

SMART ALARM

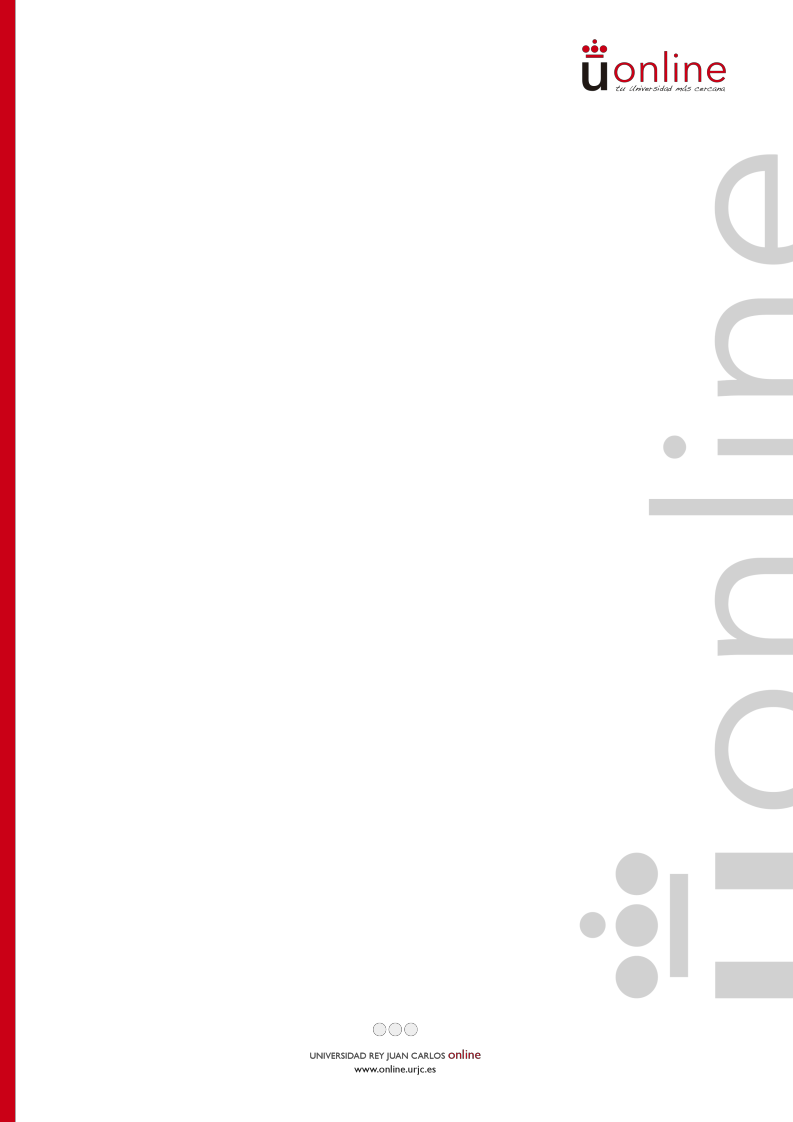
Sistemas empotrados y de tiempo real

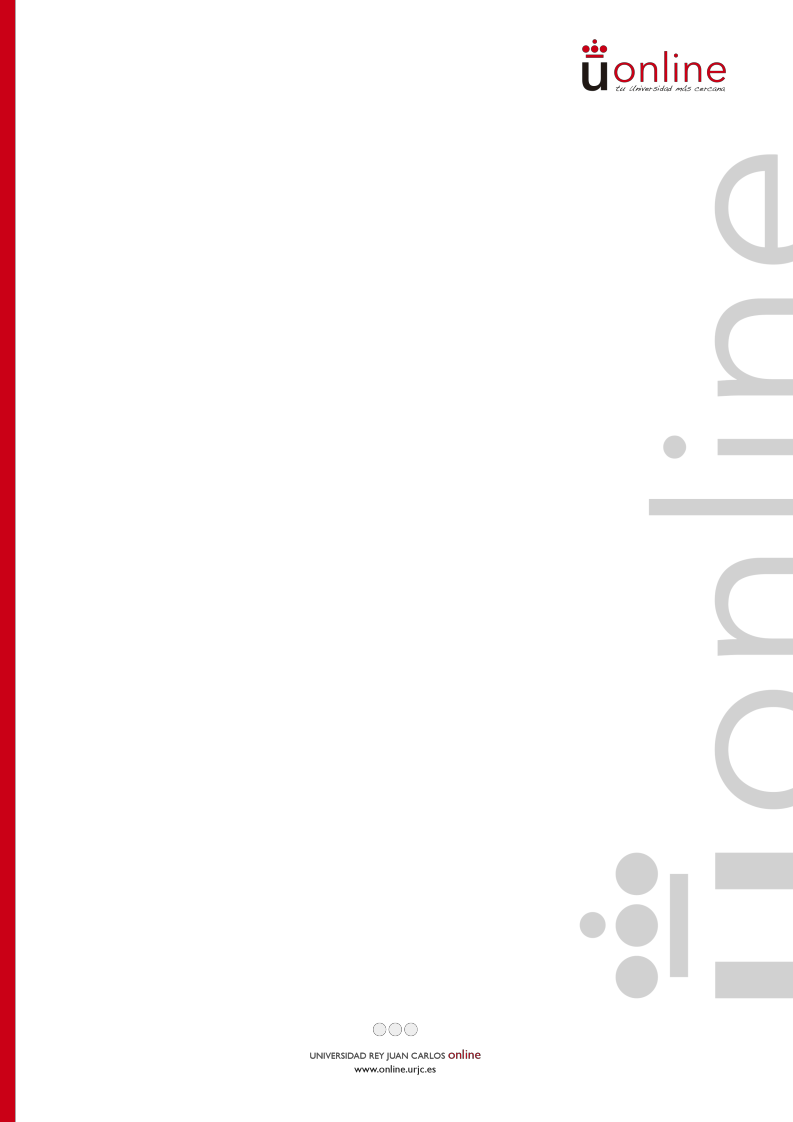
Víctor Álvarez Ramiro

Jesús Banegas Urda

Osvaldo Renato Luzuriaga L.

El objetivo de este proyecto será el diseño y desarrollo de un sistema de seguridad mediante el uso de un microcontrolador Arduino



Contenido

[Introducción 3](#_Toc479103333)

[Descripción del proyecto 4](#_Toc479103334)

[Características 4](#_Toc479103335)

[Lista de materiales 5](#_Toc479103336)

[7](#_Toc479103337)

[11](#_Toc479103338)

[12](#_Toc479103339)

