

Sistema de acceso a la biblioteca escolar con Arduino

¿Cómo monitorizar el acceso de los usuarios a la biblioteca escolar de las tardes?

XXXIV Certamen Jóvenes Investigadores 2023
convocado por el Instituto de la Juventud y el
Ministerio de Universidades

Investigación en el ámbito de las asignaturas de
Matemáticas II y TIC de 2ºBachillerato.

Curso 2021/2022

COLEGIO MARISTA “LA INMACULADA” DE GRANADA

19 de enero de 2023

Autor del trabajo (curso 2021/2022):

Arturo Olivares Martos (2ºBachillerato)

Profesor coordinador del trabajo (curso 2021/2022):

Daniel Partal García

Sistema de acceso a la biblioteca escolar con Arduino

¿Cómo monitorizar el acceso de los usuarios a la biblioteca escolar de las tardes?

Índice

RESUMEN	2
ABSTRACT	2
1. INTRODUCCIÓN	3
2. ANTECEDENTES	3
3. HIPÓTESIS DEL TRABAJO Y OBJETIVOS DE LA INVESTIGACIÓN	4
3.1 RFID COMO MÉTODO DE IDENTIFICACIÓN PERSONAL	4
3.2 BASES DE DATOS MYSQL PARA GESTIONAR EL REGISTRO	5
3.3 ARDUINO MEGA COMO SERVIDOR Y FORMULARIO PHP	5
3.4 ARDUINO UNO COMO INTERMEDIARIO	5
4. MATERIALES Y MÉTODOS	5
4.1 DISEÑO DE LA BASE DE DATOS.....	6
4.2 PROCESO DE ALTA DE NUEVOS USUARIOS	6
4.2.1 FORMULARIO DE SOLICITUD DE DATOS AL USUARIO NUEVO.....	7
4.2.2 ARDUINO MEGA PARA LEER NÚMERO DE LLAVE.....	7
4.3 USO DIARIO DE LA BIBLIOTECA. ENTRADAS Y SALIDAS.....	8
4.3.1 DEL USUARIO HACIA ARDUINO.....	8
4.3.2 DE ARDUINO AL SERVIDOR PHP (ARDUINO COMO CLIENTE).....	8
4.3.3 DE PHP A BASE DE DATOS MYSQL.....	8
4.3.4 DEL SERVIDOR PHP A ARDUINO.....	10
4.3.5 RESPUESTA. APERTURA Y CIERRE DE LA PUERTA CON LA CERRADURA ELECTRÓNICA	10
5. RESULTADOS	10
6. CONCLUSIONES	11
6.1 FORTALEZAS Y DEBILIDADES DE NUESTRO MODELO	11
7. AGRADECIMIENTOS	12
ANEXOS DE FIGURAS	13
BIBLIOGRAFÍA Y WEBGRAFÍA	18

RESUMEN

El Colegio Marista “La Inmaculada” de Granada ofrece, desde 2022, la posibilidad de que sus alumnos, por las tardes, puedan acceder a la biblioteca escolar y a la sala de estudio. Un adulto (de la asociación de madres y padres del colegio) está siempre presente.

Acceden alumnos de 6 a 18 años. Y sería deseable contar con un sistema automatizado de control de acceso, tanto de la persona adulta responsable que abre cada día la biblioteca como de los alumnos que asisten diariamente.

Con Arduino hemos diseñado un sistema de apertura electrónica de la puerta de la biblioteca y un sistema de control de entrada y salida de usuarios con tarjetas inalámbricas RFID, que cargan en tiempo real la información a una base de datos MySQL gestionada por un servidor PHP.

Las posibilidades de expansión de este proyecto son múltiples: información vía SMS/email al tutor/a del alumno/a de la entrada y salida del chico/a, gestión informatizada del préstamo de libros y reserva telemática de los espacios de uso común del centro.

En esta primera versión exponemos en el presente informe las funcionalidades ya operativas: desarrollo de la interfaz gráfica online, instalación y control de la cerradura electrónica, y registro del acceso de los usuarios.

Creemos que hemos implementado una herramienta práctica que resuelve necesidades concretas y reales del colegio.

ABSTRACT

The school "HH. Maristas La Inmaculada" of Granada offers since 2022 the possibility for its students, during the afternoons, to stay and use the school library and the study room. An adult (from the parents' association of the school) is always present.

Students from 6 to 18 years of age use it every day, and it would be desirable to have an automated access control system, both for the responsible adult who opens the library every day and for the students who attend daily.

With Arduino we have designed a system for electronic opening of the library door and a system to control the entry and exit of users with wireless RFID tags, which upload information in real time to a MySQL database managed by a PHP server.

The possibilities of expansion of this project are multiple: information via SMS/email to the student's tutor of the entrance and exit of the child, electronic management of the loan of books and telematic reservation of the spaces of common use of the school.

In this first version we present in this report the functionalities that are already operational: development of the online graphic interface, installation and control of the electronic lock, and registration of user access.

We believe that we have implemented a practical tool that solves concrete and real needs of the school.

1. INTRODUCCIÓN

Cuando un alumno/a accede, por las tardes, a la biblioteca escolar, sus padres esperan del colegio el control de la presencia del chico/a y la notificación de su momento de llegada y de salida.

Una hoja de firmas sirve de control interno y para crear una estadística de usuarios acumulados a lo largo de las semanas. Pero no permite informar a las familias en tiempo real y requiere de un proceso posterior para digitalizar la información acumulada diariamente, con el consiguiente factor humano que puede introducir errores en la transcripción de la información contenida en el papel a la base de datos digital.

Los módulos de Arduino permiten implementar de forma eficaz un sistema de entrada y salida de la biblioteca, así como de la apertura de la puerta principal. Bajo código de programación PHP del lado del servidor y con el diseño y gestión de una base de datos MySQL, podemos resolver las necesidades concretas del centro con los siguientes elementos programables:

- Diodos LED rojo y verde.
- Zumbadores buzzer.
- 2 módulos lectores RFID RC522.
- Llaves y tarjetas RFID.
- Arduino Ethernet Shield, para dar acceso a la red a Arduino.
- Una placa Arduino Mega como servidor, para registrar el alta de los nuevos usuarios. Se comunica con la base de datos MySQL mediante programas PHP.
- Dos placas Arduino Uno para controlar, respectivamente, la entrada y salida de usuarios a la biblioteca. Se comunican con la base de datos MySQL mediante programas PHP.
- 2 relés de un canal para controlar el paso de corriente hacia la cerradura electrónica.
- Cerradura electrónica NC.

