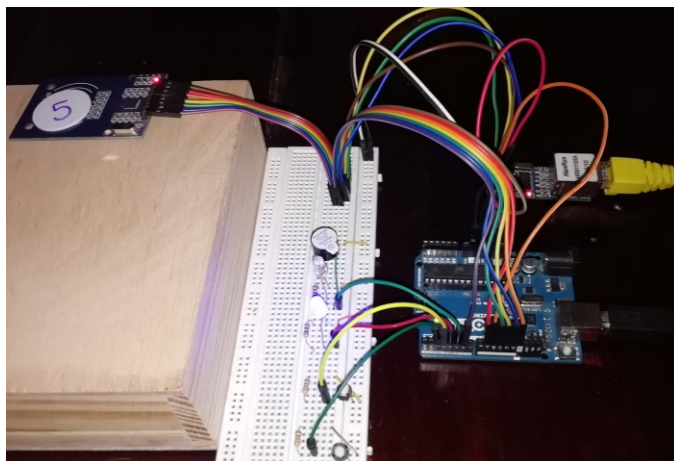
```
-> c3711f40
->
->
-> <<< REQ c3711f40
->
-> <<< REQ c3711f40
-> >>>
-> >>>
-> a2e1d8d4
-> 1
-> Acerque el tag de la bicicleta
```

**Figura 61.** Monitor serial carné séptima prueba. Autoría propia.

Se toma la lectura de la etiqueta asignada a la bicicleta, esta se compara con la variable en la que se guardó el UID de la bicicleta como se observa en la figura 62, se valida la relación entre la variable y la lectura, y se autoriza mediante la activación de un LED como se muestra en la figura 63.

```
-> Acerque el tag de la bicicleta
-> a2e1d8d4
-> a2e1d8d4
->
-> Ingreso autorizado
```

**Figura 62.** Monitor serial etiqueta séptima prueba. Autoría propia.

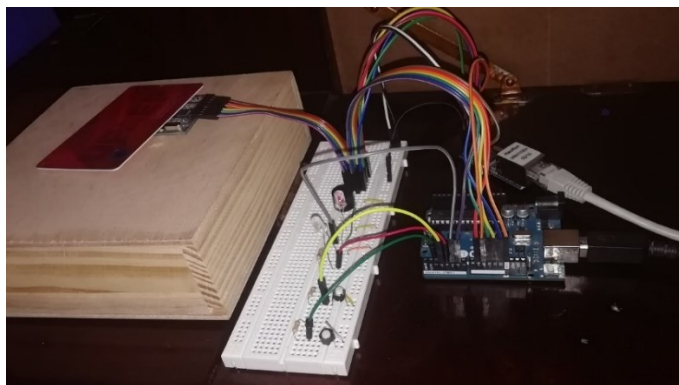


**Figura 63.** Lectura etiqueta bicicleta séptima prueba. Autoría propia.

En el caso de que no se tome la lectura de dos etiquetas que no estén registrados, ni relacionadas en la base de datos como se puede visualizar en la figura 64, se activará un LED rojo y un zumbador indicando que no se autoriza la entrada. En la figura 65 se muestra la activación del LED cuando se toma la lectura de una etiqueta que no está registrada.

```
<<< REQ
>>>
>>>
0
0
Carné no registrado en la base de datos
Ingreso no autorizado
```

**Figura 64.** Monitor serial carné octava prueba. Autoría propia.



**Figura 65.** Lectura carné octava prueba. Autoría propia.

Se tomó la lectura de un carné registrado como se muestra en la figura 66 y una etiqueta de bicicleta no registrada en la base de datos, y la respuesta dada en el monitor serial es que no se autoriza el ingreso como se muestra en la figura 67. Esta respuesta se representa físicamente mediante la activación de un led rojo, como se muestra en la figura 65.

```

-> Oprima el botón 1
-> 9b51fbad
->
->
-> <<< REQ 9b51fbad
->
-> <<< REQ 9b51fbad
-> >>>
-> >>>
-> f581dbd9
-> 1
-> Acerque el tag de la bicicleta

```

**Figura 66.** Monitor serial carné novena prueba. Autoría propia.

```

-> Acerque el tag de la bicicleta
-> f581dbd9
-> 49a957b3
->
-> Ingreso no autorizado

```

**Figura 67.** Monitor serial etiqueta bicicleta novena prueba. Autoría propia.

Como se mostró anteriormente, se realizaron nueve pruebas, en las que se lograron validar que el sistema tiene la capacidad de identificar la relación existente entre las etiquetas (carné y etiqueta de la bicicleta) registradas en las tablas de las bases de datos, mostrada en la figura 34. Se realizaron pruebas en las cuales se tomaron la lectura de etiquetas no registradas o no relacionadas, en la cual da como respuesta en el monitor serial que el ingreso no se autoriza.

