

Universidad ORT Uruguay
Facultad de Ingeniería
Taller de tecnologías 1

Obligatorio 2

Integrantes:

Guillermo López - Nro. Estudiante – 235733

Tatiana Mostovaya Nro. Estudiante – 258450

Gastón Barlocco - Nro. Estudiante – 241025

Docente: Tomás De Angelis Rodríguez

2020

Índice

1.	Propuesta proyecto.....	3
1.	Introducción.	3
2.	Descripción del sistema actual.....	3
3.	Propuesta de solución.....	3
2.	Creación del prototipo – maqueta.	6
3.	Listado de materiales.	9
4.	Código.....	10
5.	Funcionalidades del prototipo.....	14
6.	Problemas encontrados durante el desarrollo y sus soluciones.....	15
7.	Diagrama de conexiones.	16
8.	Posibles mejoras.....	17
9.	Videos:	18

1. Propuesta proyecto.

1. Introducción.

Pensando en la cantidad de estacionamientos manuales y antiguos existentes en Montevideo, se nos ocurrió la idea de mejorar, modernizar y actualizar su automatismo para hacerlo más prácticos y útiles adaptándolos a los tiempos que corren.

2. Descripción del sistema actual.

Al hablar de un estacionamiento manual, básico, convencional, nos estamos refiriendo a los siguientes puntos:

- El ingreso y la salida del cliente se realiza con apertura de puerta/barrera manual.
- La iluminación interna y externa se comanda mediante una llave ON/OFF.
- El cliente no puede visualizar ningún aviso de lugares disponibles, entonces, se generan pérdidas de tiempo a la hora de buscar estacionar un vehículo en zonas transitadas.

3. Propuesta de solución.

A continuación, presentaremos la automatización completa del estacionamiento, para ello se tendrán en cuenta los siguientes puntos:

- 1- Se instalará un panel principal al frente del garaje para indicar la cantidad de lugares disponibles, también mostrará la temperatura actual. Las luces frontales se encenderán con ausencia de luz.

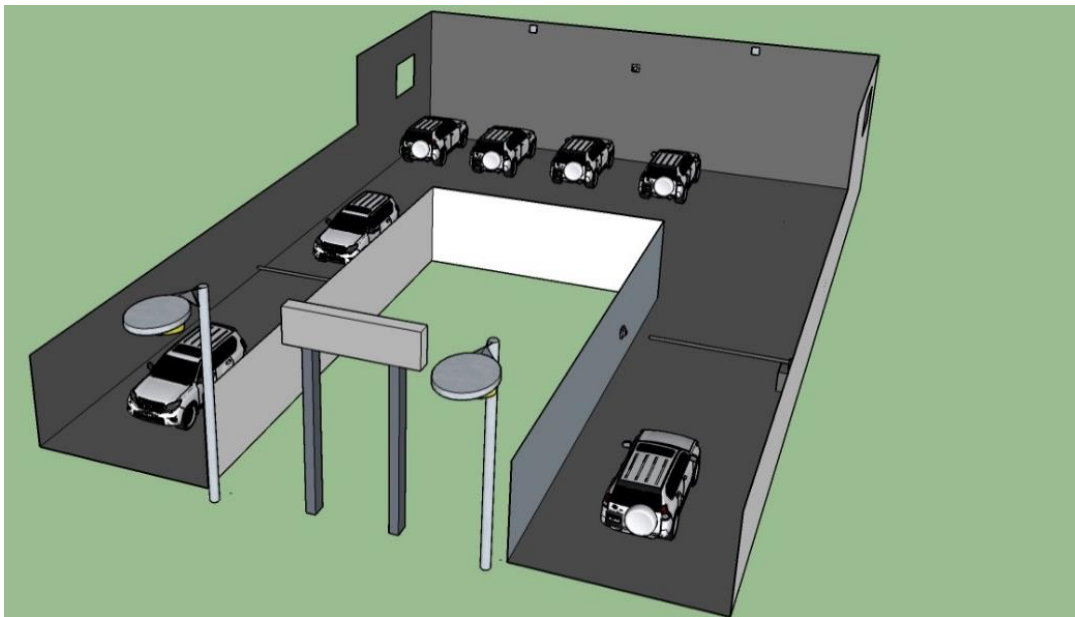


Fig. 1: Vista general del garaje.