2. ANTECEDENTES

La Ley Orgánica de Modificación de la LOE (LOMLE) es el marco legal educativo vigente en España, desde 2020, para las enseñanzas regladas preuniversitarias. Su ordenación y enseñanzas mínimas para Bachillerato están publicadas en el Real Decreto 243/2022 del 5 de abril¹.

Es una ley competencial, es decir, busca que el alumno alcance las destrezas suficientes para solventar situaciones de la vida cotidiana en los que deba expresarse con fluidez (en su lengua materna y en una lengua extranjera), aplicar razonamientos lógico-matemáticos en el contexto de las ciencias y la tecnología, manejar entornos digitales, con capacidad de autoaprendizaje, de trabajo en equipo, con iniciativa emprendedora y valorando el entorno cultural a su disposición en la sociedad.

El proyecto presentado en este informe encaja perfectamente con el espíritu y con la norma educativa actual, ya que permite al alumno trabajar descriptores operativos íntimamente relacionados con la adquisición óptima de las competencias clave:

- Se expresa con fluidez para intercambiar información y crear conocimiento con sus compañeros, sus profesores y con la comunidad de desarrolladores Arduino, PHP y MySQL en internet.
- Localiza, analiza y selecciona información de diferentes fuentes.
- Utiliza el inglés en la lectura de obras técnicas y en la participación de diversos foros online.
- Utiliza los métodos inductivos y deductivos propios del razonamiento matemático.
- Desarrolla el método científico en la creación de conocimiento, estableciendo hipótesis a verificar o refutar.
- Diseña prototipos, que son mejoradas gracias a la revisión constructiva de profesores y compañeros de clase.
- Aporta soluciones digitales y tecnológicas a necesidades concretas planteadas en las asignaturas de Matemáticas II y TIC de 2º Bachillerato.
- Aplica criterios de fiabilidad a la información publicada en internet, manejándose con seguridad y respetando los derechos de autor de los algoritmos publicados en diversos foros técnicos sobre Arduino y sobre lenguajes de programación del lado del servidor.
- Ofrece las conclusiones de su solución tecnológica al resto de compañeros de clase, que mejoran en su proceso de enseñanza-aprendizaje gracias al diseño de su base de datos, su interacción con el servidor web y su control de la placa de Arduino y de la cerradura electrónica.

- Crece en su aprendizaje de manera autónoma, más allá de los contenidos propios del Bachillerato incluso para el caso de alumnos brillantes.
- Planifica su trabajo y lo ejecuta de manera ordenada en un periodo largo de investigación.
- Evalúa necesidades y afronta retos.
- Promueve soluciones innovadoras, aplicando conocimientos técnicos específicos.

Arduino² es una plataforma electrónica de código abierto que permite programar hardware y software. Su utilización en proyectos de robótica y de gestión de sensores digitales es prácticamente ilimitado. Así, por ejemplo, en una gran variedad de accesos a habitaciones de hotel, a puertas comunes de urbanizaciones o a pistas deportivas privadas encontramos módulos RFID³ idénticos al que vamos a utilizar en el presente proyecto.

El lenguaje PHP⁴ del lado del servidor permite, mediante sus scripts y comandos, acceder a un servidor web y ejecutar algoritmos en él. Es un lenguaje muy flexible, que permite incrustar a su vez código HTML5, CCS o JavaScript. Existe una densa comunidad online de desarrolladores e infinidad de foros donde intercambiar dudas y aprender de los algoritmos de otros programadores.

Las bases de datos MySQL⁵, disponibles en la inmensa mayoría de servidores web del mercado, permiten almacenar de manera eficaz el alta de usuarios, sus momentos de entrada y salida, y conservar un histórico digital accesible desde cualquier ordenador conectado a internet gracias a la tecnología PHP.

En resumen: Arduino, PHP y MySQL nos brindan una posibilidad real, eficiente, profesional y de bajo coste de solventar la necesidad concreta que tiene la biblioteca de nuestro colegio por las tardes.

3. HIPÓTESIS DEL TRABAJO Y OBJETIVOS DE LA INVESTIGACIÓN

El propósito principal del proyecto es implementar la apertura y cierre de una cerradura electrónica en la puerta de la biblioteca, registrando a su vez la entrada y salida de usuarios que han sido dados de alta previamente en el servidor web.

Emplearemos Arduino como hardware fundamental del proceso de control, y el lenguaje PHP y las bases de datos MySQL para cargar los datos en un servidor web.

Las hipótesis de trabajo en las que descansa el proyecto son las siguientes:

- Un programa de gestión online coordina el alta de usuarios nuevos.
- Cada usuario posee un identificador que coincide con el código correspondiente de la llave que se le entrega.
- Esta llave es detectada por dos lectores RFID, próximos a ambos lados de la puerta de la biblioteca:
 - El lector situado fuera de la biblioteca permite abrir la cerradura electrónica si el usuario está identificado en la base de datos, almacenando en ella la entrada.
 - El lector situado dentro de la biblioteca realiza la misma gestión, pero en este caso de salida.
- Para evitar la entrada de varios usuarios con la misma llave, el sistema no permite la apertura desde fuera de la cerradura si no ha quedado registrada previamente su salida.

3.1 RFID como método de identificación personal

Para tener un control del acceso a la biblioteca, es necesario el registro de los usuarios que entran y salen de la misma. Para esta tarea, se va a hacer uso de la tecnología RFID (Radio Frequency Identification) que emplea ondas de radio para transmitir información.

Esta tecnología consta de un lector (módulo RFID RC522) y de llaves/tarjetas personales para cada usuario. Estas llaves contienen un identificador único, así como algunos otros datos programables (**Figura 1 de Anexos**).

El sistema RFID permite la reutilización de las tarjetas, ya que la información almacenada puede ser editada. Nosotros, en cambio, hemos optado por vincular en la base de datos MySQL la información del usuario con el identificador único de la tarjeta. La versatilidad de MySQL permite almacenar tantos datos como se desee del usuario y modificarlo directamente mediante código PHP.

El código de cada llave no lo vamos a modificar, sino que asociaremos ese código a un usuario a través de la base de datos. Si un usuario causa baja en el sistema, deja libre el uso de una llave. Así la numeración de esa llave se puede vincular a un nuevo usuario.

A través de este sistema podemos entregar a los usuarios una identificación única que permite el acceso y la salida de la biblioteca de forma rápida y controlada.

3.2 Bases de datos MySQL para gestionar el registro

Para el control de la autenticación es necesario disponer de un sistema donde tengamos almacenado el listado de usuarios con acceso a la biblioteca. Para tal fin optamos por un SGBD⁶ (sistema gestor de bases de datos) como MySQL, por ser el sistema de gestión de bases de datos relacional más extendido en la actualidad, al estar basado en código abierto.