# 

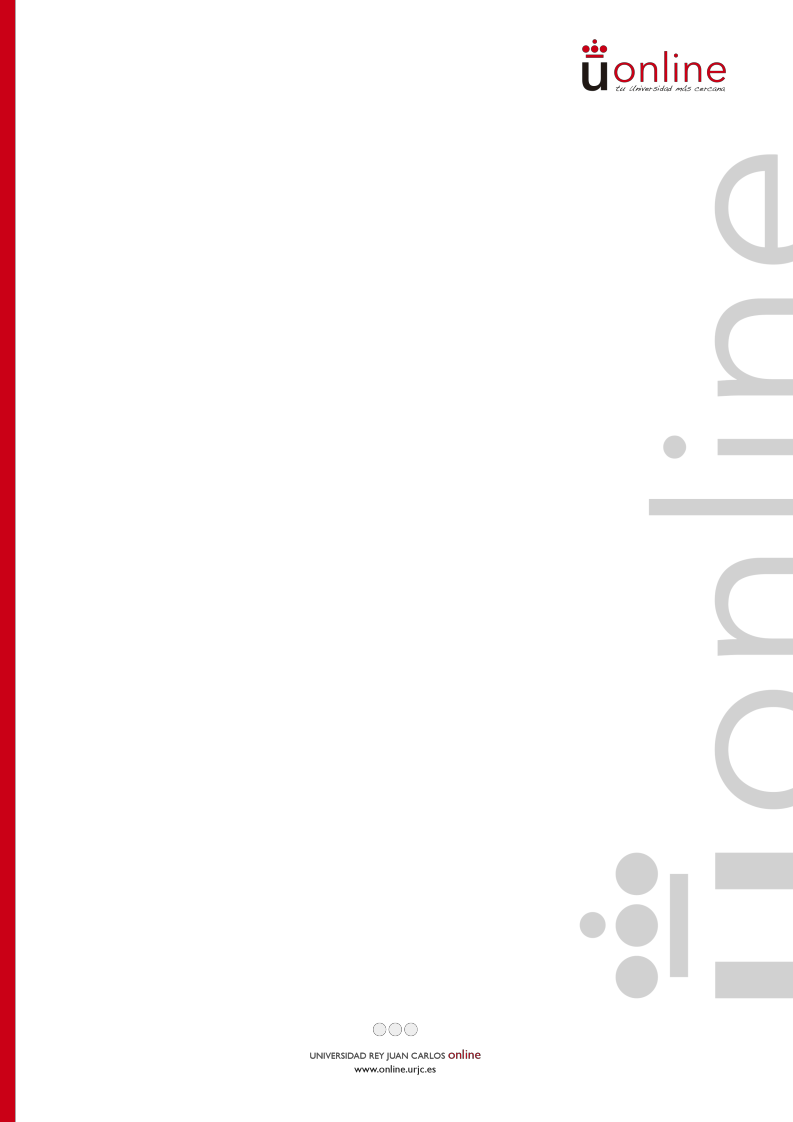
# Introducción

Este proyecto viene motivado por tratar de experimentar con un proyecto que trate de satisfacer una necesidad real, donde no solo se podrá ofrecer servicios de seguridad si no implementar distintos servicios enfocados al servicio del usuario

Trabajaremos con un microcontrolador Arduino y distintos componentes, el objetivo final es construir un sistema de seguridad donde el usuario pueda interactuar con el sistema y recibir notificaciones a su teléfono personal.

Los pasos marcados para lograr los objetivos serán:

* Buscar la tecnología que se empleará en el desarrollo.
* Realizar la especificación de los requisitos que debe cumplir el producto.
* Esquema electrónico del sistema de seguridad a desarrollar Implementar el sistema de alarma cubriendo los requisitos especificados.
* Verificar mediante un prototipo que se han cumplido los objetivos.
* Otro de los objetivos será realizar un sistema lo más potente posible mediante el uso de tecnologías libres y por tanto de bajo coste



# Descripción del proyecto

En este proyecto el objetivo claro es el de realizar un sistema de seguridad mediante

las siguientes funcionalidades.

Detección mediante un sensor de movimiento que estará controlado por el módulo

arduino este avisara de la intrusión mediante un buzzer, si esta intrusión no es aceptada

con el código de seguridad del propietario en el tiempo establecido “tiempo de retardo”

el sistema confirmara la detección como una intrusión real, por lo que informara de dicha intrusión

al propietario mediante un mensaje de texto a su teléfono, para esto el sistema estará conectado con un módulo GPRS/GSM.

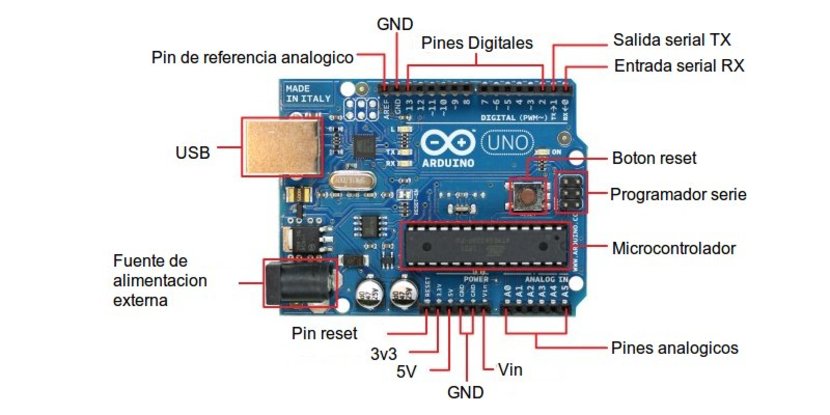
Otro de los objetivos es implementar un teclado matricial y un LCD para que el usuario propietario pueda interactuar son el sistema de seguridad, pudiendo este conectar, desconectar y visualizar los eventos que se han producido en su sistema de seguridad.

# Características

1. Conexión y desconexión del Sistema de seguridad mediante una contraseña para ello se utilizará un teclado matricial.
2. Visualización de eventos mediante un display LCD 1602B
3. Control de presencia mediante un sensor piro eléctrico.
4. Aviso acústico cuando detecte a un intruso a través de un buzzer
5. Aviso mediante un mensaje de texto a través de un módulo sim900 GPRS/GSM al teléfono del cliente.