### **Recomendaciones**

Después de comprobar el funcionamiento del prototipo del sistema de control de acceso, se puede realizar las debidas investigaciones para diseño a un nivel empresarial y así poder ofrecerlo a las instituciones que estén interesadas en él.

## Conclusiones

Para realizar el diseño del sistema de control de acceso, se debe tener en cuenta la frecuencia de operación de las etiquetas usadas por parte de la Universidad y de las etiquetas que se van a asignar a las bicicletas, esto con el fin de aprovechar las instalaciones existentes en una universidad.

Mediante la implementación de un circuito sencillo, se posibilitó la lectura diferenciada de las etiquetas con un mismo lector, logrando así simplificar el diseño del sistema de control de acceso, y proponer así un diseño de un sistema simplificado que se podría implementar en lugares que usen esta tecnología.

Se logró realizar la validación y comparación de las etiquetas, mediante la lectura de las etiquetas y envío de las mismas por medio de una petición GET al servidor web para su posterior comparación y devolución de dicha validación autorizando el ingreso al parqueadero, la relación entre la base de datos de los carnés institucionales y la base de datos de las bicicletas, permitió que esto se lleve a cabo, lo que indica que se puede implementar este diseño a cualquier universidad que use esta tecnología mediante la relación de las bases de datos.



## Referencias

- Agencia Nacional del Espectro. (2016). Resolución 711 de 2016. Bogotá, D. C.: Diario Oficial No. 50.061.
- All Electronics. (2020). *RFID Read and Write Module*. Obtenido de [www.allectronics.com](http://www.allectronics.com)
- Aprendiendo Arduino. (2020). *Ethernet Shield*. Obtenido de <https://aprendiendoarduino.wordpress.com/>
- Arduino. (2020). *Arduino Uno REV3*. Obtenido de TECH SPECS: <https://store.arduino.cc/usa/arduino-uno-rev3>
- Arduino. (2020). *Web Client*. Obtenido de <http://www.arduino.cc>
- Bakht, K., Ud Din, A., Shehzadi, A., & Aftab, M. (2019). Design of an Efficient Authentication and AccessControl System Using RFID. Rawalpindi: Fatima Jinnah Women University.
- Balsero, A., & Vargas, C. (2016). *Diseño e implementación de un prototipo para el control de acceso en la sede de Ingeniería de la universidad distrital Francisco José de Caldas mediante el uso de torniquetes controlados por carnet con tecnología NFC y lector biométrico de huella dactilar*. Obtenido de Universidad Distrital: <http://repository.udistrital.edu.co/bitstream/11349/3430/1/VargasGarciaCristianGerman2016.pdf>
- Chulde, J., & Méndez, A. (2014). Estudio, diseño e implementación de un sistema que use tecnología nfc para el control de asistencia de los alumnos de la Universidad Israel. Ecuador: Quito: Universidad Israel, 2014.
- Congreso de Colombia. (2009). Ley 1341 de 2009. Bogotá, D.C.
- Congreso de la República. (2012). Ley Estatutaria 1581 de 2012. Bogotá, D.C., Colombia: Diario Oficial 48587.
- Crespo, R., Garcia, E., & Villalba, S. (2014). Sistema de registro y atención de pacientes para eHEALTH usando tecnología RFID/NFC. *Journal Boliviano de ciencias*, 25-29.
- Díaz, M. W. (2015). Access Control System using NFC and Arduino. Tegucigalpa: 2015 IEEE Thirty Fifth Central American and Panama Convention (CONCAPAN XXXV).
- Electronilab. (2020). *Módulo RFID/NFC PN532 – 13.56Mhz – Lector/Grabador*. Obtenido de <https://electronilab.co>
- Garzon, J., & Rodriguez, C. (2017). Diseño de sistema de seguridad y registro por medio de acceso RFID para modulo de parqueadero de bicicletas. Bogotá D.C.