El hecho de ser tan usado nos facilita el acceso a muchos materiales online que facilitan su aprendizaje y puesta en marcha.

Para un correcto desarrollo y acceso a la escalabilidad del proyecto, es esencial la normalización de las bases de datos que contengan la información del sistema de gestión de la biblioteca. En el proceso de diseño nos aseguraremos de que las bases de datos estén en tercera forma normal (NF 3⁷). MySQL, como SGBD, facilita este proceso de normalización.

En la base de datos almacenaremos los datos personales de los usuarios con acceso a la biblioteca y sus fechas y horas de entrada y salida. Esta información automáticamente digitalizada nos permite conocer el grado de utilización de la biblioteca, los horarios de mayor afluencia o los usuarios que hacen más uso de la sala.

3.3 Arduino Mega como servidor y formulario PHP

Arduino es una plataforma ampliamente extendida que permite implementar circuitos electrónicos de una forma sencilla. Consta de una gran cantidad de entradas y salidas analógicas y digitales, junto con su correspondiente memoria y microprocesador.

Debido a la versatilidad de Arduino, hay gran cantidad de componentes, sensores, módulos, etc. disponibles en el mercado compatibles con él. Tan solo es necesario que estos generen una señal analógica, ya que los fabricantes de cada módulo se encargan de diseñar una biblioteca para su cómoda gestión a la hora de programarlo. Los lectores RFID son un ejemplo de esto.

Debido a la gran cantidad de proyectos para los que se emplea, Arduino ofrece una amplia variedad de modelos con diferentes características según el número de entradas, el tamaño, la mayor o menor memoria RAM, etc. Arduino Uno es el modelo más estándar, mientras que Arduino Mega ofrece características ampliadas.

A la hora de dar de alta a nuevos usuarios es necesario, en primer lugar, saber qué llave se la va a entregar. Una placa Arduino Mega, conectada a un ordenador personal situado en la biblioteca, permite leer el código de la llave que se va a vincular al nuevo usuario y mostrarlo en una web. Posteriormente ese código leído, junto con los datos personales del usuario obtenidos de un formulario PHP, se alojarán en la base de datos relacional, vinculando así la llave a dicho usuario (**Figura 2 de Anexos**).

A pesar de su coste mayor, se ha optado por un Arduino Mega en este caso porque, al funcionar como servidor, necesita una cantidad de memoria que sobrepasa las prestaciones de una placa Arduino Uno.

3.4 Arduino Uno como intermediario

Para que el sistema de apertura de la cerradura electrónica funcione, es necesario que se conecten los lectores RFID con el servidor donde está alojada la base de datos. Dos placas Arduino Uno se encargarán de ello.

Se emplea Arduino Uno como intermediario entre la dimensión más física del proyecto (la llave RFID del usuario y la cerradura) y la más abstracta (base de datos MySQL). Una vez que Arduino detecta la llave y el sistema valida al usuario, el propio Arduino envía la señal de apertura a un relé que controla la cerradura (**Figura 3 de Anexos**).

4. MATERIALES Y MÉTODOS

Desarrollamos, de manera ordenada, los diferentes elementos que forman el proyecto y sus correspondientes métodos de programación: base de datos MySQL, formulario PHP de alta de usuarios, conexión de Arduino con el servidor para validar los usuarios y conexión de Arduino con la cerradura electrónica.

Los códigos de programación completos de los archivos .php y .ino se pueden descargar en la siguiente carpeta de Google Drive:

https://drive.google.com/drive/folders/174imxFBwpZWwNn1YiZNLW3FNqBdUbSoq?usp=share_link

4.1 Diseño de la Base de Datos

El diseño adecuado de la base de datos relacional es esencial para el futuro del proyecto, ya que en el caso de que al implementar nuevas opciones fuese necesario añadir más información de cada usuario, la base de datos no debe ser un impedimento. Las diversas formas normales existentes nos permiten proteger los datos y hacer que esta sea más flexible al eliminar la redundancia y las dependencias incoherentes.

Para que esté en NF 3 es necesario, entre otras medidas, distinguir entre tablas dinámicas y estáticas, buscando así no mezclar en una misma tabla aquellos datos que varíen con gran frecuencia con aquellos datos que se modifican con menos periodicidad. Además, cada tabla debe de contar con un único campo clave cuyo valor no se repita para dos registros de la tabla.

La base de datos relacional empleada se ha diseñado en tercera forma normal, permitiendo así posibles mejoras del proyecto. Esta se encuentra dividida en 4 tablas:

1. Usuarios

Tabla estática. Almacena los datos personales de cada usuario (nombre, primer apellido, segundo apellido, correo electrónico y código de llave asociado). Los nuevos usuarios se añaden a esta tabla. El campo clave es el Id del usuario (**Figura 4 de Anexos**).

2. Usuario-Estado

Tabla dinámica con dos campos: el Id del usuario y el estado de este. Almacena en cada momento el estado de cada usuario de la biblioteca (dentro o fuera de la sala).

3. Registros

Tabla dinámica. El campo clave es el Id de cada Registro. A la hora de entrar, se crea el registro y se almacena el ID del Usuario que ha entrado, la fecha y la hora de entrada. Al salir, se modifica el campo HoraSalida, añadiéndose la hora en la que ha salido (**Figura 5 de Anexos**).

4. Estado

Tabla auxiliar estática que relaciona el valor de estado “1” con el valor “Fuera de la Biblioteca” y el valor estado “0” con el valor “Dentro de la Biblioteca”. Esta tabla es necesaria ya que, debido a la preferencia de trabajar con valores numéricos, en la tabla de Usuario-Estado se guarda un valor “1” o “0”.

4.2 Proceso de alta de nuevos usuarios

A la hora de registrar nuevos usuarios, es necesario introducir sus datos personales, como pueden ser su nombre y su correo electrónico. Con el objetivo de conseguir una interfaz cómoda y fácil de modificar/ampliar se ha optado por el lenguaje HTML5 del lado del navegador y por el lenguaje PHP del lado del servidor.

Mostramos, a continuación, un detalle del script PHP inicial para conectar con la base de datos:

```
<?php
```

```
//Variables para DataBase
```

```
$username="xxx";           $password="xxx";           $host="xxx";           $database="xxx";
```

```
//Configuración de la BD
```

```
$db = new mysqli($host, $username, $password, $database);
```

```
if ($db->connect_errno != null) {
```

```
    echo "Error número $db->connect_errno conectando a la base de datos.<br>Mensaje: $db->connect_error.";
```

```
    exit();
```

```
}
```

La aplicación PHP se encuentra alojada, en esta primera fase del proyecto, en el servidor de la web del profesor de la asignatura de Matemáticas II⁸. En una futura segunda fase, con objeto de satisfacer íntegramente con la normativa de privacidad de datos de los alumnos, las aplicaciones se llevarán a los propios servidores de la web del colegio.

