

SALESIANOS ATOCHA

ADMINISTRACIÓN DE SISTEMAS INFORMÁTICOS EN RED



“ DOMOPI ”

MEMORIA DE PROYECTO DE FINAL DE CURSO

TUTORA DE PROYECTO: EVA MARÍA MARTÍNEZ FERNÁNDEZ

ALUMNO: DANIEL GARCÍA-ALCAIDE CABRERA

En Atocha, a 12 de Marzo de 2017

Proyecto DomoPi

Descripción del proyecto	1
Materiales.....	2
Pasos.....	3
Página web.....	3
Montaje de la casa	5
Montaje del servidor.....	6
Montaje de luces.....	8
Montaje de persianas	10
Montaje de puertas	12
Montaje de la puerta de garaje	12
Montaje del sensor de temperatura.....	13
Problemas que han aparecido	15

Descripción del proyecto

DomoPi es una maqueta de una casa automatizada con cuatro habitaciones y un garaje realizada en madera de DM (cartón prensado), pintada en color Caoba y manejada desde nuestro teléfono móvil o PC.

Los sistemas que automatizan la casa son Raspberry Pi 3 a la cual está conectada una placa de Arduino Uno mediante un Hub USB.

Con estos sistemas se puede controlar las luces de toda la casa de forma independiente, también como las puertas, las persianas y la puerta del garaje, la cual mientras se abre y se cierra, la luz de dentro se mantendrá encendida.

Además de todo esto, la casa dispone de un display led que marca la temperatura actualizada cada minuto.

Por la parte de seguridad, la casa tiene una puerta que la cubre entera con un motor como cerradura, que, conectado a un panel numérico, cuando la contraseña sea correcta el motor desbloqueará la puerta.

Por último, está la página web donde podremos controlar cada parte de la casa por separado, pero solo un usuario perteneciente a la casa podrá entrar a la web, todo esto programado con HTML, PHP y con estilo en CSS con su correspondiente Responsive Design adaptado tanto para el ordenador como para el móvil.

Materiales

- **Madera de DM**
Que será toda la estructura de la casa.
- **Raspberry Pi 3**
Que actuará de servidor web.
- **Arduino Uno**
Será lo que controle todos los componentes electrónicos.
- **Servomotores**
Moverá puertas y ventanas.
- **Leds**
Todo el alumbrado de la casa.
- **Ruedas y tiras dentadas**
Para el sistema de las puertas.
- **Protoboard**
Para la conexión del cableado.
- **Resistencias**
Para controlar la tensión.
- **Sensor de temperatura**
Para medir la temperatura actual de la casa.

Pasos

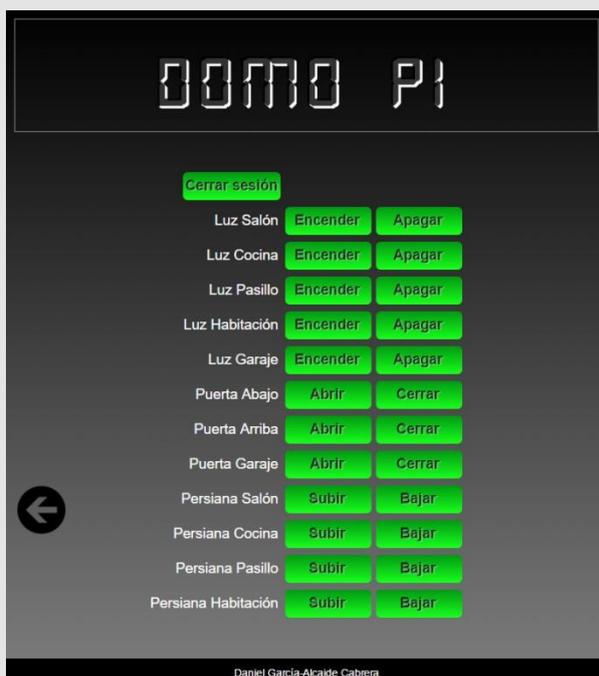
▪ Página web

Lo primero que he hecho es la página web, la cual está formada por HTML, CSS, JavaScript y PHP.