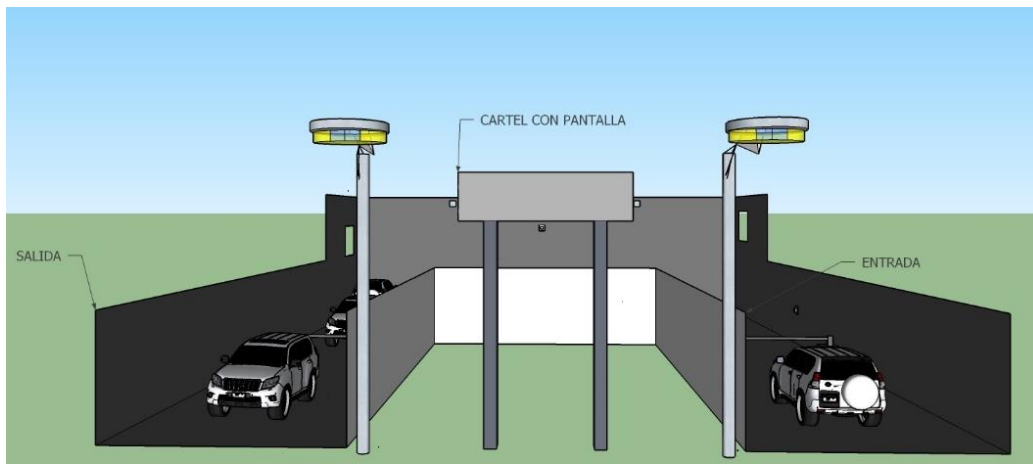


Fig. 2: Entrada y salida.

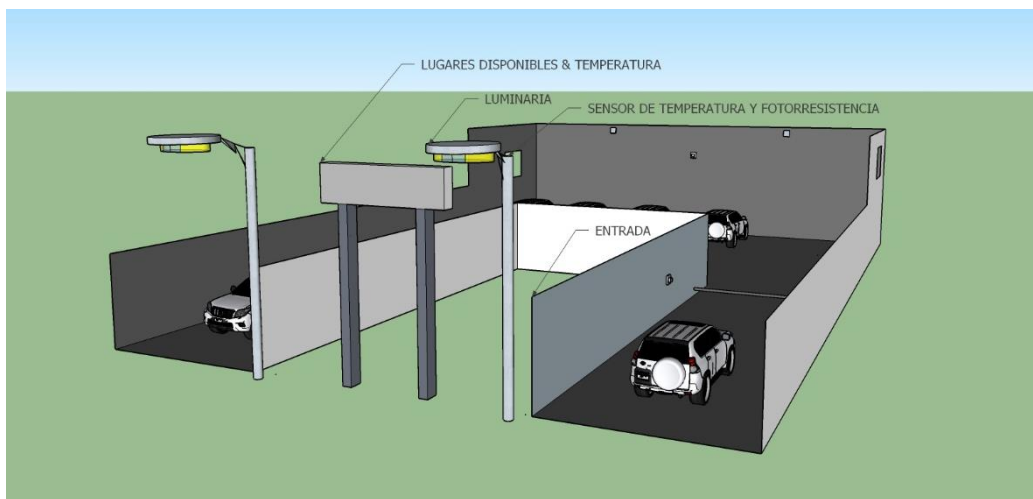


Fig. 3: Instalación de luces frontales, sensor de temperatura, sensor fotorresistencia.

- 2- La entrada de los vehículos será permitida mediante un pulsador, al accionar el mismo la barrera permite su paso (si existen lugares disponibles), cuando el vehículo se encuentra efectivamente dentro del garaje se actualiza el conteo de lugares disponibles.

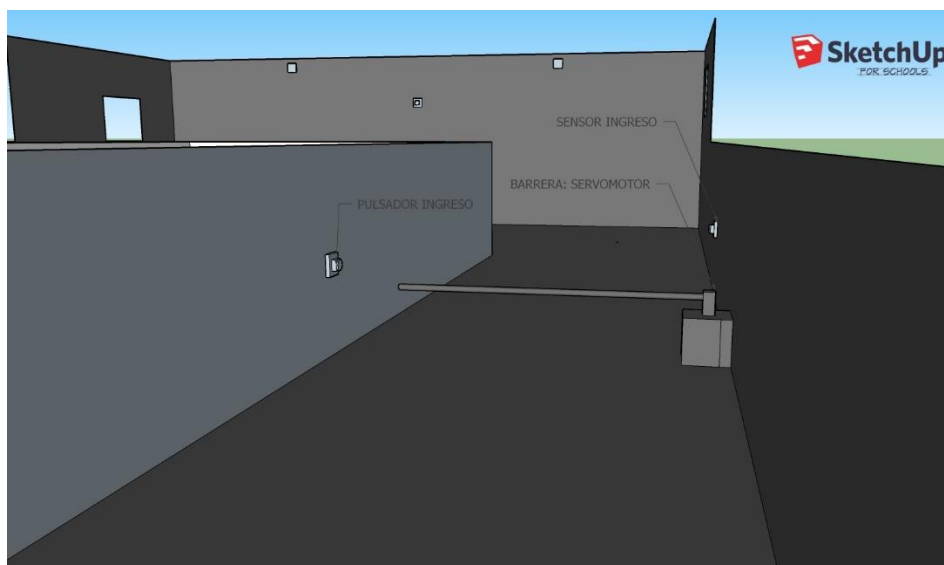


Fig. 4: Pulsador de entrada, barrera(servomotor), sensor de ingreso.

- 3- Se instalarán sensores de movimiento para activar la iluminación del garaje.