4.2.1 Formulario de solicitud de datos al usuario nuevo

Al contrario que Arduino, HTML5 está enfocado en el desarrollo de la interfaz del usuario (Front-end⁹), por lo que ofrece más herramientas para validar los datos introducidos por este, minimizando así los errores humanos (**Figura 6 de Anexos**).

El formulario de los datos personales de los usuarios se ha implementado con HTML5 (**Figura 7 de Anexos**). Se han usado expresiones regulares (RegEx¹⁰) para indicar los caracteres disponibles. Por ejemplo, un nombre no puede contener números o un correo tiene que cumplir un formato determinado. De esta forma, la primera comprobación de la validez de los datos se realiza en el ordenador del usuario y no en el servidor.

Las RegEx empleadas han sido:

- Para el nombre y apellidos: "[A-z][A-za-z]+", ya que necesariamente el primer carácter es una letra mayúscula y el resto puede ser cualquier letra o espacio.
- Para el correo electrónico: "[a-z0-9!#\$%&'*/+=?^_`{|}~-]+(?:\.[a-z0-9!#\$%&'*/+=?^_`{|}~-]+)*@(?:[a-z0-9](?:[a-z0-9-]*[a-z0-9])?\.)+[a-z0-9](?:[a-z0-9-]*[a-z0-9])?". Esta es la forma correcta que debe cumplir un email según el estándar RFC2822¹¹.
- Para el código de la llave: "[A-Z0-9]{2}-[A-Z0-9]{2}-[A-Z0-9]{2}-[A-Z0-9]{2}", ya que obligatoriamente consta de 4 pares de letras mayúsculas y dígitos separados cada par por un guion.

Además de usar las expresiones regulares, se han usado también métodos de validación propios de PHP, como se puede ver en el siguiente código.

```
//Variables de POST
```

```
$CodLlave=""; $Nombre=""; $PrimerApellido=""; $SegundoApellido=""; $CorreoElectronico="";  
  
if ($_SERVER["REQUEST_METHOD"] === "POST") {  
    $CodLlave=htmlspecialchars(trim($_POST["CodLlave"]));  
    $Nombre=htmlspecialchars(ucwords(trim($_POST["nombre"])));  
    $PrimerApellido=htmlspecialchars(ucwords(trim($_POST["PrimerApellido"])));  
    $SegundoApellido=htmlspecialchars(ucwords(trim($_POST["SegundoApellido"])));  
    $CorreoElectronico=filter_var(trim($_POST["email"]), FILTER_SANITIZE_EMAIL); ...
```

4.2.2 Arduino Mega para leer número de llave

La placa Arduino Mega funciona como servidor. Contiene una web en la que se muestra el código de la última llave leída por el lector. Esta web muestra fácilmente el código de la última llave consultada.

Detalle del código Arduino para leer el código de una llave:

```
String ObtenerCodigo(byte *buffer, byte bufferSize) {  
    String CodLlave="";  
    for (byte i = 0; i < bufferSize; i++) {  
        CodLlave += (buffer[i] < 0x10 ? "-0" : "-");  
        CodLlave += String(buffer[i], HEX);  
    }  
    CodLlave.remove(0,1); //Elimina el guión inicial  
    CodLlave.toUpperCase();  
    return CodLlave;  
}
```

Detalle del código Arduino para mostrar el código de la llave en una web, implementando código HTML.

```
client.print("<body><h1>Código de llave</h1><p>Código de la última llave leída: <strong>");  
client.print("<a id='CodLlave'>");  
client.print(CodLlave);  
client.print("</a></strong>>t");
```


4.3 Uso diario de la biblioteca. Entradas y Salidas

Con una red de área local (LAN, de las iniciales en inglés de Local Area Network) (**Figura 8 de Anexos**) interconectamos el servidor que aloja los programas PHP y la base de datos con las placas Arduino que controlan el proceso de entrada y salida de la biblioteca y la apertura de la cerradura electrónica (**Figura 9 de Anexos**).

4.3.1 Del usuario hacia Arduino

A la hora de entrar y salir de la biblioteca, el usuario pasa la llave por el lector correspondiente y espera a que el sistema gestione si se abre o no la puerta.

Detalle del código Arduino de detección de la llave:

```
String CodLlave="";  
  
void loop(){  
  
  // Detectar tarjeta  
  
  if (mfrc522.PICC_IsNewCardPresent()){  
  
    if (mfrc522.PICC_ReadCardSerial()){  
  
      CodLlave = ObtenerCodigo(mfrc522.uid.uidByte, mfrc522.uid.size);  
  
      // Finalizar lectura actual  
  
      mfrc522.PICC_HaltA();  
  
      // Arduino hace la petición al servidor.  
  
      respuesta_Sql = httpRequest(CodLlave);
```

4.3.2 De Arduino al servidor PHP (Arduino como cliente)

Cada Arduino Uno actúa como cliente en este caso. Una vez se ha obtenido el código de la llave correspondiente, Arduino realiza una petición al servidor indicando el archivo PHP deseado (el correspondiente a la entrada o a la salida) y el código de la llave del usuario. Este código es enviado mediante el método GET, debido a la simplicidad con la que se pasan los datos con este método.

Detalle del código Arduino que gestiona la petición GET:

```
//Dirección del archivo php dentro del servidor  
  
peticion += "GET ";  
  
peticion+="/biblioteca/hora_entrada.php?CodLlave=";  
  
//Mandamos la variable junto a la línea de GET  
  
peticion+=String(identificador);  
  
...  
  
client.println(peticion);
```

4.3.3 De PHP a base de datos MySQL

La parte principal del sistema se lleva a cabo en el servidor mediante PHP. El servidor recibe por parte de Arduino la petición GET con el código de llave del usuario, y comprueba si está registrado o no. Para ello, una vez se ha conectado a la base de datos MySQL, PHP permite ejecutar consultas SQL.

Detalle del código PHP que permite leer los datos del usuario en la base de datos:

```
//Variable de GET  
  
$CodLlave=$_GET['CodLlave'];  
  
//Obtengo los datos de la persona que ha pasado la llave  
  
$query1 = 'SELECT * FROM Usuarios WHERE CodLlave="'.$CodLlave.'";'  
  
$resultado1 = $db->query($query1);  
  
$usuario = $resultado1->fetch_array(MYSQLI_ASSOC);
```

```
//Comprobación de si la llave está registrada
```

```
if ($usuario["CodLlave"] != $CodLlave){  
    $valor=0;  
    echo "valor=" . $valor . "<br>";  
    die ("Key not found");  
}
```

En el caso afirmativo de que el usuario se encuentre registrado, se obtiene el estado del usuario en la base de datos. Esto se hace para evitar que un usuario que ya está dentro abra la puerta desde fuera, ya que esto implicaría que están entrando varias personas con la misma llave. Solo un usuario que está fuera de la sala puede entrar en la biblioteca.