Mi idea es que solo hay un único usuario, como en una casa real, nadie puede registrarse, tienes tu usuario con tu contraseña encriptada y listo. Por lo tanto, la página lo primero que hace es pedirte el usuario y contraseña, una vez dentro, podremos visualizar la fecha y hora actual, la temperatura en ese momento de la casa y otro botón al que podremos acceder a los controles de la casa.



En esos controles lo que se puede hacer es encender y apagar las luces de manera independiente, al igual que las puertas y las ventanas.



Los controles que se manejan desde la web, están a su vez controlando un archivo Python el cual manda una señal a la placa Arduino para que el programa creado en Arduino se ejecute, lo explicaré más adelante.

```
$luz1e = $_REQUEST['luz1e'];
$luz1a = $_REQUEST['luz1a'];

if (isset($luz1e))
{
    $luz1 = exec('sudo python ./enciendeled.py');
    echo $luz1;
}
if (isset($luz1a))
{
    $luz2 = exec('sudo python ./apagaled.py');
    echo $luz2;
}
```

Ejemplo de código PHP que llama a un Python llamado **enciendeled.py** que enciende un led y a otro llamado **apagaled.py** que lo apaga.

Nota: Además esta página web está hecha con Responsive Design, diseñada para ordenador, Tablet y móvil.

▪ Montaje de la casa

La idea de la casa no me vino a la cabeza así de repente, sino que usando el programa SketchUp de google para modelado 3D fui creando poco a poco, hasta que se me fueron ocurriendo ideas y yo las iba plasmando en el modelo.

Así quedó el modelo final hecho con las medidas reales:



Se pueden observar todas las luces, las dos puertas correderas, las cuatro ventanas y la puerta de garaje.

Todo el cableado, va justo encima del garaje.

Así que una vez decidido el modelo, pasamos a la acción, y después de dedicarle incontables horas cortando, limando y barnizando, he aquí el resultado:



▪ Montaje del servidor

Ahora lo siguiente es montar el servidor web en la Raspberry Pi 3 con sistema Raspbian descargado en: <https://www.raspberrypi.org/downloads/raspbian/>

Lo primero es dar crear y dar permisos al usuario de apache (sevidor web):

- `sudo addgroup www-data`
- `sudo usermod -a -G www-data www-data`

Ahora instalamos apache, MySQL y PHP:

- `sudo apt-get install apache2 php5 libapache2-mod-php5`
- `sudo apt-get install mysql-server mysql-client php5-mysql`
- `sudo apt-get install php5 libapache2-mod-php5 libapache2-mod-perl2 php5-cli php5-cgi php5-common php5-curl`

Ahora se reinicia la Raspberry Pi y guardo en la carpeta `/var/www/html` la carpeta con toda mi página web, de esta manera ya puedo acceder a ella desde mi ordenador que está en la misma red.

También he cambiado los permisos y los propietarios y los grupos de todos los archivos y quedó de la siguiente manera (además de dar permisos a `www-data` en *sudoers*):

```
-rwxrwxrwx 1 root root      125 mar  8 14:15 apagaled11.py
-rwxr-xr-x 1 root root      125 mar  8 14:16 apagaled2.py
-rwxr-xr-x 1 root root      125 mar  8 14:16 apagaled3.py
-rwxr-xr-x 1 root root      125 mar  8 14:16 apagaled4.py
-rwxr-xr-x 1 root root      125 mar  4 16:30 apagaled5.py
-rwxrwxrwx 1 root root      125 mar  8 14:15 apagaled.py
-rw-rw-r-- 1 root www-data 6657 feb 21 11:38 atras.png
-rw-rw-r-- 1 root www-data 5490 mar 10 22:57 control.php
-rw-r--r-- 1 pi   www-data 5428 mar 10 22:57 control.php~
-rwxrwxrwx 1 root root      137 mar 10 22:44 enciendeled11.py
-rwxr-xr-x 1 root root      136 mar  8 14:15 enciendeled2.py
-rwxr-xr-x 1 root root      136 mar  8 14:16 enciendeled3.py
-rwxr-xr-x 1 root root      136 mar  8 14:16 enciendeled4.py
-rwxr-xr-x 1 root root      136 mar  4 16:30 enciendeled5.py
-rwxrwxrwx 1 root root      137 mar 10 22:55 enciendeled.py
-rw-r--r-- 1 pi   pi        4188 mar  4 16:50 estilo1.css
-rw-r--r-- 1 pi   pi        2748 mar  4 16:49 estilo2.css
-rw-rw-r-- 1 root www-data 134393 feb 21 11:38 fader.png
drwxrwxr-x 2 root www-data  4096 feb 27 16:48 fuente
-rw-r--r-- 1 pi   pi        2497 mar  4 16:07 index.php
-rw-r--r-- 1 pi   pi        2113 mar  3 21:55 menu.php
-rwxr-xr-x 1 root root      136 mar  8 14:08 puerta1.py
-rwxr-xr-x 1 root root      136 mar  8 14:08 puerta2.py
-rwxrwxrwx 1 root root      251 mar 10 22:45 raspduino.py
-rw-rw-r-- 1 root www-data 167513 feb 21 11:38 termometro.png
-rwxr-xr-x 1 root root      136 mar 10 20:09 ventana1.py
-rwxr-xr-x 1 root root      136 mar 10 20:09 ventana2.py
```