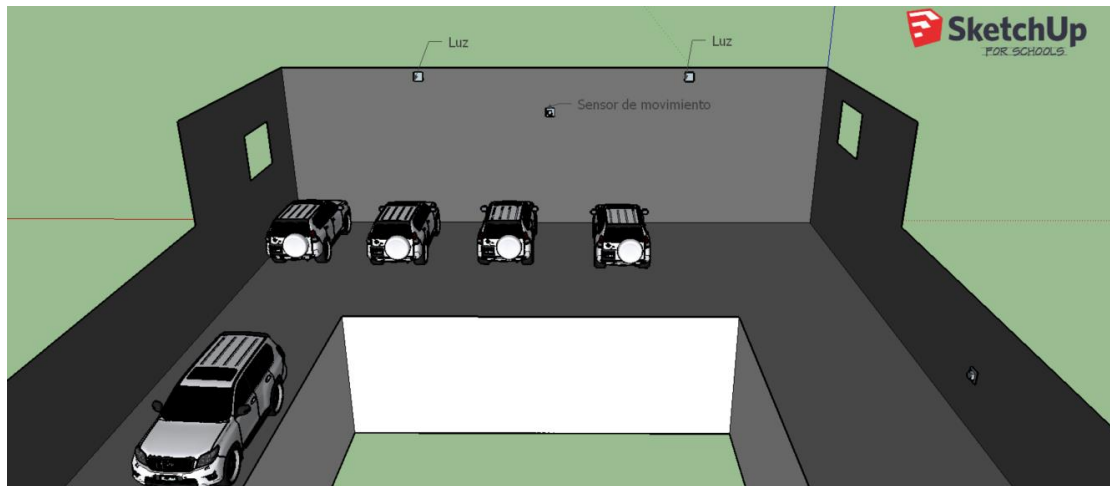


Fig. 5: Sensor de movimiento y luces.

- 4- La salida de los vehículos se realizará de la misma manera que la entrada (pulsador + sensor + barrera), cuando se confirme la salida efectiva de un vehículo se incrementarán los lugares disponibles.

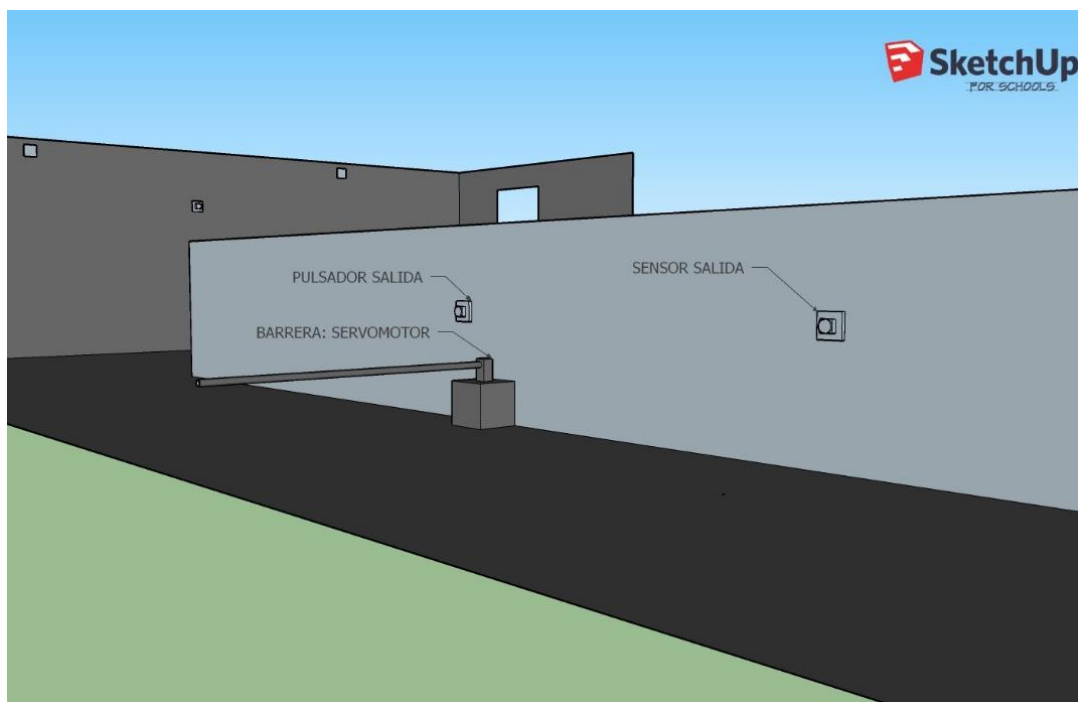


Fig. 6: Salida de vehículos.

2. Creación del prototipo – maqueta.

A continuación, se presentan secuencias de fotos sobre la maqueta:

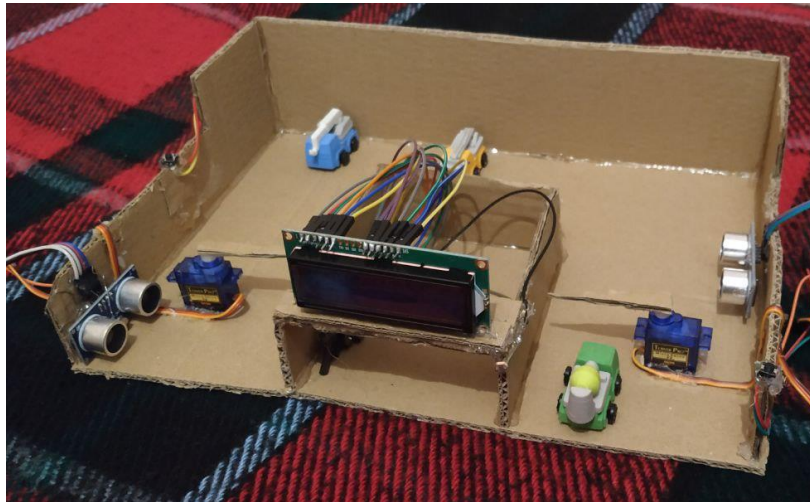


Fig. 7: Construcción de la maqueta.