Detalle del código PHP que valida si el usuario puede entrar en la biblioteca:

```
//Obtener el estado actual. ¿Está dentro o fuera?  
$query2 = "SELECT * FROM UsuarioEstado WHERE IdUsuario='".$usuario["IdUsuario"]."."  
$resultado2 = $db->query($query2);  
$estado_prev = $resultado2->fetch_array(MYSQLI_ASSOC);  
//Según el estado, le deja entrar o no.  
if ($estado_prev["IdEstado"] == 0){ //Si el usuario está FUERA, puede entrar.  
    $query3 = "UPDATE UsuarioEstado SET IdEstado='1' WHERE IdUsuario='".$usuario["IdUsuario"]."."  
    $resultado3 = $db->query($query3);  
}  
else{ //El usuario está ya DENTRO. No puede entrar.  
    $valor=0;  
    echo "valor=" . $valor . "<br>";  
    die ("User inside the library. Cannot enter.");  
}
```

Para salir se sigue un proceso análogo. Si el sistema confirma que se debe abrir la puerta, en primer lugar, se modifica su estado. Posteriormente se añade el registro a la tabla correspondiente y se envía la respuesta afirmativa (valor = 1) en la respuesta de la petición hecha por Arduino.

Detalle del código PHP que inserta el registro de entrada, incluyendo nombre, fecha y hora:

```
//Añadir Registro de Entrada.  
$query5 = "INSERT INTO Registros(IdUsuario, Dia, HoraEntrada) VALUES ('" . $usuario["IdUsuario"] . "', CURDATE(), CURTIME())";  
$resultado5 = $db->query($query5);  
//Responder con Valor=1.  
if (!($resultado1 && $resultado2 && $resultado3 && $resultado4 && $resultado5)){  
    $valor=0;  
    echo "valor=" . $valor . "<br>";  
    echo "Ha habido un error. Notificar al encargado.<br><strong>Error:</strong> " . $db->error . "<br> Vuelva a intentarlo."  
}else{  
    $valor=1;  
    echo "valor=" . $valor . "<br>";  
}
```

4.3.4 Del servidor PHP a Arduino

En función del valor devuelto por el servidor (1 o 0), Arduino genera una respuesta u otra. Si la respuesta es afirmativa (1), significa que la llave está previamente registrada y que el estado del usuario (dentro o fuera) es el correcto, por lo que se abre la puerta: se activa un LED verde y suena el buzzer. En caso contrario, la puerta permanece cerrada y se enciende el LED rojo.

Detalle del código Arduino que genera la respuesta (encender LED, activar buzzer y abrir la puerta) en caso de que permita la apertura de la cerradura electrónica:

```
void reaccion (String resultado){  
    if (resultado == "1"){ //Entra  
        digitalWrite (rele, HIGH);          tone(buzzer, 1600);          digitalWrite (LedVerde, HIGH);  
        delay(80);                          noTone(buzzer);          delay(2000);  
        digitalWrite (LedVerde, LOW);       digitalWrite (rele, LOW);  
    }  
    else if (resultado == "0"){  
        digitalWrite (LedRojo, HIGH);  
        for (int i=0; i<3; i++){  
            tone(buzzer, 1600);  
            delay(80);  
            noTone(buzzer);  
            delay(80);  
        }  
        digitalWrite (LedRojo, LOW);  
    }  
}
```

Si la respuesta es positiva se activa un relé, que es el encargado de manejar de forma directa la cerradura.

4.3.5 Respuesta. Apertura y cierre de la puerta con la cerradura electrónica

Arduino está conectado a un relé de un canal que controla directamente las conexiones de la cerradura.

El relé consta de 6 conexiones. Tres de ellas están directamente conectadas al Arduino (DC+, DC- e IN). Las dos primeras proporcionan tensión al componente, y la señal IN es la que controla si el relé está activado o no. Las otras tres conexiones (COM, NO (Normally Open) y NC (Normally Closed)) son las que están conectadas, en nuestro caso, a la cerradura y controlan si le llega corriente o no (**Figura 10 de Anexos**)¹².

Como su propio nombre indica, el relé normalmente cierra el circuito entre NC y COM, pero al proporcionarle tensión en IN se excita la bobina, abriéndose la conexión con NC y cerrándose a través de NO.

El relé controla la cerradura electrónica (tipo NC) (**Figura 11 de Anexos**), por lo que necesita corriente de forma continua siempre que se desee que la puerta esté cerrada. Para abrirla, tan solo es necesario abrir el circuito que conecta la cerradura. Al pasar la llave, y en el caso de que el sistema decida abrir la puerta, tan solo es necesario que Arduino active el relé. La puerta se abre y el usuario puede acceder a la biblioteca.

5. RESULTADOS

Recordamos que los códigos de programación completos de los archivos .php y .ino se pueden descargar en la siguiente carpeta de Google Drive:

https://drive.google.com/drive/folders/174imxFBwpZWwNn1YiZNLW3FNqBdUbSoq?usp=share_link

Nuestro proyecto ofrece las siguientes evidencias de su funcionamiento:

- Registro de nuevos usuarios.
Se ha establecido un ordenador en la biblioteca, al lado de la puerta. La placa Arduino Mega encargada de leer los códigos de llave, para dar de alta, se encuentra conectado a él (**Figura 12 de Anexos**).
El responsable de la biblioteca es el encargado de gestionar el alta de los nuevos usuarios mediante el formulario implementado en la web.
Una vez dado de alta el usuario, también se le da la llave que se le ha asignado.
- Apertura de la puerta desde fuera.
Al lado de la puerta por la parte exterior se ha situado una caja en cuyo interior se encuentra el Arduino Uno junto con el lector RFID. Al pasar el usuario registrado su llave correspondiente, se abre la puerta (**Figura 13 de Anexos**). El perno de la cerradura electrónica vuelve a subir cuando se cierra la puerta, tras la entrada del usuario.
- Apertura de la puerta desde dentro.
Análogamente, se encuentra una caja en el interior de la biblioteca para que se pueda pasar la llave al salir. En el interior se encuentra el Arduino Uno encargado de la salida junto con el relé encargado de gestionar la apertura de la puerta.
Tras la estancia dentro de la biblioteca, el usuario pasa su llave personal y se abre la puerta (**Figura 14 de Anexos**).
- Actualización de los campos de la base de datos.
En ambos procesos, tanto de entrada como de salida, se actualiza la base de datos relacional añadiendo el registro correspondiente a la tabla *Registros* y modificando el estado del usuario en la Tabla *Usuarios-Estado*. Mediante diversas consultas SQL hemos comprobado que podemos obtener los usuarios que más frecuentan la biblioteca, entre muchos otros resultados.