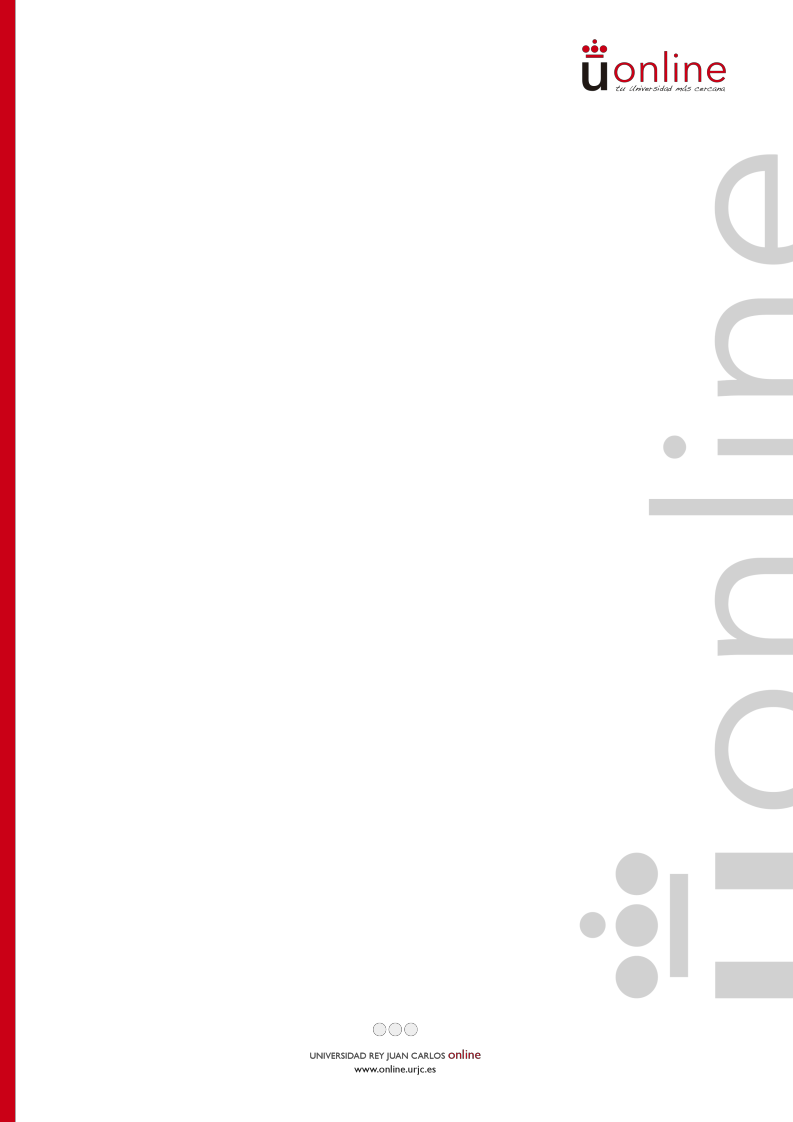
# Lista de materiales

1. Arduino UNO



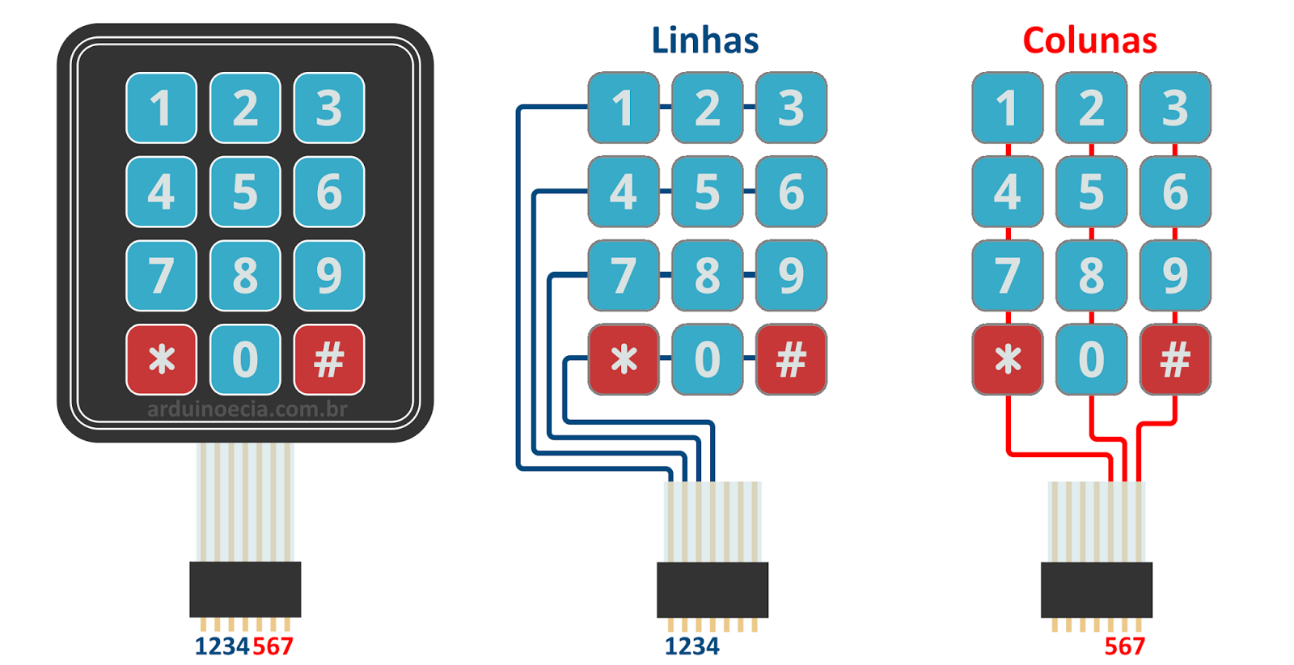
1. Modulo sim900 GPRS/GSM



1. Display LCD 1602B



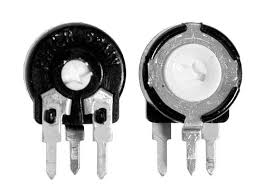
1. Teclado matricial (3x4)