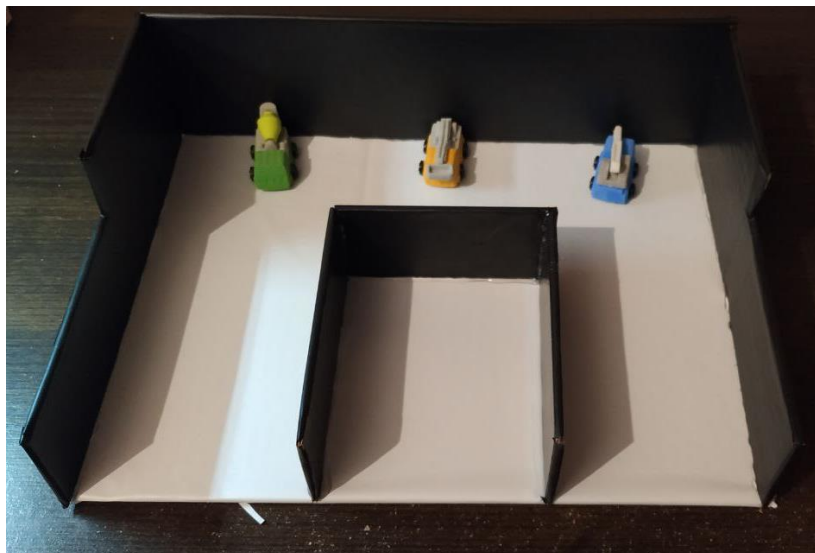


Fig. 8: Construcción de la maqueta.



Fig. 9: Construcción de la maqueta.

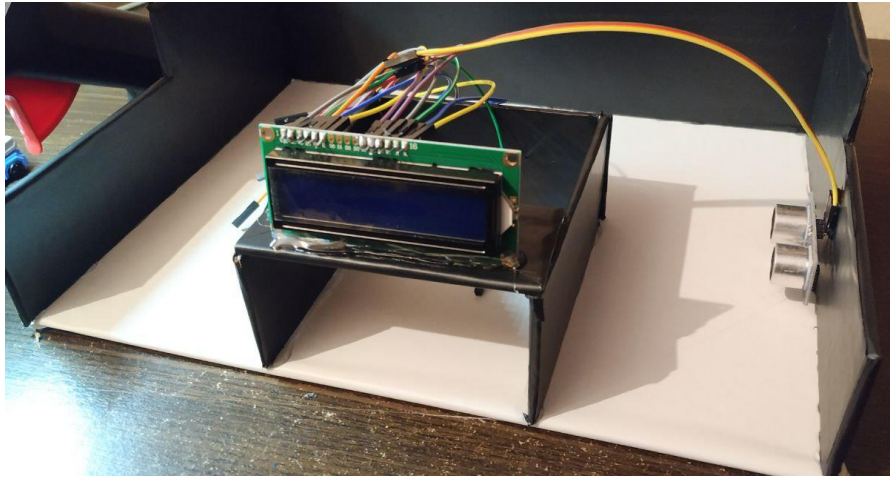


Fig. 10: Construcción de la maqueta.

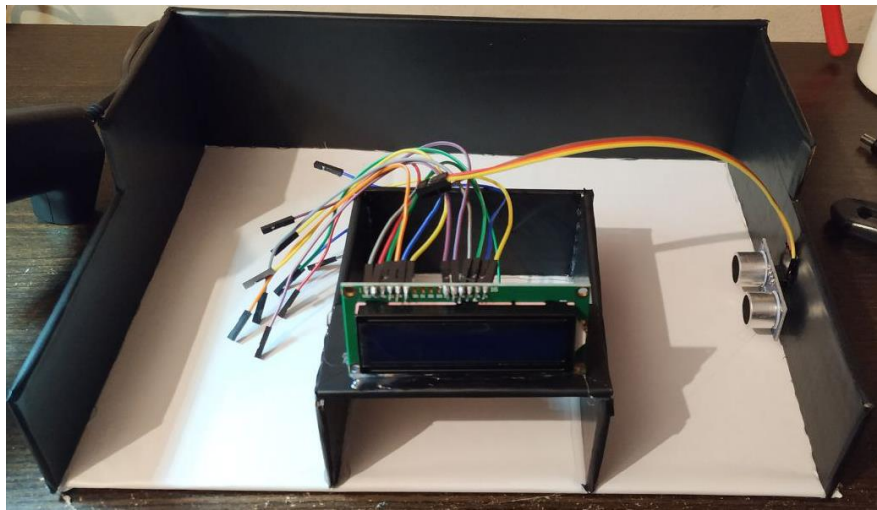


Fig. 11: Construcción de la maqueta.

