

Universidad Nacional de Luján
Lic. en Sistemas de Información
Artefactos Digitales Interactivos
Curso 2019



Trabajo Práctico Integrador

Control de Acceso Biométrico

Rodriguez Agustin

Nro Legajo: 96898

Mail: agustinrodriguez206@gmail.com

Quiroga Mario

Nro. Legajo: 113329

Mail: marioqui17@gmail.com

1. Introducción

El problema que se intenta resolver en este trabajo se encuentra en el ámbito de la Universidad Nacional de Luján. La cual quiere actualizar su sistema de control de asistencia del personal. Para esto se cuenta con un software administrativo de código libre cedido por la UNQ. Las especificaciones de este software (tienen la posibilidad de descargar los registros de cualquier api-rest), motivan la realización mediante Hardware libre, de un prototipo de reloj para el registro de los ingresos y egresos del personal, mediante los métodos actuales (tarjeta rfid) y huellas digitales, con la posibilidad de ser utilizado por el software cedido.

Siendo de gran importancia el Hardware libre, porque en el mercado hay pocos relojes con estas características, y generalmente son muy dependientes de software propietario. Y lo más importante es que un desarrollo libre y propio permitiría a las universidades no quedar atadas a proveedores externos. El proyecto puede ser también extensible a cerraduras electrónicas para el control de acceso en aulas con informáticas.

1.1 Objetivos del trabajo

El objetivo principal del trabajo consiste en construir el prototipo de Reloj de control de acceso de personal (Fichero), con sensor de huella digital y lector de tarjetas RFID, contando con almacenamiento propio (tarjeta micro-sd) del registro del ingreso y egreso del personal. Agregando como característica novedosa un servidor web para consumir la información del mismo, por cualquier sistema. También tendrá la posibilidad de enrolar el personal que usará el sistema.

El personal se registrará tanto en la entrada como en la salida en el fichero. Por lo tanto, según el horario de trabajo de cada empleado, se tendrá que decidir si el empleado entra o sale del establecimiento.

El artefacto, además contará con una display LCD para indicar la hora actual, y el número de legajo del empleado que registre la entrada o salida. Se construye además una pequeña interfaz web, a través del servidor, para poder probar el funcionamiento del prototipo.

1.2 Trabajos relacionados

En la actualidad la solución más aplicada se resuelve mediante productos empaquetados que proveen con ellos algún software de gestión sin posibilidades de ser modificados, generando problemas para cualquier institución que quiera personalizar su control de personal, también teniendo el problema que ante una ampliación de la organización los productos ya no están disponibles o fueron modificados.

Entre estos productos la mayoría cuenta con lector de huella y tarjetas rfid, pero no cuentan con acceso web y alguna api-rest para ser utilizados por software personalizado.

A continuación, se muestran dos ejemplos de productos ofrecidos en el mercado:



REIN2

Características:

- tecnología rfid y huellas
- conexión usb
- almacenamiento de 100000 registros
- alimentacion con bateria



ieco360

Características:

- Tecnología rfid y huellas
- Conexión usb
- Administración mediante web

2. Descripción del prototipo

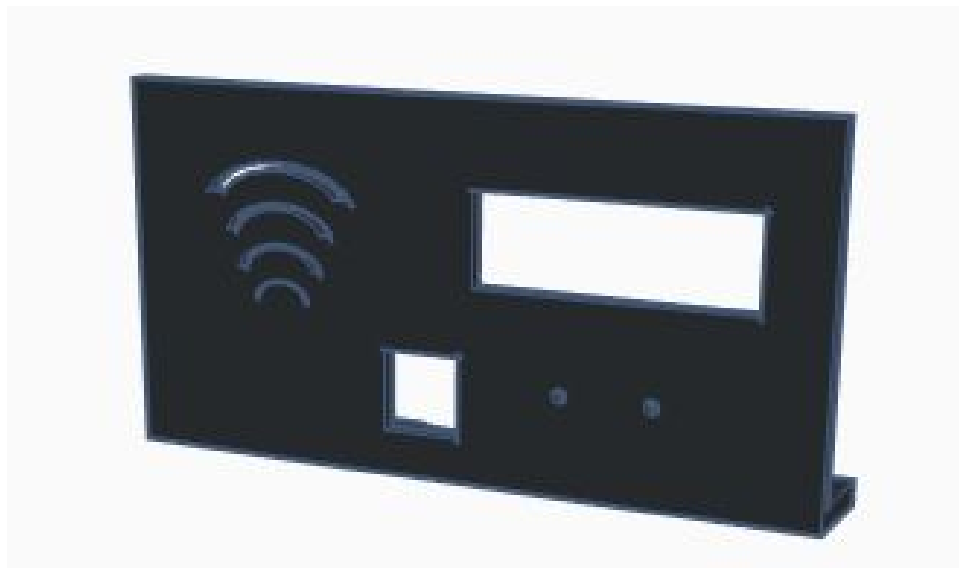
El prototipo de control de acceso de personal, está compuesto por un lector de huellas, un lector de tarjetas, un display, luces (verde y roja) y un buzzer. Además, se utilizará un shield para que la placa Arduino Mega pueda conectarse a una red. Permitiendo construir un servidor que enviará las fichadas a través de la red, más una pequeña interfaz web para gestionar el prototipo.

2.1 Diseño

Los componentes periféricos del prototipo se integran en un panel frontal, el cual será la interfaz para el personal que registre su ingreso o egreso al establecimiento. Se construye un modelo 3D imprimible, utilizando los componentes utilizados para modelar el objeto y respetar sus medidas. El objeto puede descargarse en versión STL en el sitio:

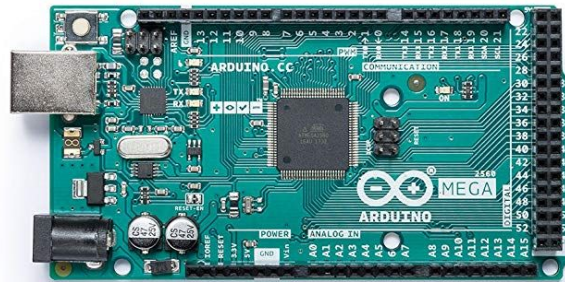
https://github.com/agrup/Arduino-control-access/blob/master/stl/Arduino_control_access-front.stl

A continuación se muestra una imagen del objeto diseñado para este prototipo:



2.2 Componentes de hardware

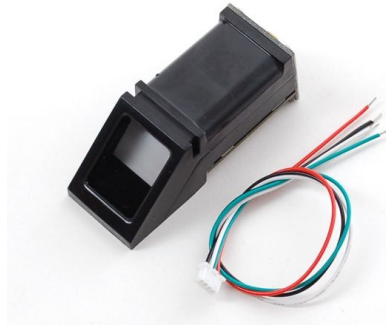
- **Arduino Mega:** Seleccionado por su mayor capacidad de memoria con respecto al Arduino Uno, esto se debe a la utilización de librerías como la “SD” , que consume gran parte de los recursos del Arduino.



- **Arduino Shield Ethernet :** Es un shield que cuenta con salida ethernet y un lector de tarjeta sd, características fundamentales para la realización del proyecto. La conexión ethernet es fundamental para poder acceder remotamente y consumir de su api-rest. También cuenta con con un lector de tarjetas sd, funcionalidad también necesaria para el proyecto, ya que se deben almacenar grandes cantidades de información.



- **Adafruit Fingerprint:** Lector de huellas digitales, que permite la detección y almacenamiento de patrones de huellas digitales: Componente que caracteriza el proyecto y que permite la identificación única de personas sin contar con un elemento extra.



- **Lector RFID:** Componente que permite la lectura de tarjetas con esta tecnología, este componente se agregó al proyecto para contar con una segunda posibilidad de registro de las personas, y así evitar inconvenientes que se puedan dar con el lector de huellas.