Ahora antes de explicar los archivos .py que se ven en la imagen de arriba, voy a mostrar el programa que he usado para programar el Arduino.

Primero instalamos el programa Arduino IDE:

- `sudo apt-get install arduino arduino-core`

Ahora comprobamos el puerto USB donde está conectado el cable serial:

- `dmesg | grep ttyACM`

Si por ejemplo el puerto es `ttyACM0` hacemos lo siguiente:

- `sudo chmod 666 /dev/ttyACM0`

Ahora desde la Raspberry ya se puede programar el Arduino, y este son un par de ejemplos de la programación:

```
char c;
const int led=12;
const int led2=11;
const int led3=10;
const int led4=9;
const int led5=8;

void setup ()
{
  pinMode(led, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
  pinMode(led4, OUTPUT);
  pinMode(led5, OUTPUT);
  Serial.begin(9600);
}

void loop ()
{
  if (Serial.available())
  {
    char c = Serial.read();
    if (c == 'A')
    {
      digitalWrite(led,HIGH);
    }
    else if (c == 'B')
    {
      digitalWrite(led,LOW);
    }
  }
}
```

Aquí he definido que cuando se inserte una 'A' se encenderá un led y cuando se inserte una 'B' se apagará.

Ahora lo siguiente es los archivos .py de los que he hablado antes, en los que se programará que cuando se ejecute cada archivo, será como escribir esa 'A' o 'B' de antes. Lo explicaré en detalle en su determinado apartado.

▪ Montaje de luces

En la madera he hecho dos taladros, el primero con una broca más fina que traspase la madera, y el otro la broca un poco más gorda sin llegar a traspasar del todo para que el led entre, pero no pueda salirse.

Su programación en el Arduino es de la siguiente manera:

```
char c;
const int led=12;

void setup ()
{
  pinMode(led, OUTPUT);
  Serial.begin(9600);
}

void loop ()
{
  if (Serial.available())
  {
    char c = Serial.read();
    if (c == 'A')
    {
      digitalWrite(led,HIGH);
    }
    else if (c == 'B')
    {
      digitalWrite(led,LOW);
    }
  }
}
```

Ahora tengo que hacer el archivo .py que le mande esa letra que pide el programa Arduino:

```
import serial
import time
arduino = serial.Serial('/dev/ttyACM0',9600)
comando='A'
if comando:
    arduino.write(comando)
arduino.close()
```

Esto manda la letra 'A'.

```
import serial
import time
arduino = serial.Serial('/dev/ttyACM0',9600)
comando='B'
if comando:
    arduino.write(comando)
arduino.close()
```

Esto manda la letra 'B'.