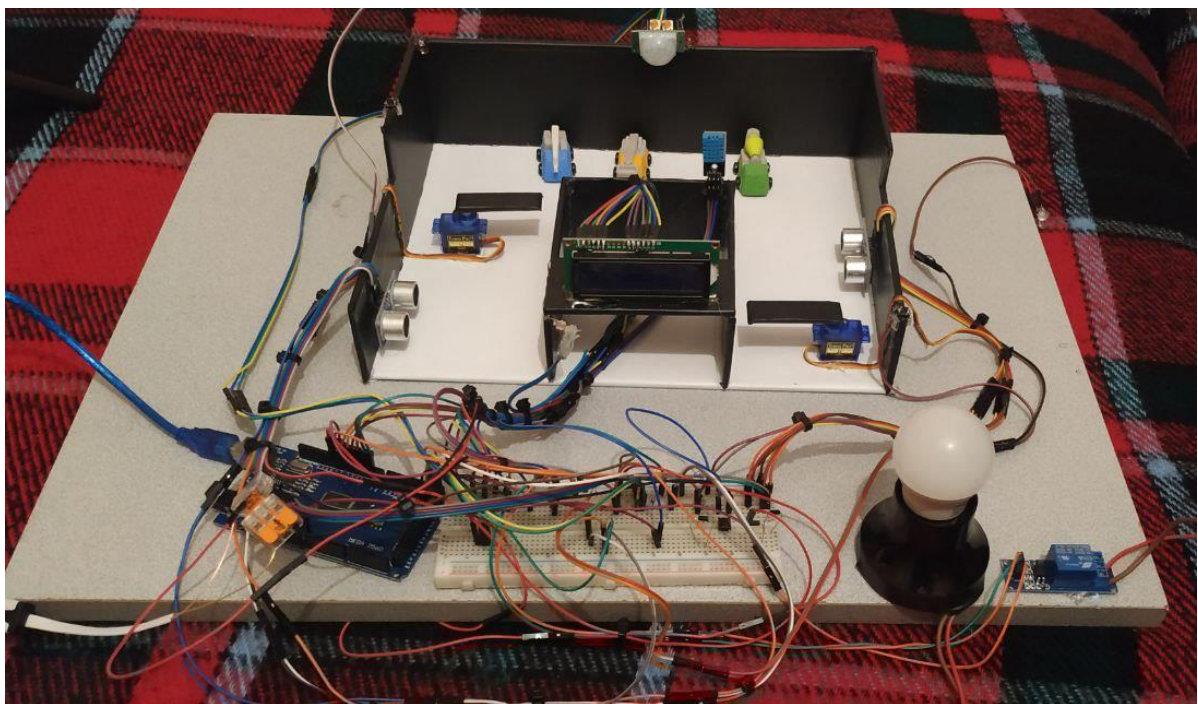


Fig. 12: Maqueta finalizada.