6. CONCLUSIONES

El proyecto se encuentra, actualmente, en perfecto funcionamiento. De lunes a jueves, de 16.00 a 19.00 horas, la biblioteca escolar atiende semanalmente a una media de 40 alumnos registrados (**Figura 15 de Anexos**).

El alta de usuarios está siendo realizado por madres/padres de la asociación del colegio, muchos de los cuales no poseen grandes destrezas digitales. Por lo tanto, la interfaz web y el registro de nuevos usuarios se está mostrando como sencilla y accesible a todo tipo de usuarios.

La apertura y cierre de la puerta principal funciona correctamente con la cerradura electrónica. Como medida de seguridad, el perno solo sube cuando ambas partes de la cerradura están alineados (evitando así que el perno suba con la puerta abierta). Una vez alineadas, la cerradura ofrece la posibilidad de subir el perno tras 0-3-6 segundos.

En un futuro, está proyectado activar el mismo sistema de entrada y salida a otros espacios comunes del colegio, como son la sala de grabación de vídeo y multimedia, por albergar material electrónico de bastante valor económico, que necesita de un control exhaustivo de las personas que acceden a ella.

6.1 Fortalezas y debilidades de nuestro modelo

Creemos que nuestro proyecto goza de las siguientes bondades:

- Es simple de mantener, eficiente en los recursos empleados y sencillo de gestionar independientemente de la habilidad digital de los usuarios.
- Su coste económico es muy bajo. La pérdida de una llave es fácilmente reemplazable por otra.
- La versatilidad de PHP y MySQL permiten que, una vez realizado el montaje de la cerradura y de los lectores correspondientes, se puedan realizar gran cantidad de gestiones y/o nuevas implementaciones de forma muy sencilla, tan solo cambiando el código en el servidor.
- El código Arduino es bastante flexible, y permite implementar de manera sencilla futuras mejoras en la funcionalidad del hardware.

- Ha sido muy bien acogido tanto por los alumnos, por los profesores como por la asociación de madres y padres que coordina el uso de la biblioteca. Por lo que nos invita a pensar que podemos llevar el proyecto para controlar el acceso a otras salas del colegio.

Y, por supuesto, detectamos las siguientes mejoras a implementar en un futuro:

- Traslado de las aplicaciones web al servidor común del colegio, por protocolo de privacidad de datos personales.
- Codificar el acceso al formulario de entrada, para que la url no sea pública. Utilizar método POST en sustitución de GET, mejorando así la privacidad de la información que viaja por la red.
- Diseñar un menú inicial en la web que permita nuevas funcionalidades como acceder a los datos de un usuario, modificarlos, cambiar la llave asociada, eliminar usuarios de la biblioteca, etc.
- Implementar un servicio de aviso automático a los padres/madres/tutores de los alumnos vía SMS/email con las horas de entrada y salida de sus hijos/as.
- Digitalizar la referencia de los libros de lectura de la biblioteca, para poder gestionar un servicio de préstamo de libros vinculado al número de serie de la llave de cada usuario.
- Con objeto de que el proyecto sea autosuficiente, económicamente hablando, podría plantearse la creación de una cuota testimonial de 1€ al año, que se devolvería al finalizar el curso si el alumno devuelve la llave en buen estado.

7. AGRADECIMIENTOS

Gracias, en primer lugar, al personal de mantenimiento del Colegio Marista “La Inmaculada” de Granada (Germán y José Miguel) por ayudarnos en el cableado del proyecto y en la instalación física del switch en la biblioteca y de la cerradura en el marco de la puerta.

Gracias a la asociación de madres y padres del colegio por acceder a que nuestro proyecto piloto de registro de usuarios y cerradura electrónica se pudiese implementar en la biblioteca.

Gracias al claustro de profesores (no solo de las asignaturas vinculadas de Matemáticas II y TIC) por cedernos ciertos huecos horarios para el desarrollo del proyecto.

Y gracias, finalmente, a los compañeros de 2º Bachillerato Tecnológico del curso 2021-2022 por su revisión constante en pro de la mejora del proyecto.

Terminamos nuestro trabajo con la sensación de haber aprendido mucho (y de haber trabajado aún más...), de haber resuelto dificultades que ni imaginábamos (dichosas conexiones de la cerradura electrónica...) y de haber resuelto de manera eficaz una necesidad concreta del colegio.

Vimos un problema, pensamos posibles alternativas y ejecutamos una solución que funciona. Hemos cerrado el círculo del método científico, aplicado a la informática y la electrónica.

ANEXOS DE FIGURAS

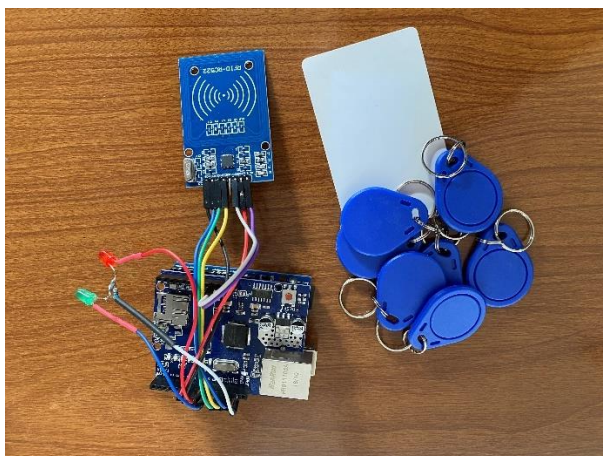


Figura 1. Módulo RFID RC522 conectado a Arduino Uno y llaves/tarjetas de de acceso.

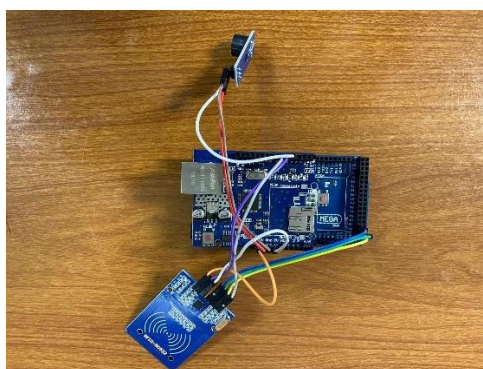


Figura 2. Arduino Mega utilizado como servidor conectado a Módulo RFID RC522.