Y por último hay que mandar esta señal desde la página web, así que en el archivo .php hacemos lo siguiente:

```
<?PHP

$luz1e = $_REQUEST['luz1e'];
$luz1a = $_REQUEST['luz1a'];

if (isset($luz1e))
{
    $luz1 = exec('sudo python ./enciendeled.py');
    echo $luz1;
}
if (isset($luz1a))
{
    $luz2 = exec('sudo python ./apagaled.py');
    echo $luz2;
}

?>
```

Y dentro del <form> ponemos el botón:

```
<tr align=right>
    <td><h2>Luz Salón</h2>
    <td><input type=submit value="Encender" name=luz1e>
    <td><input type=submit value="Apagar" name=luz1a>
</tr>
```

Ahora ya se puede encender y apagar la luz desde la página web (este proceso es el mismo para las demás luces).

▪ Montaje de persianas

Para el montaje de las persianas le pegué al servomotor una barra de acero al que iría pegada la tela (le doy las gracias a mi madre por coserme las telas).

Para la conexión del servo a la placa hay que conectar el cable marrón a GND, el rojo a 5V y el amarillo a un pin.

Ahora su programa en Arduino es de la siguiente manera:

```
#include <Servo.h>

Servo myservo; // crea el objeto servo
int vel = 0; // velocidad del servo
char c;

void setup ()
{
  myservo.attach(7); // vincula el servo al pin digital 7
  vel = 49;
  myservo.write(vel);
  Serial.begin(9600);
}

void loop ()
{
  if (Serial.available())
  {
    char c = Serial.read();
    if(c == 'K')
    {
      vel = 90;
      myservo.write(vel);
      delay(1000);
      vel = 49;
      myservo.write(vel);
    }

    else if(c == 'L')
    {
      vel = 0;
      myservo.write(vel);
      delay(1000);
      vel = 49;
      myservo.write(vel);
    }
  }
}
```

Lo siguiente es el archivo Python:

```
import serial
import time
arduino = serial.Serial('/dev/ttyACM0',9600)
comando='K'
if comando:
    arduino.write(comando)
arduino.close()
```

Esto manda la letra 'K'.

```
import serial
import time
arduino = serial.Serial('/dev/ttyACM0',9600)
comando='L'
if comando:
    arduino.write(comando)
arduino.close()
```

Esto manda la letra 'L'.

Ahora para que funcione desde la web, el código en el archivo .php es el siguiente:

```
<?PHP
$ventanala = $_REQUEST['ventanala'];
$ventanalc = $_REQUEST['ventanalc'];

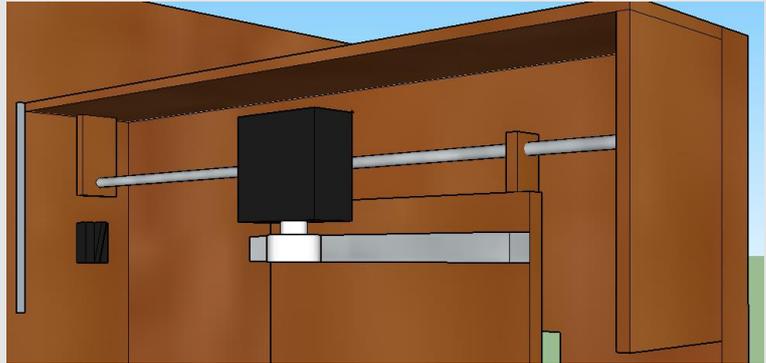
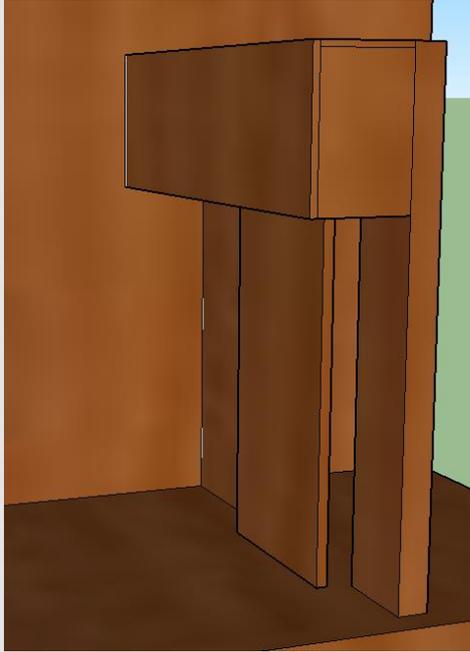
if (isset($ventanala))
{
    $ventana1 = exec('sudo python ./ventana1.py');
    echo $puerta1;
}
if (isset($ventanalc))
{
    $ventana2 = exec('sudo python ./ventana2.py');
    echo $ventana2;
}
?>
```