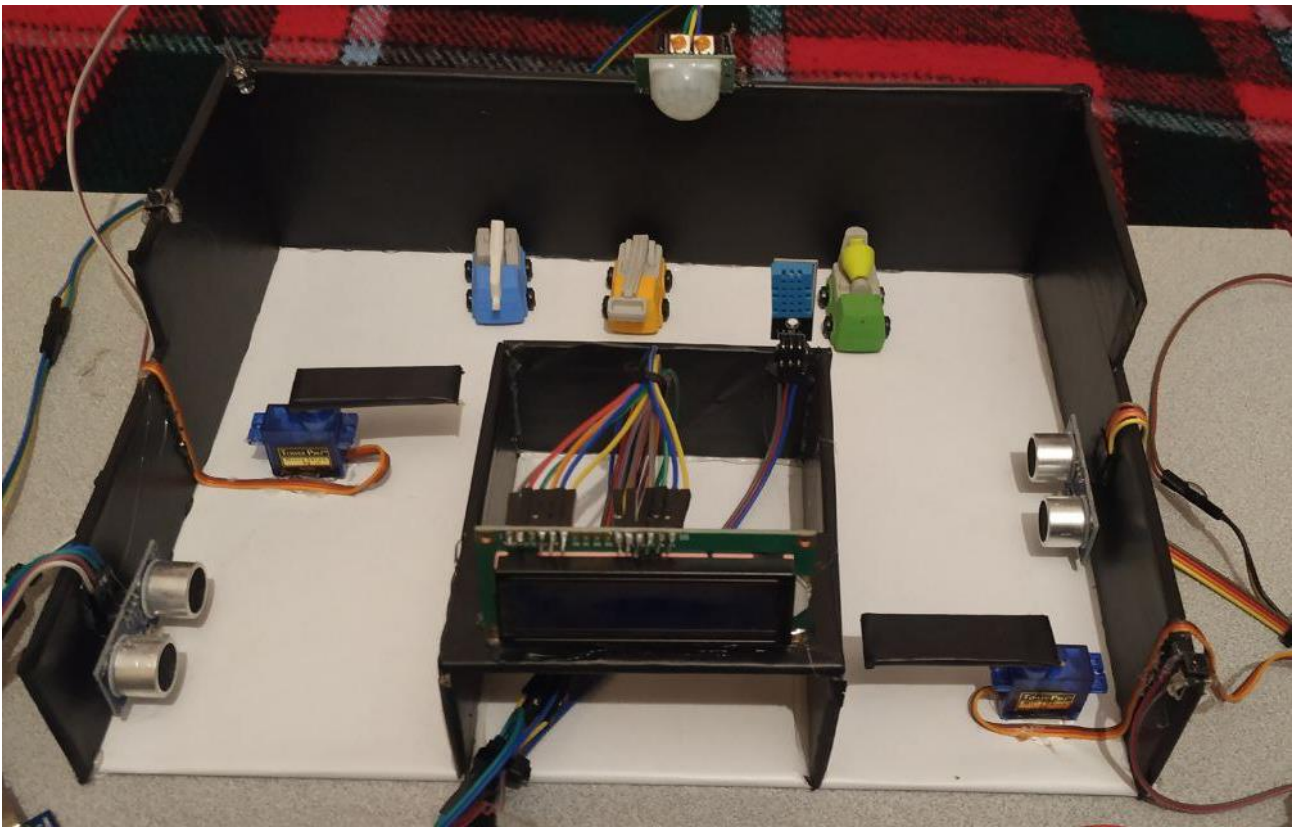


Fig. 13: Maqueta finalizada.

3. Listado de materiales.

- 1- Arduino MEGA.
- 2- Pulsadores.
- 2- Servomotores – SG90
- 2- Sensores ultrasónicos – HC-SR04
- 1 Sensor de temperatura y humedad – DHT-11
- 1- Pantalla LCD 16 x 2.
- 1- Sensor de movimiento – HC-SR-501
- 1- LDR.
- 1 – Lámpara 220 VAC.
- 1 – Relé.
- 1- Leds color blanco.
- 2- Resistencias pull down – 10k ohms.
- 1- Resistencia para LDR – 10k ohms.
- 1 Resistencia para LED - 220 ohms.
- 1- Protoboard
- Cables.

4. Código.

```
#include <LiquidCrystal.h>
#include <Servo.h>
#include <dht.h>
#define DHT11_PIN 8
dht DHT;

LiquidCrystal lcd(7, 6, 5, 4, 3, 2);
Servo barreraEntrada;
Servo barreraSalida;

int intensidadLuz = A0;
int intensidad =0;

int pulsadorIngreso = 22;
int pulsadorSalida = 23;

int lucesExterior =28;

int echoSensorEntrada = 24;
int triggSensorEntrada = 25;

int echoSensorSalida = 26;
int triggSensorSalida = 27;

int lucesInternas =29;
int sensorMovimiento =30;

float distanciaCm =0;
float tiempoSensor =0;

int maxAutos =5;
int contAutos =0;

bool seAbrioBarreraEntrada = false;
bool seAbrioBarreraSalida = false;
bool estacionamientolleno = false;

unsigned long tiempoTemp = 0;

void setup() {
  Serial.begin(9600);
  lcd.begin(16, 2);
  lcd.clear();

  barreraEntrada.attach(10);
  barreraSalida.attach(9);

  pinMode(pulsadorIngreso,INPUT);
  pinMode(pulsadorSalida,INPUT);
  pinMode(echoSensorEntrada, INPUT);
  pinMode(echoSensorSalida, INPUT);
  pinMode(sensorMovimiento, INPUT);

  pinMode(triggSensorEntrada, OUTPUT);
  pinMode(triggSensorSalida, OUTPUT);
  pinMode(lucesExterior, OUTPUT);
  pinMode(lucesInternas, OUTPUT);

  inicio();
}

void loop() {
```

```

    medirTemperatura();
    entradaAutos();
    salidaAutos();
    luzFrenteGaraje();
    luzInternaGaraje();
    verificarDisponibilidadEstacionamiento();
    delay (50);
}

void inicio(){
    cerrarBarreraEntrada();
    cerrarBarreraSalida();
    mostrarCantidadAutos();
    apagarLuces();
}

void entradaAutos(){
    if (digitalRead(pulsadorIngreso)==HIGH && contAutos < maxAutos){
        seAbrioBarreraEntrada = true;
        abrirBarreraEntrada();
        delay (10);
    }
    if (seAbrioBarreraEntrada){
        realizarMedicion(triggSensorEntrada,echoSensorEntrada);
    }

    if (distanciaCm <= 5 && seAbrioBarreraEntrada){
        entroAuto();
    }
    delay (10);
}

void salidaAutos(){
    if (digitalRead(pulsadorSalida)==HIGH && contAutos >= 1){
        seAbrioBarreraSalida = true;
        abrirBarreraSalida();
        delay (10);
    }
    if (seAbrioBarreraSalida){
        realizarMedicion(triggSensorSalida, echoSensorSalida);
    }

    if (distanciaCm <= 5 && seAbrioBarreraSalida ){
        salioAuto();
    }
    delay (10);
}

void medirTemperatura(){
    if (millis() >tiempoTemp + 2000){
        DHT.read11(DHT11_PIN);
        tiempoTemp = millis();
        mostrarTemperatura();
    }
}

float realizarMedicion(int trigger, int echo ){
    digitalWrite (trigger, LOW);
    delayMicroseconds(5);
    digitalWrite (trigger, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigger, LOW);
    tiempoSensor = pulseIn(echo, HIGH);
    distanciaCm =(0.01715*tiempoSensor);
    return distanciaCm;
}