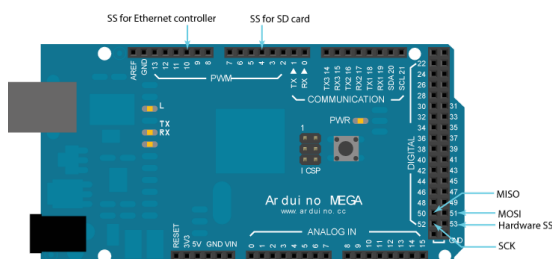


- **Display y módulo i2c:** Se utiliza el módulo i2c junto con el display, con el objetivo de poder controlar el display con menos cables. De modo que no ocupe demasiado espacio en el protoboard del proyecto.



2.3 Componentes de software

- **Arduino IDE:** Se utilizó para gestionar el proyecto y cargar el software de registro de fichadas. Disponible en: <https://www.arduino.cc/en/Main/Software>
- **Librería SD:** Librería utilizada para la lectura y guardado en la memoria sd. Nativa de Arduino.
- **Librería Ethernet:** Se utilizó para aprovechar la conexión ethernet del shield y generar un web services para la comunicación desde internet sin necesidad de acceso al dispositivo. Nativa de Arduino.



- **Librería ArduinoJson:** librería utilizada para la formación de documentos json. <https://github.com/bblanchon/ArduinoJson>
- **Librería SPI:** Esta librería permite la comunicación serial de dispositivos, en este proyecto utilizada para la comunicación entre el Arduino Mega y el el shield Ethernet. Nativa de Arduino.
- **Librería MFRC522:** Se utilizó para realizar lecturas de tarjetas con tecnología RFID, que es otro de los métodos de registración de fichadas. Disponible en el sitio: <https://github.com/miguelbalboa/rfid>
- **Librería Adafruit Fingerprint:** Utilizada para el sensor de huellas. Disponible en el sitio: <https://github.com/adafruit/Adafruit-Fingerprint-Sensor-Library>
- **Librería RTCLib:** Utilizada para las funciones de hora y fecha. Disponible en el sitio: <https://github.com/adafruit/RTCLib>

2.4 Pruebas del prototipo

Como se mencionó, para probar y administrar el prototipo se construyó un servidor que ofrece una interfaz web, accesible a través de un navegador. Las pruebas realizadas con el prototipo son:

- **Agregar huella al sistema:** En la interfaz web, apretar el botón agregar huella. Luego ingresar número de legajo en la caja de texto y apretar el botón “enrolar”. El

dispositivo le indicará que ingrese la huella dos veces, e indicará si falla o guarda correctamente la huella ingresada.

- **Agregar tarjeta al sistema:** El procedimiento es muy parecido a Agregar huella al sistema, con la excepción de que solo se ingresará una vez la tarjeta.
- **Registrar ingreso o egreso:** Un usuario previamente registrado debe ingresar su huella o tarjeta al dispositivo, y el sistema debe verificar si existe la persona asociada a un id de huella o tarjeta. Si existe se muestra el número de legajo del empleado y se guarda la fichada en un archivo ubicado en la memoria sd.
- **Ver fichadas:** Apretar el botón Ver Fichadas en la interfaz web. Se mostrarán las fichadas guardadas en el archivo de fichadas.