- Haro Flores, E. G. (2016). Sistema de control de acceso y monitoreo de estudiantes con el uso de tecnología inalámbrica de identificación automática (RFID) en la Universidad Técnica del Norte. Ibarra, Ecuador.
- Hernández Sampieri, R. (2014). *Metodología de la investigación* (6ta Edición ed.). México D.F: McGraw-Hill / Interamericana editores, S.A. DE C.V.
- Microchip. (2020). *PIC18F4550 Datasheet*. Obtenido de <https://ww1.microchip.com/>
- NaylampMechatronics. (2020). *Módulo Ethernet ENC28J60*. Obtenido de <https://naylampmechatronics.com/alambrico/87-modulo-ethernet-enc28j60.html>
- NaylampMechatronics. (2020). *Tutorial módulo Lector RFID RC522*. Obtenido de <https://naylampmechatronics.com>
- Niño, N. (2016). Diseño e implementación de un sistema para la identificación de animales domésticos mediante la escritura y lectura de etiquetas NFC a través de dispositivos portátiles. 25-29. Bogotá D.C.
- NXP. (2020). *MIFARE Type Identification Procedure*. Obtenido de <https://www.nxp.com/>
- Olmos, H., & Lopez, C. (2018). Propuesta para la viabilidad de un dispositivo/plataforma que permita la optimización de espacio en ciclo - parqueaderos. Bogotá D.C: Universidad Distrital Francisco José de Caldas.
- Revista Semana. (2018). *Cuántas personas montan bicicleta en Bogotá*. Obtenido de Semana: <https://www.semana.com/nacion/articulo/cuantas-personas-montan-bicicleta-en-bogota/572660>
- Rincon Ingenieril. (2020). *Enviar datos de Arduino a MySQL con ethernet*. Obtenido de <https://www.rinconingenieril.es/>
- Romero, A. (2019). Desarrollo de un sistema para el control de inventario y seguridad de textos aplicando tecnología rfid en bibliotecas. Ecuador: Quito.
- T Haoyu Electronics. (2020). *Mifare 13.56Mhz RC522 RFID Card Reader Module*. Obtenido de [www.hotmcu.com](http://www.hotmcu.com)
- Tami, D., Ballesteros, J., & Fino, R. (2015). Sistema de identificación y registro para bicicletas a través de tecnología RFID en la Universidad Distrital “Francisco Jose de Caldas” sede tecnológica. *Visión electrónica: algo más que un estado sólido*, 5-22.

- Vargas, C. (2019). *Aumentan cifras de denuncias por hurto de bicicletas en Bogotá*. Obtenido de <https://www.rcnradio.com/bogota/aumentan-cifras-de-denuncias-por-hurto-de-bicicletas-en-bogota>
- Wang, X., & Wang, Y. (2018). An office intelligent access control system based on RFID. Zhejiang: 2018 Chinese Control And Decision Conference (CCDC).

## Anexos

### Código Arduino Uno

```

#include <SPI.h>
#include <MFRC522.h>
#include <EtherCard.h>
#define SS_PIN 6
#define RST_PIN 9
MFRC522 lector (SS_PIN, RST_PIN); // Crear instancia del MFRC522

String id;
String data;
String data0;
String check = "1";
String uidb;
String uband = "";
int pboton=0, pboton2=0;
int led_aut = 4;
int led_na = 5;

static byte mymac[] = { 0x00,0x25,0xAB,0x5C,0x86,0x5F };
byte Ethernet::buffer[700];
static uint32_t timer;
const char website[] PROGMEM = "pruebacontrolacceso.000webhostapp.com";
//called when the client request is complete

static void my_callback (byte status, word off, word len)
{
  Serial.println (">>>");
  String z = (const char*) Ethernet::buffer + off;
  Serial.println (">>>");
  Ethernet::buffer[off+300] = 0;