1. Buzzer



1. Potenciometro 10kΩ



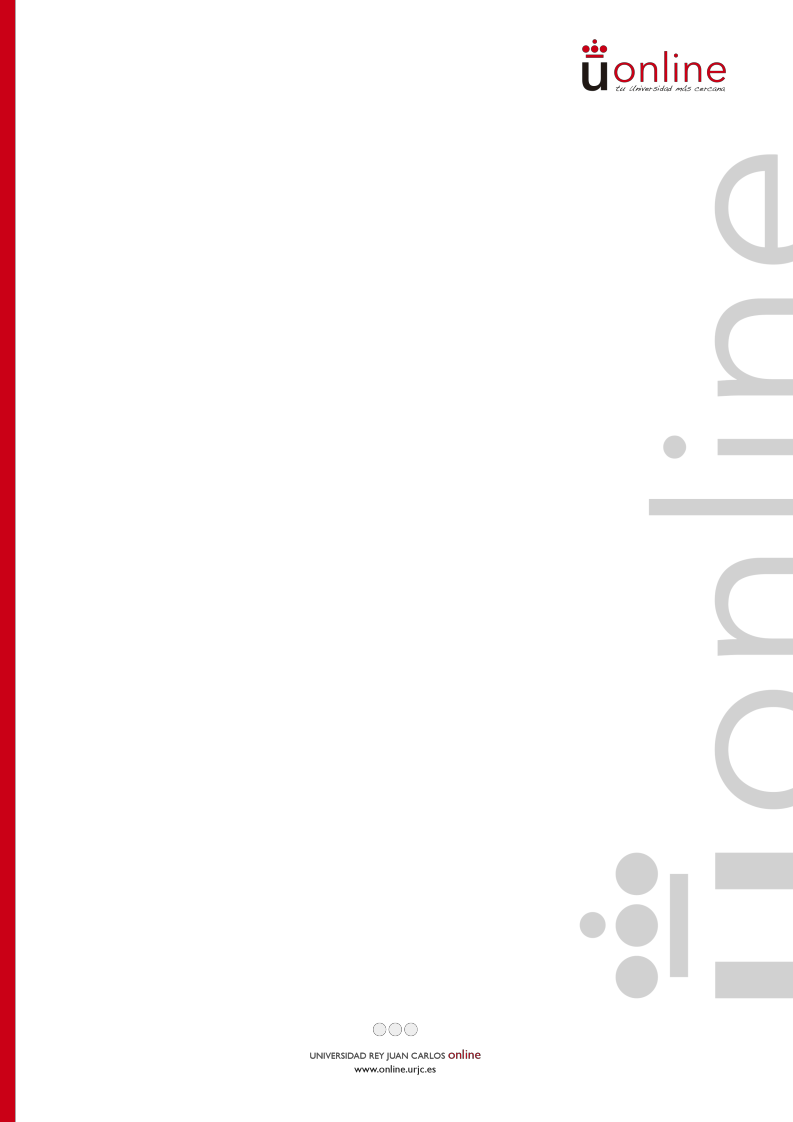
1. Fuente de alimentación



# 

# Esquema de conexión protoboard

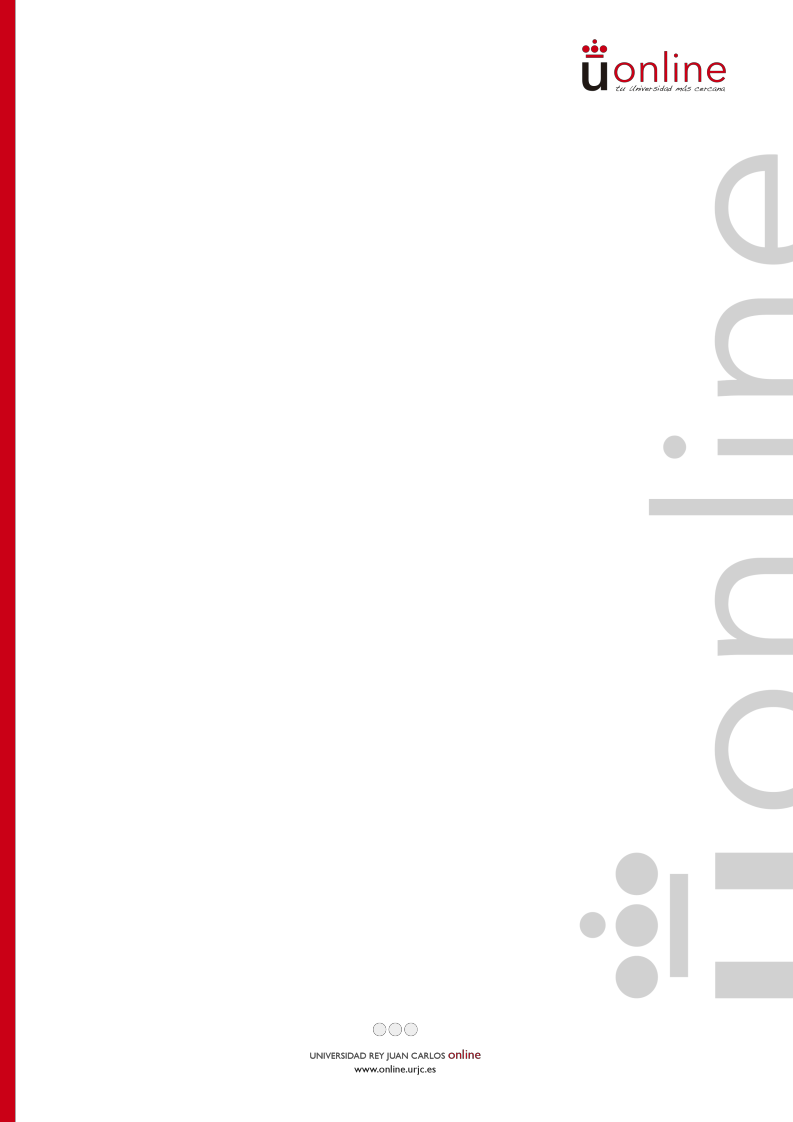
# 



# Plano esquemático

# 

# Código fuente



#include <LiquidCrystal.h>

#include <SoftwareSerial.h>

#include <Password.h> //Incluimos la libreria Password

#include <Keypad.h> //Incluimos la libreria Keypad

#include <EEPROM.h>

SoftwareSerial SIM900(7, 8); //Inicializamos el modulo GPRS

LiquidCrystal lcd(A0, A1, A2, A3, A4, A5); // Inicializamos el lcd

//Password password = Password( "1234" );

char\* password = "1234"; // Establecemos el password

char\* master = "123456";

int position = 0;

long tiempo;

byte sensorpir = 10;

byte buzzer = 13;

byte led = 9;

byte num\_click = 0;

byte estado\_alarma = 0;

byte intentos = 0;

char key;

boolean passOK = false;

boolean passWrong = false;

int tamanyo\_password = 4;

/\*\*\*\*Inicializamos el teclado\*\*\*\*\*/

const byte ROWS = 4; // Cuatro Filas

const byte COLS = 3; // Cuatro Columnas

char keys[ROWS][COLS] = {

{'1','2','3'},

{'4','5','6'},

{'7','8','9'},

{'\*','0','#'}

};

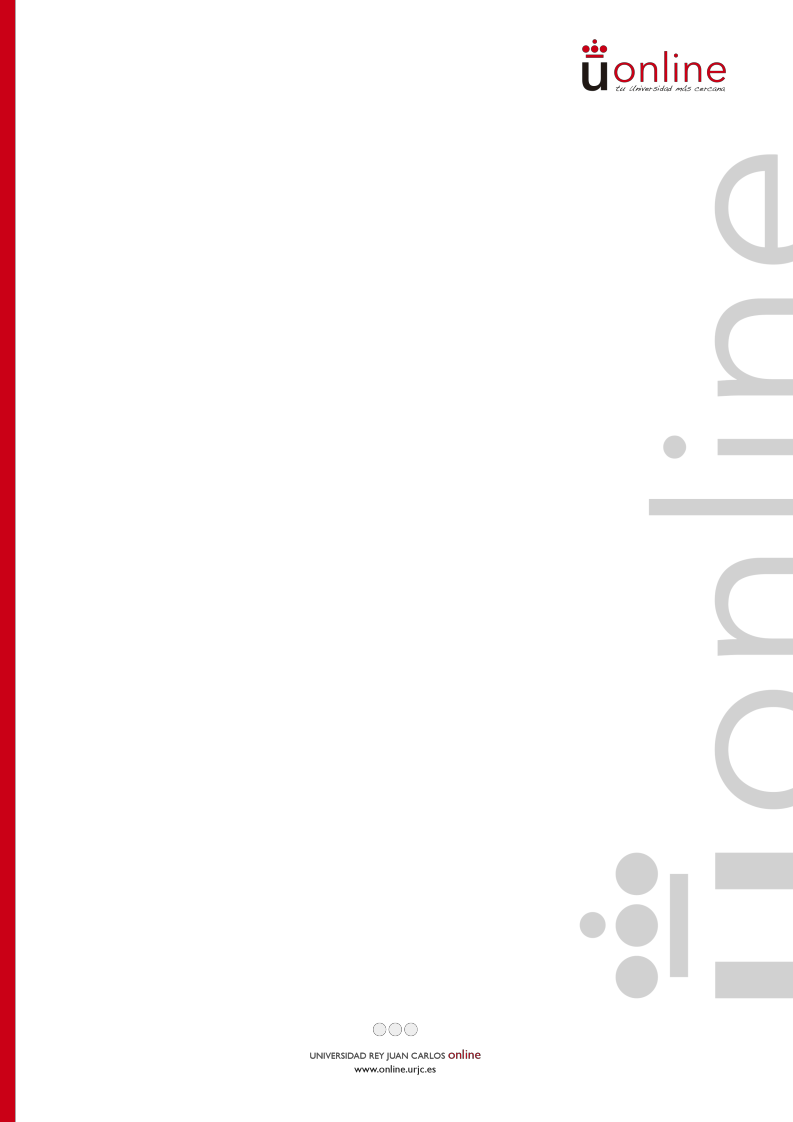
byte rowPins[ROWS] = { 12,11,6,5 };

byte colPins[COLS] = { 4,3,9};

// Creamos el Keypad

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

void setup(){

 lcd.begin(16, 2);

lcd.print("INICIALIZANDO..."); // Enviar el mensaje

pinMode(sensorpir,INPUT);

pinMode(buzzer, OUTPUT);

//pinMode(led, OUTPUT);

/\*--------------encendemos el módulo GPRS ---------------------------\*/

delay (5000); //Nos damos un tiempo para encender el GPRS y la alimentación de la tarjeta