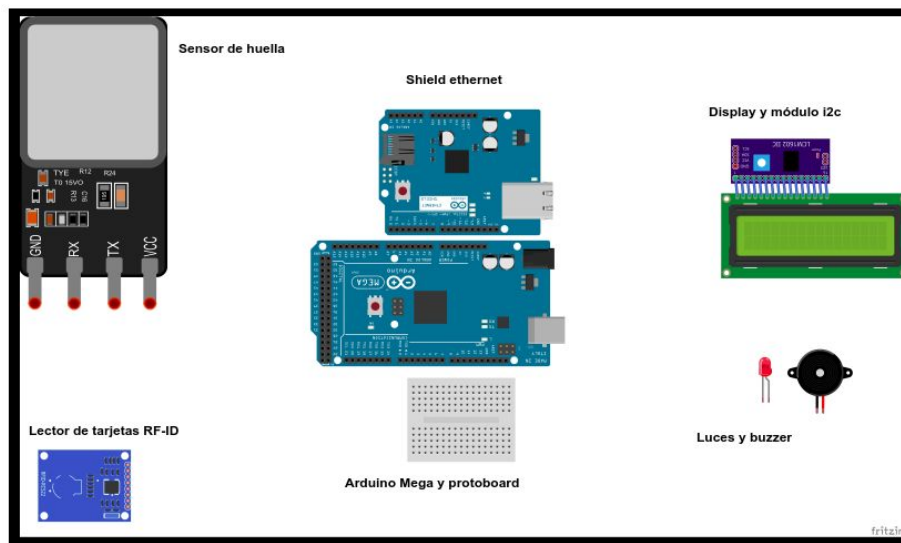
A partir de probar y poner en marcha el prototipo, surgieron problemas con el shield ethernet. El problema es que el shield usa pines digitales de forma transparente en sus librerías, bloqueando su uso por otros dispositivos. La solución a este problema fue no usar los pines bloqueados por el shield.

3 .Construcción del prototipo

Con el fin de que el prototipo construido sea open hardware, se crea un proyecto en github. Este tendrá el código fuente del sistema, los esquemas de conexión en formato .fzz, el modelo 3D del panel frontal en formato .stl, y este documento. El link al proyecto es: <https://github.com/agrup/Arduino-control-access>.

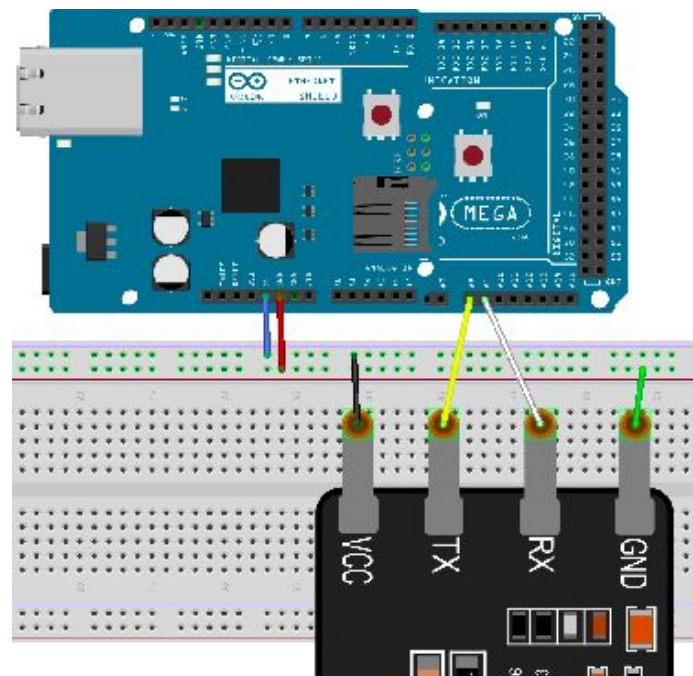
3.1 Hardware

La siguiente imagen muestra los componentes de hardware del proyecto. Se necesitarán además, otro led de color verde y una memoria micro sd para guardar los registros de la fichada.