```

```

}

void mostrarCantidadAutos(){
  lcd.setCursor(0, 0);
  if (estacionamientoLleno){
    lcd.print("          ");
    lcd.setCursor(0, 0);
    lcd.print("SIN LUGARES");
  }else{
    lcd.print("          ");
    lcd.setCursor(0, 0);
    lcd.print("AUTOS: ");
    lcd.print(contAutos);
  }
}

void mostrarTemperatura(){
  lcd.setCursor(0, 1);
  lcd.print("T(C)= ");
  lcd.print(DHT.temperature);
}

void entroAuto(){
  contAutos ++;
  mostrarCantidadAutos();
  delay (10);
  cerrarBarreraEntrada();
  seAbrioBarreraEntrada = false;
}

void salioAuto(){
  contAutos --;
  mostrarCantidadAutos();
  delay (10);
  cerrarBarreraSalida();
  seAbrioBarreraSalida = false;
}

void luzFrenteGaraje(){
  intensidad = analogRead (intensidadLuz);

  if (intensidad >700){
    digitalWrite(lucesExterior,LOW);
  }else{
    digitalWrite(lucesExterior,HIGH);
  }
}

void luzInternaGaraje(){
  if (digitalRead(sensorMovimiento)==HIGH){
    digitalWrite (lucesInternas, HIGH);
    delay (10);
  }else{
    digitalWrite (lucesInternas, LOW);
  }
}

void apagarLuces(){
  digitalWrite (lucesInternas, LOW);
  digitalWrite(lucesExterior,LOW);
}

void verificarDisponibilidadEstacionamiento(){
  if (contAutos == maxAutos){
    estacionamientoLleno = true;

```

```
        mostrarCantidadAutos();
    }else{
        estacionamientoLleno = false;
        mostrarCantidadAutos();
    }
}

void abrirBarreraEntrada(){
    barreraEntrada.write(61);
}

void cerrarBarreraEntrada(){
    barreraEntrada.write(151);
}

void abrirBarreraSalida(){
    barreraSalida.write(61);
}

void cerrarBarreraSalida(){
    barreraSalida.write(151);
}
```


5. Funcionalidades del prototipo.

A continuación, se explicará el funcionamiento del prototipo, para entender mejor estas explicaciones mirar el *video solución*:

Ingreso de vehículos:

- Para que un cliente pueda ingresar al estacionamiento el mismo debe tener lugares disponibles, en caso de ser así, al presionar el pulsador se abre la barrera (servomotor), luego de detectar su ingreso (sensor de distancia) se incrementa y muestra en la pantalla LCD la cantidad de autos en el estacionamiento.

Salida de vehículo:

- La salida del cliente es análoga a la entrada, se debe presionar un pulsador, luego el sensor de distancia confirmara la salida. Al detectar el retiro de un vehículo del estacionamiento se restará el conteo de autos disponibles.

Dentro del estacionamiento:

- Se da el accionamiento automático de luces internas en base a presencia de movimiento.

Luces externas:

- Las luces externas se activan con ausencia de iluminación, entonces, de esta manera se tendrá iluminación automática en la entrada del estacionamiento cuando sea de noche. La intensidad de luz es medida con un LDR.

Información brindada en la pantalla LCD:

- Se mostrará de forma actualizada la temperatura actual y la cantidad de autos dentro del estacionamiento. Cuando el estacionamiento se encuentra lleno se informará por la pantalla, de esta manera el cliente no perderá tiempo.

6. Problemas encontrados durante el desarrollo y sus soluciones.

- A la hora de conectar la lámpara de 220 V mediante un relé nos encontramos con problemas de ruido eléctrico, notamos esto ya que al conectar la lámpara se nos apagaba el arduino. Para solucionar este problema alejamos los cables que alimentaban la lámpara del arduino y colocamos un filtro RC ($R=100\Omega$ 3W, $C=100\text{nF}$).