SIM900.begin(19200); //Configura velocidad del puerto serie para el SIM900

Serial.begin(19200); //Configura velocidad del puerto serie del Arduino

Serial.println("OK");

delay (1000);

SIM900.println("AT + CPIN = \"1386\""); //Comando AT para introducir el PIN de la tarjeta

delay(25000); //Tiempo para que encuentre una RED

lcd.clear();

}

void loop(){

/\*

\* Estado 0 = alarma inactiva

\* Ponemos en nivel bajo las salidad de buzzer, led y mostramos mensaje en el LCD

\* Comprobamos si hay una tecla pulsada con la funcion eventoTeclado()

\* Verifica si la clave pulsada es correcta o incorrecta, actualiza las variables passOK = false

\* y estado\_alarma = 1 si el password es correcto, pasando al estado 1

\*/

while(estado\_alarma == 0)

{

digitalWrite(buzzer, LOW);

lcd.setCursor(0, 0);

lcd.print("Alarm off:");

lcd.setCursor(0, 1);

lcd.print("<<Enter key>>");

eventoTeclado();

if(passOK)

{

estado\_alarma = 1;

Serial.print("1");

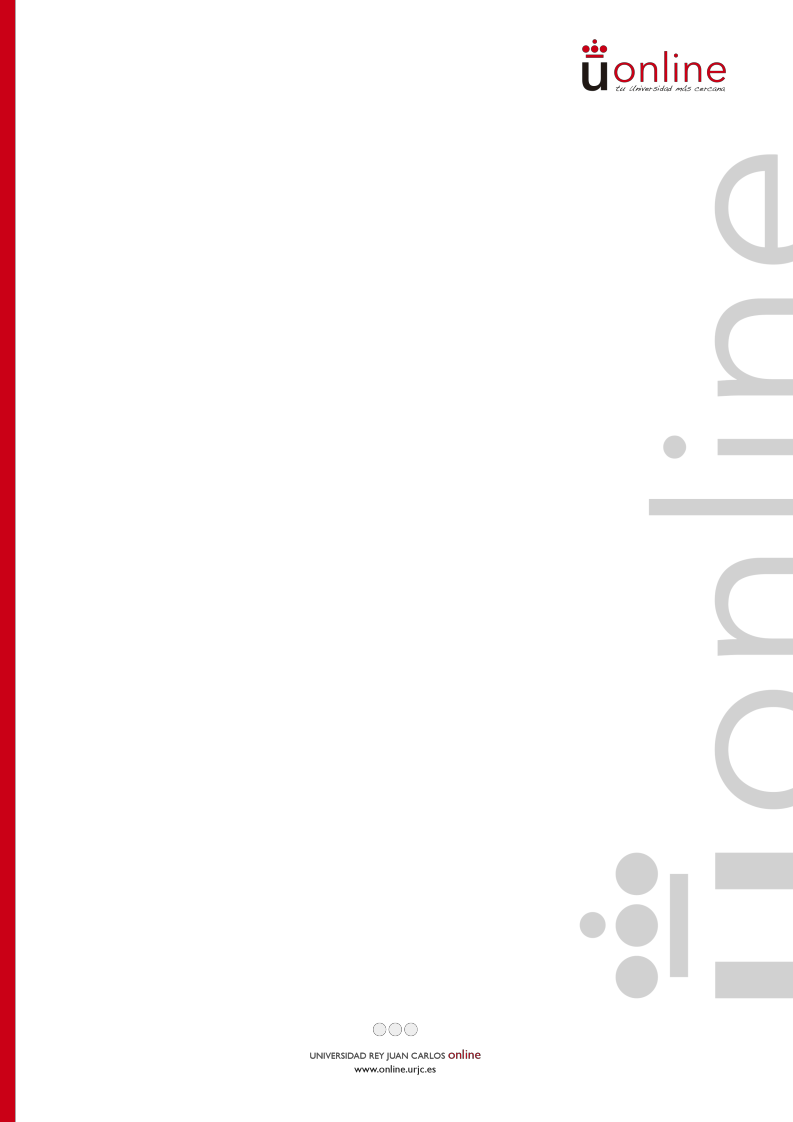
passOK = false;

}

else if(passWrong)

{

estado\_alarma = 5;

 Serial.print("5");

passWrong = false;

}

}

/\*

\* Estado 1 = alarma activada pero la salida del sensorpir esta en low

\* en este estado se estable en un retardo de 20 segundos para activar

\* la alarma donde se llama al metodo retardo(), se envia por el LCD

\* el mensaje de activacion en 20 segundo y por el buzzer se emite una señal

\* sonora, posteriormete nuestro sistema pasa al estado 2

\*

\*/

while(estado\_alarma == 1)

{

lcd.setCursor(0, 0);

lcd.print("Alarm ON, 20 sc ");

lcd.setCursor(0, 1);

lcd.print("to activate ");

retardo();

estado\_alarma = 2;

Serial.print("2");

}

/\*

\* Estado 2 = el sistema esta activo y comprueba si las salida del sensopir esta en HIGH

\* se indica por pantalla que la salida esta activa y es necesario pulsar la contraseña para desactivarla

\* se llegara a captar un intruso pasamos al estado 3

\*/

while(estado\_alarma == 2)

{

lcd.setCursor(0, 0);

lcd.print("System activated ");

lcd.setCursor(0, 1);

lcd.print("press key: OFF");

if(digitalRead (sensorpir) == HIGH)

{

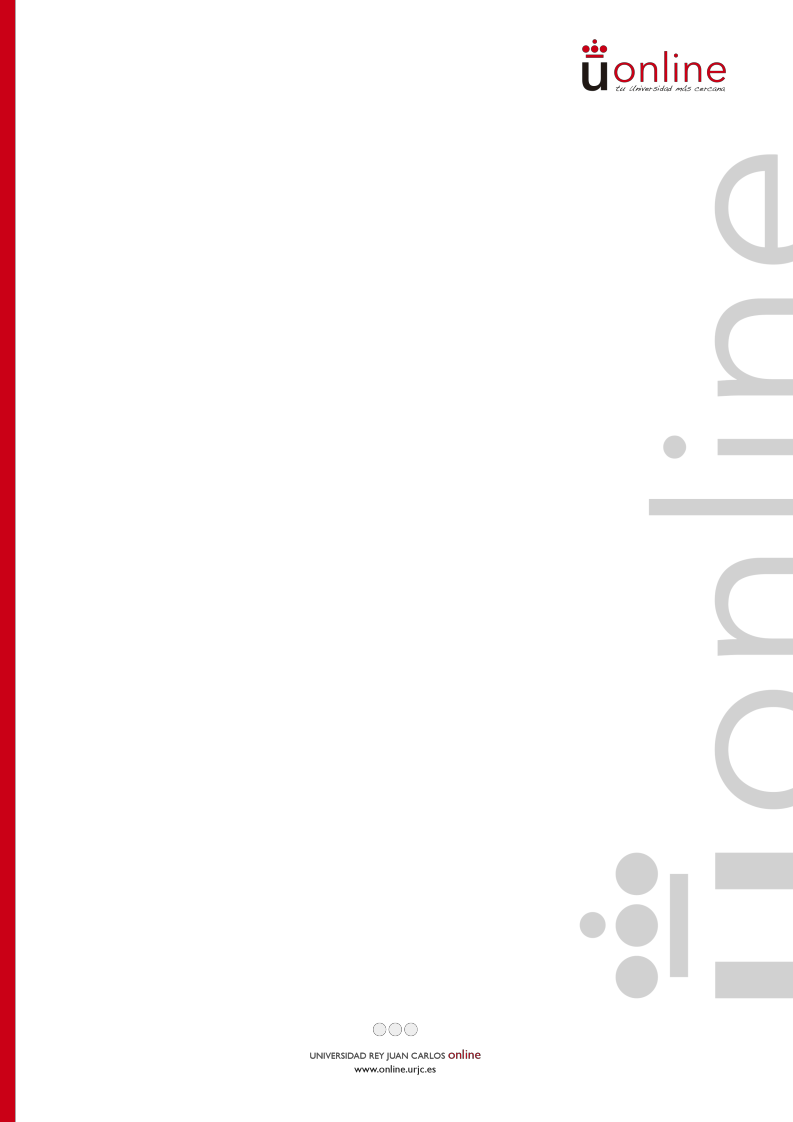
estado\_alarma = 3;