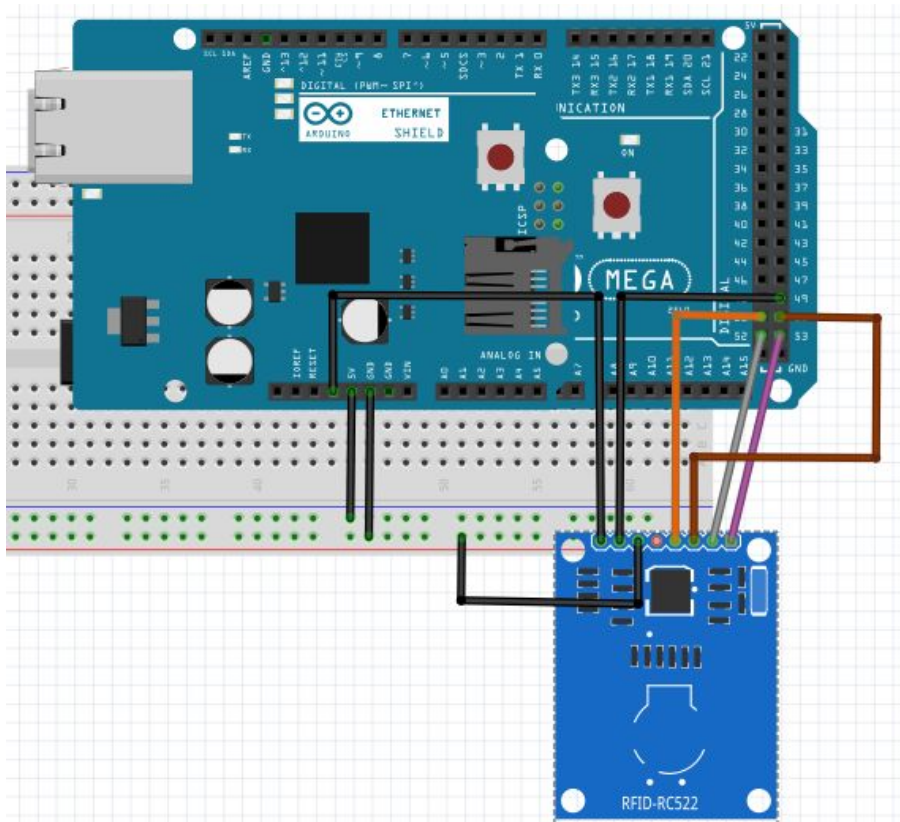


A continuación se detallan los esquemas de conexión de cada módulo usado en el prototipo. El shield ethernet es conecta encima del Arduino mega. Por lo tanto, se muestra en todos los esquemas conectado al arduino.