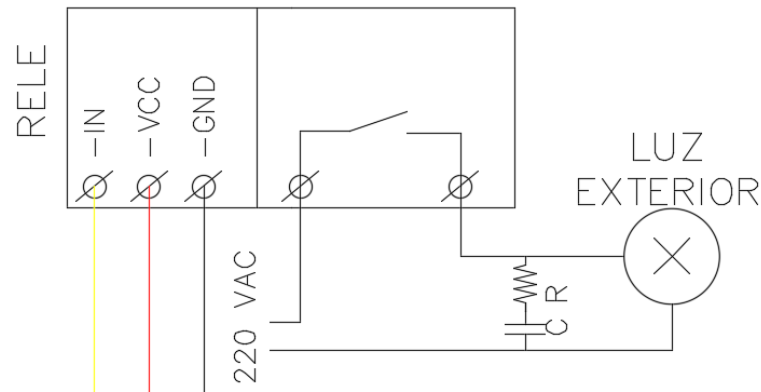
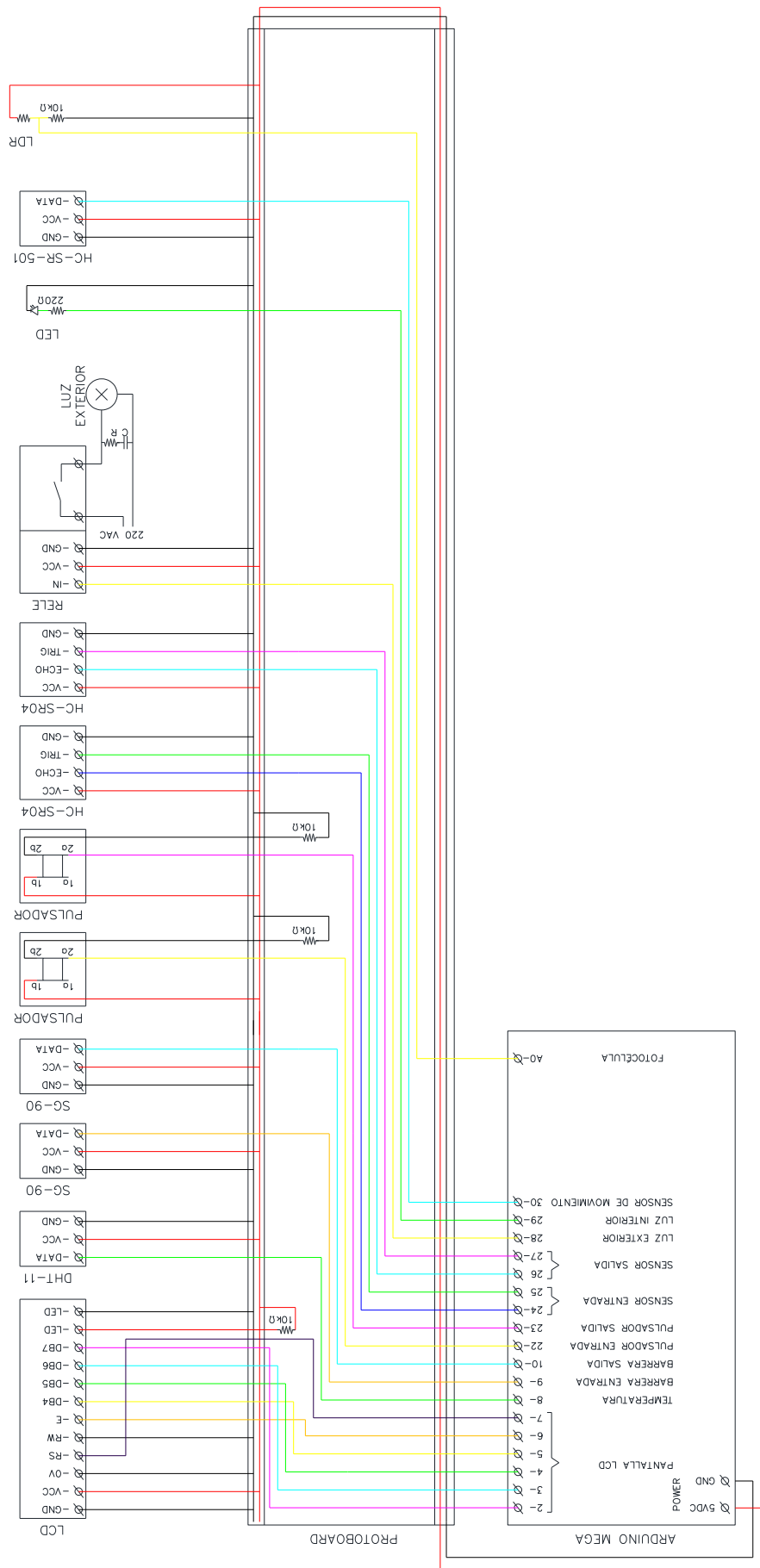


Fig. 14: Filtro para ruidos eléctricos.

7. Diagrama de conexiones.



8. Posibles mejoras.

- Se podría instalar un sistema de identificación por tarjeta, o acceso al estacionamiento mediante contraseña para brindar mayor seguridad.
- Envío de mensaje al dueño del estacionamiento en caso de presenciarse alguna alarma (robo, incendio, etc.)
- Registrar histórico de ingreso y egreso de clientes para tener mayor control en caso de algún problema. Además, se podrían agregar cámaras para filmar el predio y poder observar las mismas en caso de que se presente un choque o cualquier tipo de problema.

9. Videos:

- Video Problema – Solución:

<https://www.youtube.com/watch?v=Yz3UO1Cbk8w>

- Video solución:

<https://www.youtube.com/watch?v=5XfH6EyDJuA&t=3s>