Serial.print("3");

delay(1000);

}

}

/\*

\* Estado 3= Se acaba de captar un intruso y se pondrá en funcionamiento el buzzer que indicara el tiempo de retardo que tiene

\* el usuario para desconectarla, se muestra un mensaje por pantalla indicando el tiempo de retardo que tiene para desconectarla

\* el sistema

\*/

while(estado\_alarma == 3)

{

tiempo = millis() + 20000;

do

{

//digitalWrite(leds, HIGH);

lcd.setCursor(0, 0);

lcd.print("Desactivate in ");

lcd.setCursor(0, 1);

lcd.print("20 seconds ");

//Activamos el retardo sonoro

digitalWrite(buzzer, HIGH);

delay(400);

digitalWrite(buzzer, LOW);

delay(100);

eventoTeclado();

//Si el password es correcto detenemos el bucle

if(passOK)

{

break;

}

}while(tiempo > millis());

if(passOK){

estado\_alarma= 0;

Serial.print("0");

passOK = false;

}else if(passWrong){

estado\_alarma = 6;

Serial.print("6");

passWrong = false;

}else if (estado\_alarma!= 0){

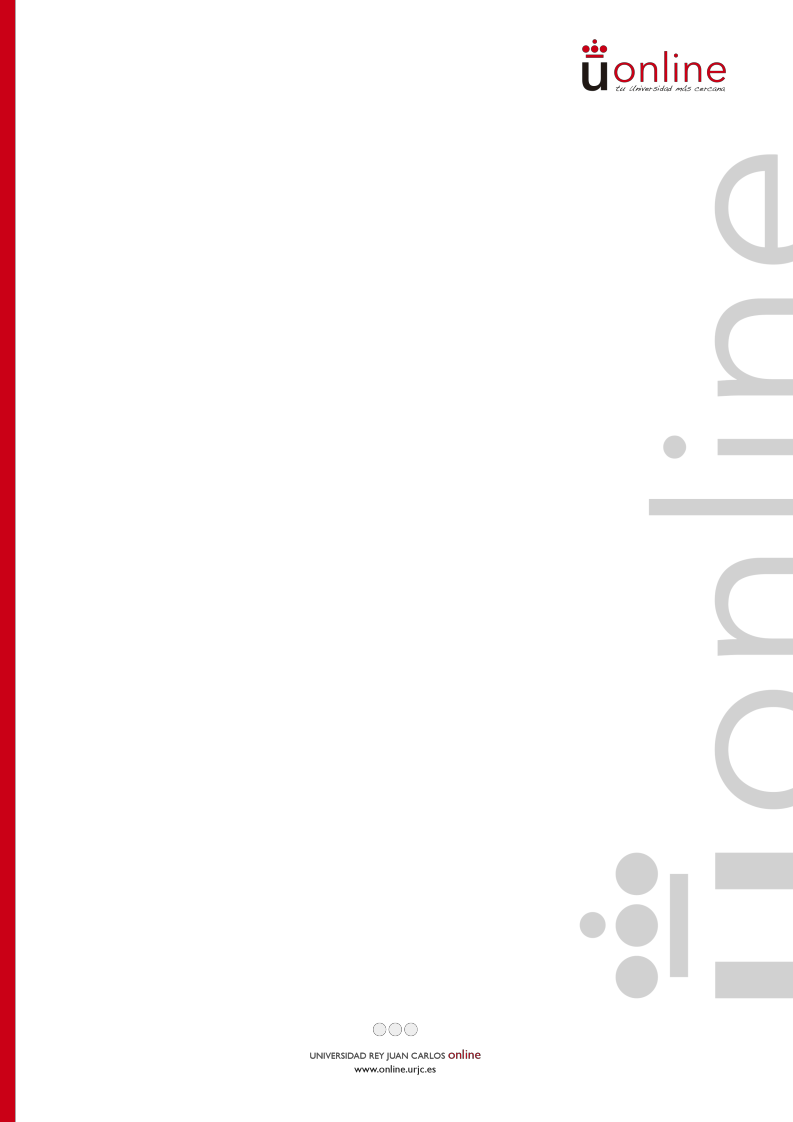
estado\_alarma = 4;

Serial.print("4");

}

}

/\*

 \* Estado 4 = El sistema esta activo, se confirma que hay un intruso en la instalacion del usuario para ello se utiliza el buzzer

\* para notificar la intrusion y se enviara un mensaje de texto a través de la funcion mensaje\_sms(), tambien indicaremos en el

\*/

while(estado\_alarma == 4){

//digitalWrite(leds, HIGH);

digitalWrite(buzzer, HIGH);

// digitalWrite(altavoz, LOW);

mensaje\_sms();

lcd.setCursor(0, 0);

lcd.print("<<ALERT>>");

lcd.setCursor(0, 1);

lcd.print(" INTRUDER ");

eventoTeclado();

//Si el password es correcto pasamos al estado 0 si es incorrecto al estado 6

if(passOK){

estado\_alarma = 0;

Serial.print("0");

passOK = false;

}else if(passWrong)

{

estado\_alarma = 6;

Serial.print("6");

passWrong = false;

}

}

/\*

\* Estado 5 = El password esta bloqueado, nos pide el codigo master para poder conectar el sistema nos ayudamos de la funcion claveMaster

\* que comprobara si el codigo es correcto

\*/

while(estado\_alarma == 5)

{

//digitalWrite(leds, LOW);

digitalWrite(buzzer, LOW);

//digitalWrite(altavoz, LOW);

lcd.setCursor(0, 0);

lcd.print("Password locked ");