Ahora dentro del <form> ponemos el botón:

```
<tr align=right>
<td><h2>Persiana Cocina</h2>
<td><input type=submit value="Subir" name=ventanala>
<td><input type=submit value="Bajar" name=ventanalc>
</tr>
```

▪ Montaje de puertas

Para el montaje físico de las puertas he hecho un modelo 3D para estudiar como montarlo:



Todo lo demás, desde el cableado a Arduino, como todos los programas son idénticos a los de las ventanas.

▪ Montaje de puerta de garaje

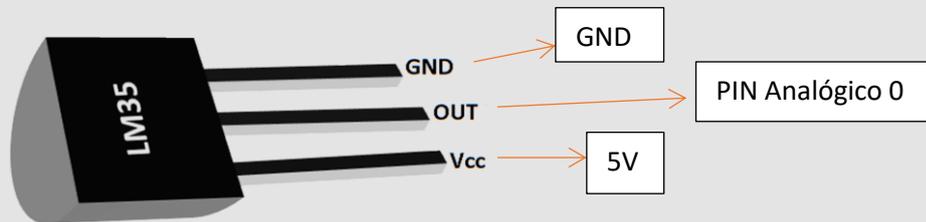
La puerta de garaje está dividida en dos partes, la parte de arriba más pequeña que la de abajo.

En la parte de abajo he puesto una sujeción a la que he atado un hilo de nailon y está conectado al motor, que este, al tirar, lo va recogiendo y la puerta sube.

El cableado y funcionamiento es idéntico al de la puerta y ventana.

▪ Montaje del sensor de temperatura

Lo primero es la conexión a la placa Arduino:



Ahora voy a programarlo en Arduino de la siguiente manera:

```
int analog_pin = 0;
float temperatura;

void setup ()
{
  Serial.begin(9600);
}
void loop()
{
  temperatura = analogRead(analog_pin);
  temperatura = 5.0*temperatura*100.0/1024.0;
  Serial.print(temperatura);
  Serial.println(" Grados");
  delay(5000);
}
```

Ahora tenemos que hacer un programa Python para recoger en una variable la temperatura que el programa anterior nos da:

```
import serial
PuertoSerie = serial.Serial('/dev/ttyACM0',9600)

if True:
    sArduino = PuertoSerie.readline()
    print sArduino.rstrip('\n')
arduino.close()
```

Lo último que falta es hacer que la página PHP active el script y saque la temperatura en su debido <div>:

```
<?php
    $temp = exec('sudo python ./temperatura.py');
    $tempbien=substr($temp,0,2);
    echo "<h2>$tempbien oC</h2>";
?>
```

Problemas que han aparecido

El primer problema que he tenido fue con las luces, conseguía que desde el terminal de la Raspberry pudiera ejecutar los Python para encender y apagar los led, pero no desde la web, le daba mil vueltas y por más que modificaba cosas, cuando desde PHP intentaba ejecutar un comando que necesitaba sudo no se ejecutaba...

Hasta que di con la solución, en *sudoers*, como expliqué arriba, le he dado permisos a www-data y cambiando los propietarios y grupos a cada archivo.

Por otro lado, un problema mucho más serio y que me ha desquiciado fueron los servomotores.

Lo primero, mis motores son de 180º únicamente, ya que son mucho más baratos que los de rotación continua, así que los “truqué” por decirlo de alguna manera. Hasta aquí bien, pero a la hora de programarlo no había por dónde cogerlo, por más que intentaba cosas no me hacía ni caso, desde que lo conectaba ya se activaba, no había manera e Internet no me ayudaba en absoluto, lo que a la gente le funcionaba a mí no, así que decidí hacerlo yo, intentar apañarlo de alguna manera, hasta que una de las veces, funcionó, después de ni se sabe el tiempo intentándolo