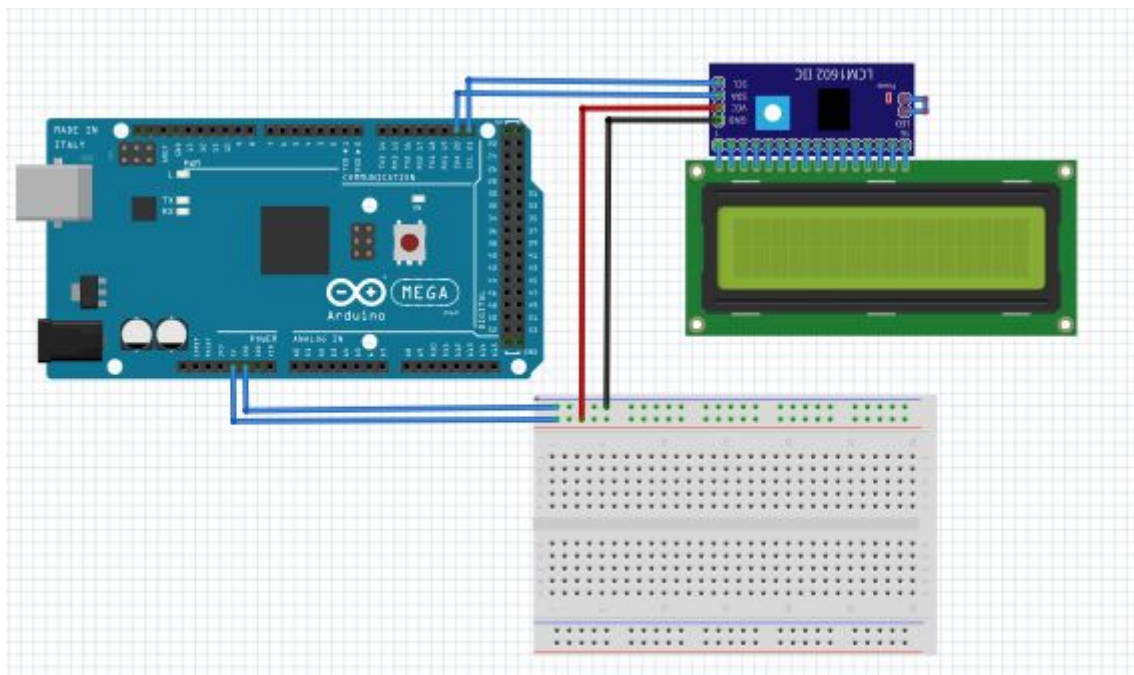
- Sensor de huella



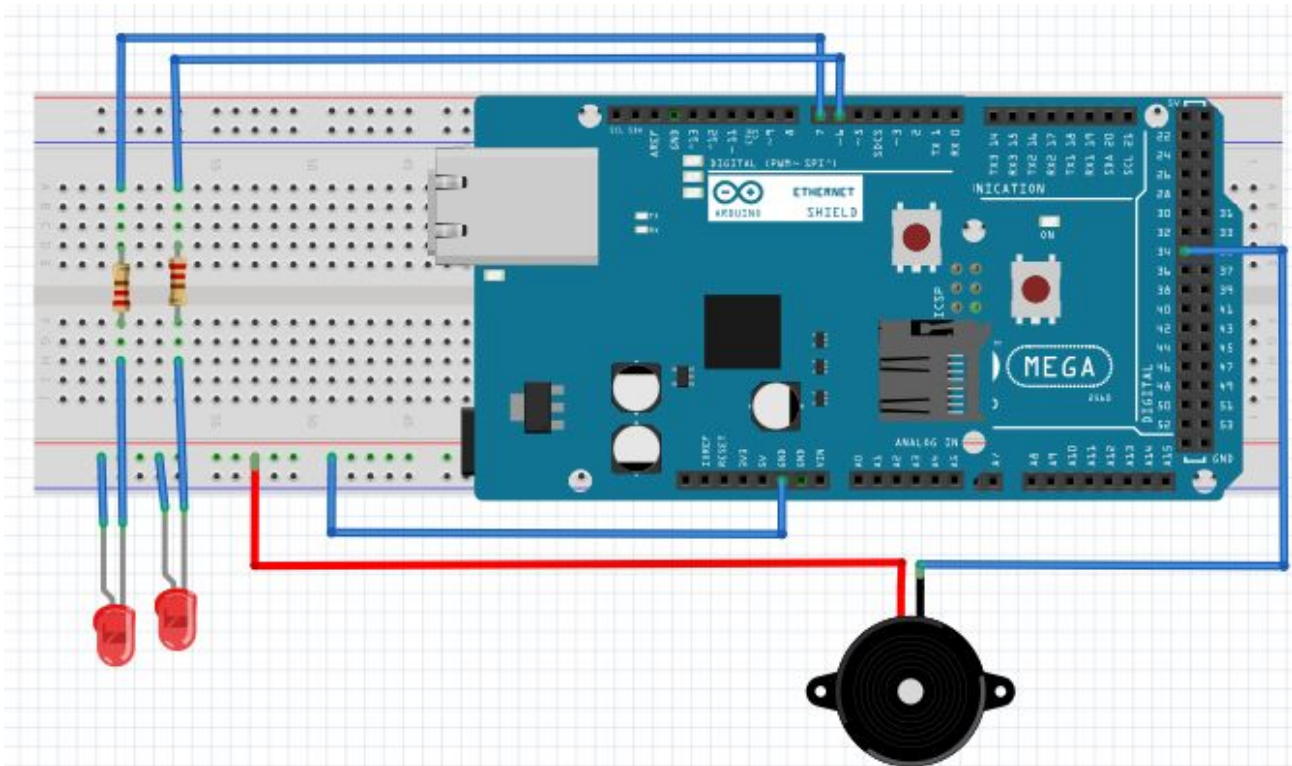
- Lector de tarjetas RF-ID



- Display



- Leds y buzzer



Todos los esquemas de conexión se encuentran en el repositorio GitHub creado dentro de la carpeta fritzings.