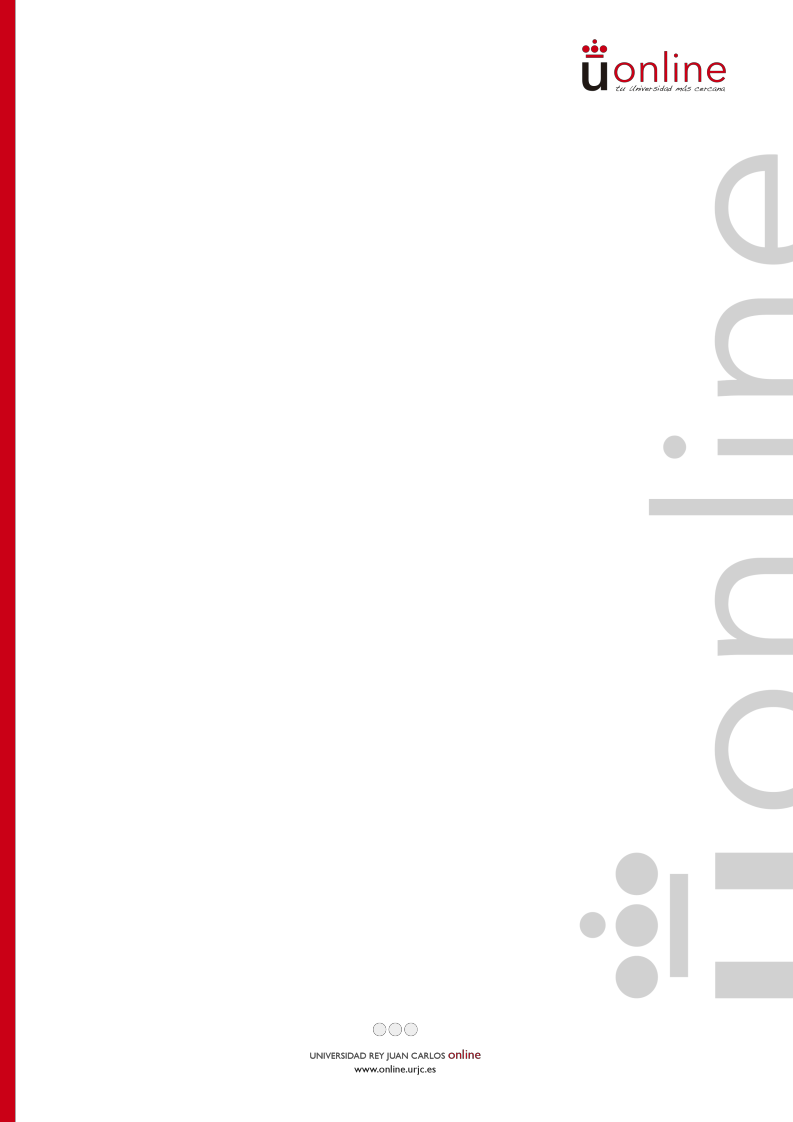
lcd.setCursor(0, 1);

lcd.print("Enter Master ");

claveMaster();

if(passOK)

{

 estado\_alarma = 1;

Serial.print("1");

passOK = false;

}

}

/\*

\* Estado 6 se utiliza para desconectar el sistema con el codigo master para desactivar el sistema

\* se indica un mensaje por el display pidiendo el codigo master para desconectar el sistema

\*/

while(estado\_alarma == 6)

{

//digitalWrite(leds, HIGH);

digitalWrite(buzzer, HIGH);

//digitalWrite(altavoz, LOW);

lcd.setCursor(0, 0);

lcd.print("ON, enter Master ");

lcd.setCursor(0, 1);

lcd.print("to desactivate ");

claveMaster();

if(passOK)

{

estado\_alarma = 0;

Serial.print("0");

passOK= false;

}

}

}

/\*

\* Emite un pitido de 20 segundo de retardo

\*/

void retardo(){

tiempo = millis() + 20000;

do

{

digitalWrite(buzzer, HIGH);

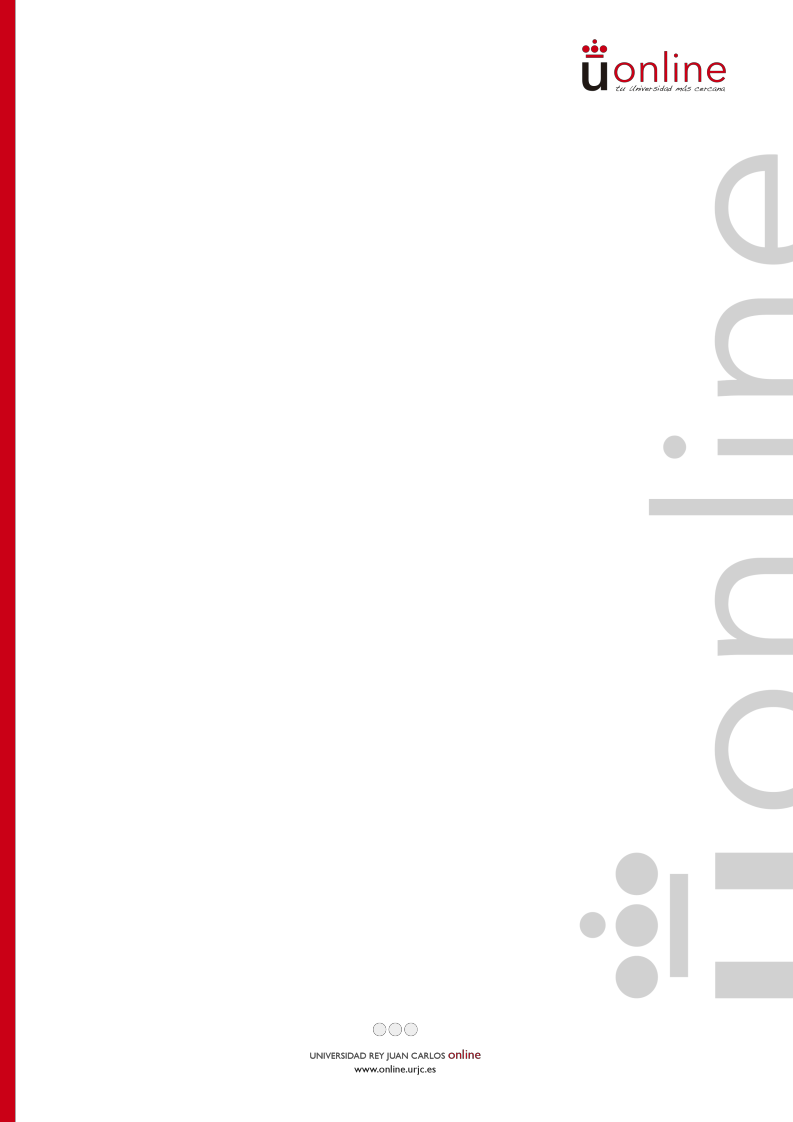
delay(400);

digitalWrite(buzzer, LOW);

delay(100);

}while(tiempo > millis());

}

/\*

\* EN CASO DE QUE QUERAMOS LLAMAR EN LUGAR DE ENVIAR SMS

void llamar()

{

Serial.println("Realizando llamada...");

SIM900.println("ATD606747597;"); //Comando AT para realizar una llamada

delay(30000); // Espera 30 segundos mientras realiza la llamada

SIM900.println("ATH"); // Cuelga la llamada

delay(1000);

Serial.println("Llamada finalizada");

}\*/

/\*

\* Envia un mensaje al modulo si se produce un salto de alarma

\*/

void mensaje\_sms()

{

Serial.println("Enviando SMS...");

SIM900.print("AT+CMGF=1\r"); //Configura el modo texto para enviar o recibir mensajes

delay(1000);

SIM900.println("AT+CMGS=\"699787684\""); //Numero al que vamos a enviar el mensaje

delay(1000);

SIM900.println("ALARMA HAY INTRUSOS EN SU HOGAR");// Texto del SMS

delay(100);

SIM900.println((char)26); //Comando de finalización ^Z

delay(100);

SIM900.println();

delay(5000); // Esperamos un tiempo para que envíe el SMS

Serial.println("SMS enviado");

}

/\*

\* Almacena un tecla pulsada, emite un sonido y la oculata por el caracter "\*"