Figura 3. Las dos placas Arduino Uno controlan la entrada y salida de los usuarios de la biblioteca. El relé controla el perno de la cerradura electrónica.


	IdUsuario	CodLlave	Nombre	PrimerApellido	SegundoApellido	CorreoElectronico
orrar	39	E9-74-83-CB	Arturo	Olivares	Martos	arturo@gmail.com
orrar	40	A0-82-47-22	Jghjg	Gghjghjg	Ghjghjghjg	1@g.com

Figura 4. Captura de pantalla de la Tabla Usuarios de la base de datos. Figura de elaboración propia.

	IdRegistro	IdUsuario	Dia	HoraEntrada	HoraSalida
	26	39	2022-10-17	14:15:33	NULL
	27	39	2022-10-17	14:15:44	NULL
	28	39	2022-10-17	14:15:48	NULL
	29	39	2022-10-17	14:15:57	NULL
	30	39	2022-10-17	14:16:32	NULL
	31	39	2022-10-17	14:16:39	NULL
	32	39	2022-10-17	14:18:13	NULL
	33	39	2022-10-17	14:18:28	NULL
	34	39	2022-10-18	11:57:11	12:03:41
	35	39	2022-10-18	11:57:20	12:03:41
	36	39	2022-10-18	11:57:30	12:03:41
	37	39	2022-10-18	11:58:38	12:03:41

Figura 5. Captura de pantalla de la Tabla Usuario-Estado de la base de datos. Figura de elaboración propia.

Añadir un nuevo usuario

Código de Llave: 


Nombre:

Primer Apellido:

Segundo Apellido:

Correo Electrónico:

Añadir un nuevo usuario

Código de Llave: 

Nombre:

Primer Apellido:

Segundo Apellido:

Correo Electrónico:


 Incluye un signo "@" en la dirección de correo electrónico. La dirección "arturo.com" no incluye el signo "@".

Figura 6. Ejemplo de validación del formulario desde el navegador. Figura de elaboración propia.

```
<label>C digo de Llaves:</label>
<input type="text" name="CodLlave" id="CodLlave" required pattern="[A-Z0-9]{2}-[A-Z0-9]{2}-[A-Z0-9]{2}-[A-Z0-9]{2}" value="<?=$CodLlave?>">
<button onclick="pasteFromClipboard()"><ion-icon name="clipboard-outline"></ion-icon></button>
<br><br>

<label>Nombre:</label>
<input id="nombre" name="nombre" type="text" required minlength="3" maxlength="50" pattern="[A-Za-z][A-Za-z ]+" value="<?=$nombre?>">
<br><br>

<label>Primer Apellido:</label>
<input id="PrimerApellido" name="PrimerApellido" type="text" required minlength="3" maxlength="50" pattern="[A-Za-z][A-Za-z ]+" value="<?=$PrimerApellido?>">
<br><br>

<label>Segundo Apellido:</label>
<input id="SegundoApellido" name="SegundoApellido" type="text" minlength="3" maxlength="50" pattern="[A-Za-z][A-Za-z ]+" value="<?=$SegundoApellido?>">
<br><br>

<label>Correo Electr nico:</label>
<input type="email" name="email" id="email" required pattern="^[a-zA-Z0-9.!#$%&'*/+=?^_{}|~]+@[a-zA-Z0-9]?(?:[a-zA-Z0-9]{0,61}[a-zA-Z0-9])?(?:\.[a-zA-Z0-9]{0,61}[a-zA-Z0-9])?(?:\.[a-zA-Z0-9]{0,61}[a-zA-Z0-9])?$">
<br><br>

<input type="submit" value="Submit">
</form>

<p><?=$feedback?></p>
```

Figura 7. Captura de pantalla del código HTML5 del formulario del alta de usuarios. Figura de elaboración propia.



Figura 8. Detalle del Switch de 5 puertos para gestionar la conexión a internet de las placas Arduino.

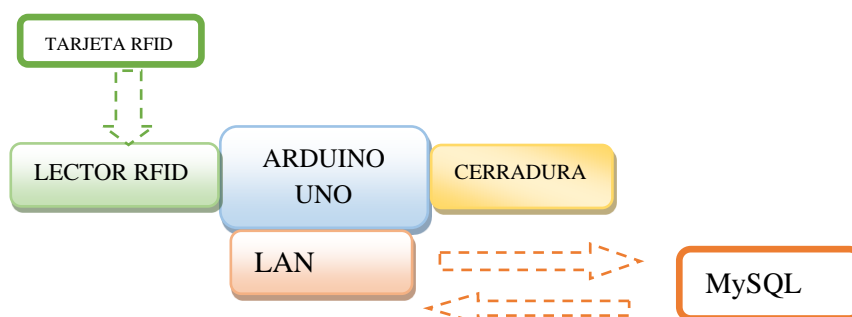


Figura 9. Visualización gráfica de las conexiones del proyecto. Figura de elaboración propia.

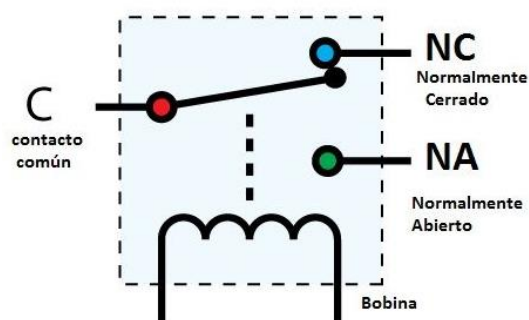


Figura 10. Relé y esquema electrónico. Referencia de las figuras:

<https://solectroshop.com/es/modulos-rele/71-modulo-rele-5v-10a-de-1-canal-disparo-bajo-alto-para-arduino.html>

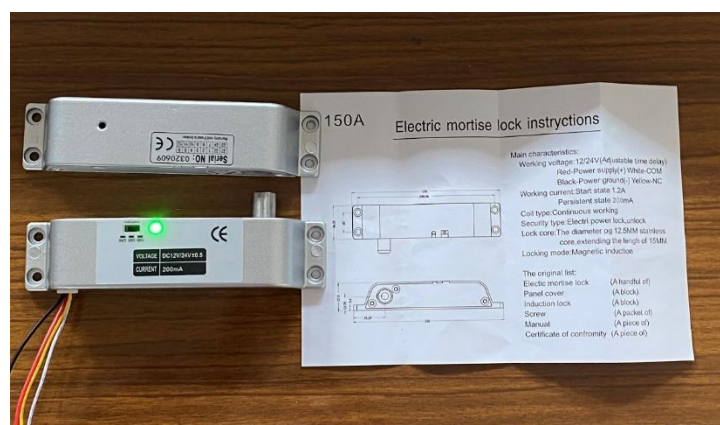


Figura 11. Cerradura electrónica de tipo NC empleada en el proyecto.