3.2 Software

Se debe clonar o descargar el código del repositorio, y abrir el archivo **main.ino** desde el IDE Arduino. Para que el código pueda compilarse, se deben descargar e instalar las librerías en el IDE. Para clonar el proyecto en una terminal linux ejecutar:

```
>>>git clone https://github.com/agrup/Arduino-control-access.git
```

Para instalar las librerías, primero deben descargarse en formato .zip a partir de los links proporcionados en la sección 2.3. Una vez descargados todos los archivos, se deben instalar en el IDE Arduino. Para esto en el IDE Arduino ir al menú Herramientas, Librerías, Instalar Librerías de un zip. Luego, seleccionar el archivo .zip de la librería que desea instalar.

Para información más detallada visitar el sitio: <https://www.arduino.cc/en/Guide/Libraries>.

La estructura del programa consiste en un directorio main. Dentro del directorio se encuentran subdirectorios que contienen archivos .h con las funciones de cada componente del prototipo, más el archivo main.ino que debe abrirse con el IDE Arduino.

El código fuente del archivo main.ino es muy sencillo y se presenta a continuación:

```
#include "sd/sd_fp.h"
#include "ethernet/ethernet.h"
#include "fingerprint/FingerPrint.h"
#include "display/Display.h"
#include "rf-id/Rfid.h"
#include "time/time.h"
#include "leds/led.h"

void setup()
{
  Serial.begin(9600); delay(100);
  ethernet_init(); delay(100);
  card_init(); delay(100);
  leds_init(); delay(100);
  time_init(); delay(100);
  display_init(); delay(100);
  finger_init();
}

void loop()
{
  print_time();
  ethernet_listen();
  write_display("Esperando... ",0,0);
  read_card();
  read_finger();
}
```

En la función setup se inicializan los módulos que utiliza el programa. La función loop primero imprime la hora, luego el server pregunta si hay peticiones que responder, y por último se leen las tarjetas y las huellas.

3.3 Puesta en marcha

Una vez que nuestro archivo `main.ino` puede compilarse con el IDE Arduino, debemos configurar la hora al programa. Para esto debe modificarse el archivo **main/time/time**. En la línea 17 puede configurarse la fecha y hora con la que comenzará el prototipo.

Para administrar el dispositivo, es necesario conectarse con una computadora personal a través de un cable de red. Para que el dispositivo sea alcanzable por la red en nuestra computadora, se debe modificar la ip de la interfaz de red de nuestra computadora. La ip debe pertenecer a la red 192.168.2.0/24. La ip 192.168.2.178 no puede utilizarse, puesto que esta es la ip del servidor en el dispositivo.

El siguiente comando en linux cambia la ip de la interfaz seleccionada:

\$ sudo ip addr add 192.168.2.1/24 dev <nombre-interfaz-red>

A partir de este momento es posible cargar nuestro programa al dispositivo. Enchufar la placa arduino a la computadora, incluyendo el cable de red. Subir el programa `main.ino` a la placa mediante el IDE Arduino.

Por último, ingresar a un browser e ingresar la ip 192.168.2.178. Deberá abrirse la interfaz web para administrar el prototipo.

4. Consideraciones finales.

El proyecto demuestra la posibilidad de construir un sistema de control de acceso de personal utilizando tanto tecnologías open hardware como open software. Este prototipo construido puede ser mejorado incluyendo funciones nuevas o adaptándolo a cualquier sistema de control de acceso.

A continuación se listan las posibles mejoras para trabajos futuros:

- Diseñar el modelo 3D para cerrar el dispositivo.
- Incluir otras funciones en la interfaz web, tales como descargar las fichadas o mostrar fichadas por número de legajo.
- Agregar un módulo wifi para no tener que utilizar un cable para acceder a la interfaz web.
- Integrar el prototipo a una cerradura eléctrica, mediante un actuador.

5. Referencias.

- <https://arduinojson.org/>
- <https://www.arduino.cc/reference/en/>
- <http://fritzing.org/home/>
- <https://learn.adafruit.com/adafruit-optical-fingerprint-sensor?view=all>
- <https://www.arduino.cc/en/Reference/Ethernet>
- <https://www.arduino.cc/en/Reference/SD>
- <https://www.arduino.cc/en/Reference/SPI>