```

```
uidb = z.substring(243, 251);
Serial.println(uidb);
uband = z.substring(len - 1);
Serial.println(uband);
delay(500);
if(uband==check)
{
  Serial.println("Acerque el tag de la bicicleta");
}
else {
  Serial.println("Carné no registrado en la base de datos");
  Serial.println("Ingreso no autorizado");
  digitalWrite(led_na, HIGH);
  delay(2000);
  digitalWrite(led_na, LOW);
  Serial.println();
  id = "";
  data = "";
}
uband="";
pboton=0;
pboton2=0;
}
void setup()
{
  pinMode(led_aut, OUTPUT);
  pinMode(led_na, OUTPUT);
  Serial.begin(57600);
  SPI.begin(); // Initiate SPI bus
  lector.PCD_Init(); // Initiate MFRC522
  Serial.println("\n[webClient]");
```

```

if (ether.begin(sizeof Ethernet::buffer, mymac) == 0)
  Serial.println( "Failed to access Ethernet controller");

if (!ether.dhcpSetup())
  Serial.println("DHCP failed");
ether.printIp("IP: ", ether.myip);
ether.printIp("GW: ", ether.gwip);
ether.printIp("DNS: ", ether.dnsip);
if (!ether.dnsLookup(website))
  Serial.println("DNS failed");
ether.printIp("SRV: ", ether.hisip);
attachInterrupt(digitalPinToInterrupt(2),ISRA,RISING);
attachInterrupt(digitalPinToInterrupt(3),ISRB,RISING);
Serial.println("Oprima el botón 1");
}

void loop()
{
  while (pboton==1)
  {
    id = "";
    // Si hay tarjeta nueva presente la leemos
    if (lector.PICC_IsNewCardPresent() && lector.PICC_ReadCardSerial())
    {
      // Mostramos el ID de la tarjeta leída
      //Serial.print("ID tarjeta: ");

      for (byte i = 0; i < lector.uid.size; i++)
      {
        if(lector.uid.uidByte[i] < 0x10) {
          Serial.print("0");

```

```

    }
    else {
        //Serial.print(" ");
    }
    id += String(lector.uid.uidByte[i], HEX);
}
Serial.println(id);
Serial.println();
data0=id;
lector.PICC_HaltA();
delay(500);
}
ether.packetLoop(ether.packetReceive());
data = data0;
char charBuf[70];
data.toCharArray(charBuf, 50);
if (millis() > timer)
{
    timer = millis() + 5000;
    Serial.println();
    Serial.print("<<< REQ ");
    Serial.println (data);
    ether.hisport = 80;//to access local host
    ether.browseUrl(PSTR("/rfid.php?id="), charBuf, website, my_callback);
    data = "";
}
}
while(pboton2==1)
{
    Serial.println(uidb);
    //id = "";

```

```
// Si hay tarjeta nueva presente la leemos
if (lector.PICC_IsNewCardPresent() && lector.PICC_ReadCardSerial())
{
  for (byte i = 0; i < lector.uid.size; i++)
  {
    if(lector.uid.uidByte[i] < 0x10) {
      Serial.print("0");
    }
    else {
      //Serial.print(" ");
    }
    id += String(lector.uid.uidByte[i], HEX);
  }
  Serial.println(id);
  Serial.println();
  lector.PICC_HaltA();
  delay(500);
}
if(id == uidb)
{
  Serial.println("Ingreso autorizado");
  digitalWrite(led_aut, HIGH);
  delay(2000);
  digitalWrite(led_aut, LOW);
  Serial.println();
}
else {
  Serial.println("Ingreso no autorizado");
  digitalWrite(led_na, HIGH);
  delay(2000);
  digitalWrite(led_na, LOW);
}
```



```

    Serial.println();
  }
  delay(200);
  pboton=0;
  pboton2=0;
}
}
void ISRA(){
  pboton = 1;
}
void ISRB(){
  pboton2 = 1;
}

```

### **Código conexión Arduino – base de datos**

```

<?php
$id=$_GET["id"];
$mysql_server = "localhost";
$mysql_db = "id14881111_control_acceso";
$mysql_user = "id14881111_localhost";
$mysql_pass = "2u\>YP[|<E-NQ-%F";

$mysqli = new mysqli("localhost", "id14881111_localhost", "2u\>YP[|<E-NQ-%F",
"id14881111_control_acceso");

$result = $mysqli->query("select usu.UID_UTAG, bici.UID_BTAG from usuario as usu join
bicicleta as bici on usu.id=bici.id_user WHERE usu.UID_UTAG='$id'");

if ($result) {
    while ($row = $result->fetch_array()) {
        $UID_UTAG = $row['UID_UTAG'];
        $UID_BTAG = $row['UID_BTAG'];
        echo $UID_BTAG;
    }
}

```

```
$contar = mysqli_num_rows ($result);  
if($contar == 0){  
    echo "0";  
    }  
else {  
    echo "1";  
    $result1 = $mysqli->query ("insert into RegistroUsuario (UID_USER, FECHA) values  
('$id', NOW());  
    }  
?>
```