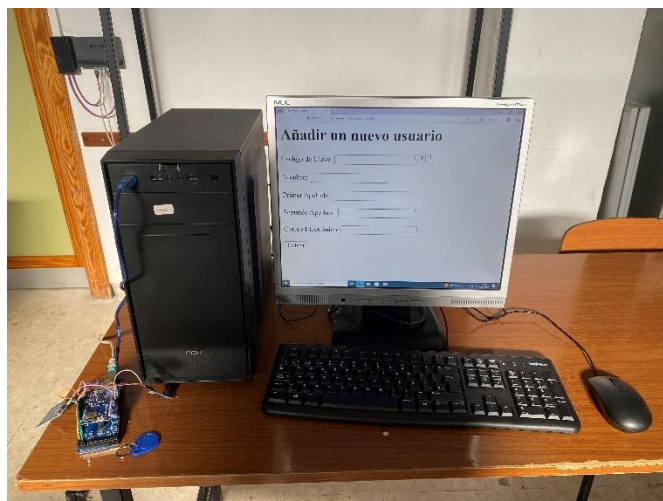


Figura 12. Ordenador situado en la biblioteca preparado para gestionar el alta de nuevos usuarios, junto con el Arduino Mega para leer los códigos de las llaves que se deseen.

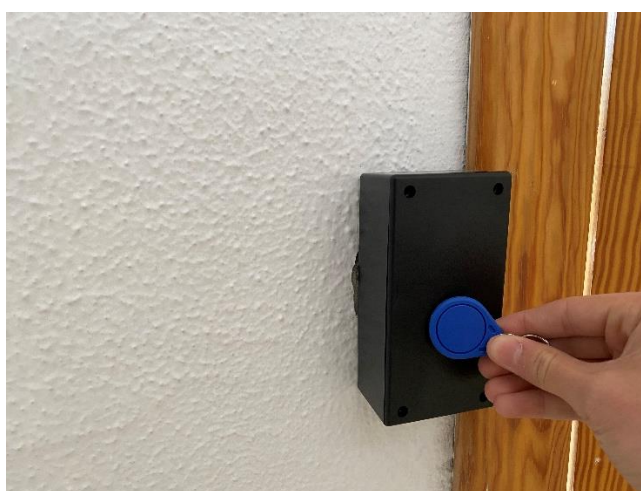


Figura 13. Usuario pasando su llave personal por el lector situado en el exterior de la biblioteca para poder entrar.

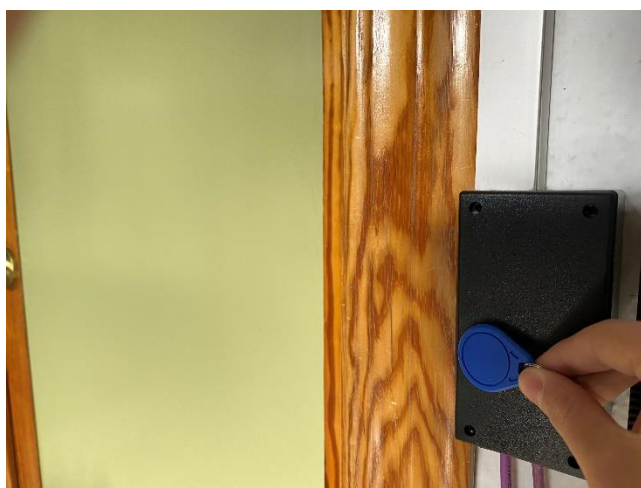


Figura 14. Usuario pasando su llave personal por el lector situado en el interior de la biblioteca para poder salir.



Figura 15. Cartel de apertura de la biblioteca escolar por las tardes.

BIBLIOGRAFÍA Y WEBGRAFÍA

- ¹ Gobierno de España. Ministerio de Educación y Formación Profesional. 2020. *Real Decreto 243/2022, de 5 de abril, por el que se establecen la ordenación y las enseñanzas mínimas del Bachillerato*. [en línea]
Available at: <https://www.boe.es/buscar/act.php?id=BOE-A-2022-5521>
[Último acceso: 18 Febrero 2022]
- ² Arduino. Arduino is an open-source electronics platform based on easy-to-use hardware and software. It's intended for anyone making interactive projects. [en línea]
Available at: <https://www.arduino.cc>
[Último acceso: 2 junio 2022]
- ³ RFID. Tecnología de identificación por radiofrecuencia
Available at: https://www.kimaldi.com/rfid_tecnologia_de_identificacion_por_radiofrecuencia
[Último acceso: 20 abril 2022]
- ⁴ PHP. Lenguaje de desarrollo web del lado del servidor. [en línea]
Available at: <https://www.php.net>
[Último acceso: 20 abril 2022]
- ⁵ MySQL. The world's most popular open source database. [en línea]
Available at: <https://www.mysql.com>
[Último acceso: 20 abril 2022]
- ⁶ ¿Para qué sirve un gestor de base de datos? [en línea]
Available at: <https://universidadeuropea.com/blog/para-que-sirve-gestor-base-datos>
[Último acceso: 20 abril 2022]
- ⁷ Normal Forms in DBMS. [en línea]
Available at: <https://www.geeksforgeeks.org/normal-forms-in-dbms>
[Último acceso: 13 abril 2022]
- ⁸ Web del profesor de Matemáticas II que aloja las aplicaciones PHP del proyecto. [en línea]
Available at: https://danipartal.net/biblioteca/add_user.php
[Último acceso: 2 junio 2022]
- ⁹ Desarrollo Web Front-end. [en línea]
Available at: https://developer.mozilla.org/es/docs/Learn/Front-end_web_developer
[Último acceso: 10 octubre 2022]
- ¹⁰ RegEx. [en línea]
Available at: <https://regexr.com>
[Último acceso: 24 mayo 2022]
- ¹¹ RFC-editor. [en línea]
Available at: <https://www.rfc-editor.org/rfc/rfc2822>
[Último acceso: 10 octubre 2022]
- ¹² Relé implementado en el proyecto. [en línea]
Available at: <https://solectroshop.com/es/modulos-rele/71-modulo-rele-5v-10a-de-1-canal-disparo-bajo-alto-para-arduino.html>
<https://www.areatecnologia.com/electricidad/rele.html>
[Último acceso: 19 enero 2023]