\*/

void eventoTeclado(){

key = keypad.getKey();

if (key == '\*' || key == '#')

{

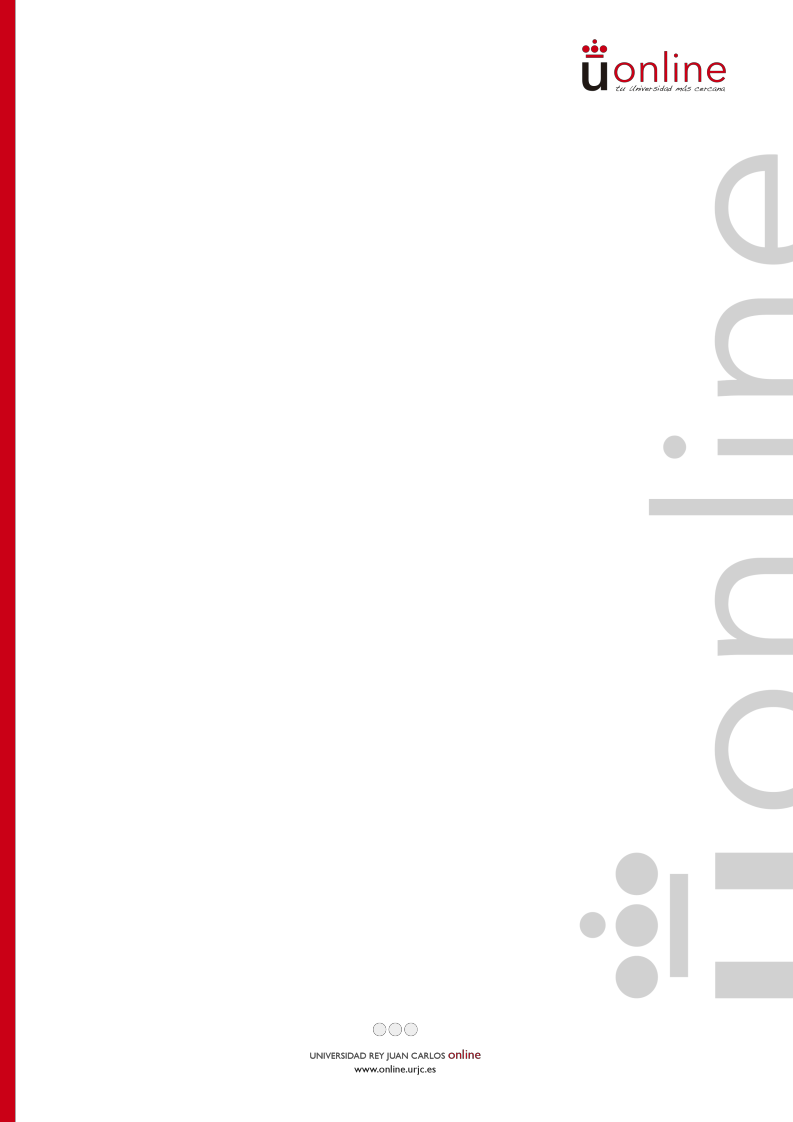
position = 0;

num\_click = 0;

Serial.print("reiniciado");

delay(700);

}

 // Si es correcta la contraseña se incrementa

if (key == password[position]){

position ++;

delay(200);

}

if (key >= '0')

{

num\_click ++;

delay(200);

}

if (num\_click == 4)

{

if (position == 4)

{

passwordCorrecto();

passOK = true;

num\_click = 0;

intentos = 0;

position = 0;

}else{

passwordIncorrecto();

num\_click = 0;

position = 0;

}

}

if (intentos == 3){

intentos = 0;

passWrong = true;

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("pass bloqueado ");

lcd.setCursor(0, 1);

lcd.print("Enter master code ");

//digitalWrite(altavoz, HIGH);

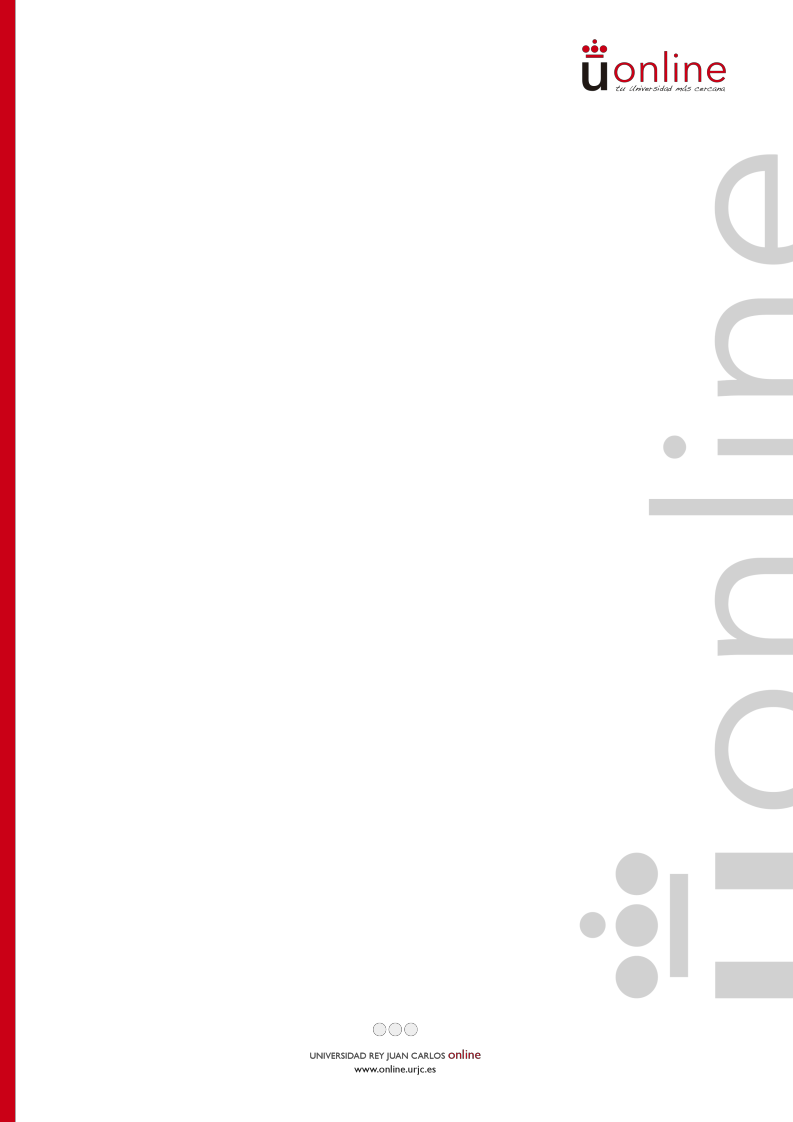
delay(2000);

}

}

void claveMaster()

{

key = keypad.getKey();

if (key == '\*' || key == '#')

{

position = 0;

num\_click = 0;

delay(100);

}

if (key == master[position])

{

position ++;

delay(100);

}

if (key >= '0')

{

num\_click ++;

delay(100);

}

if (num\_click == 6)

{

if (position == 6)

{

passOK = true;

num\_click = 0;

position = 0;

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Correct key ");

digitalWrite(buzzer, HIGH);

delay(500);

digitalWrite(buzzer, LOW);

delay(500);

}

else

{

num\_click = 0;

position = 0;

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Wrong key ");

digitalWrite(buzzer, HIGH);

delay(2000);

}

}

}

/\*

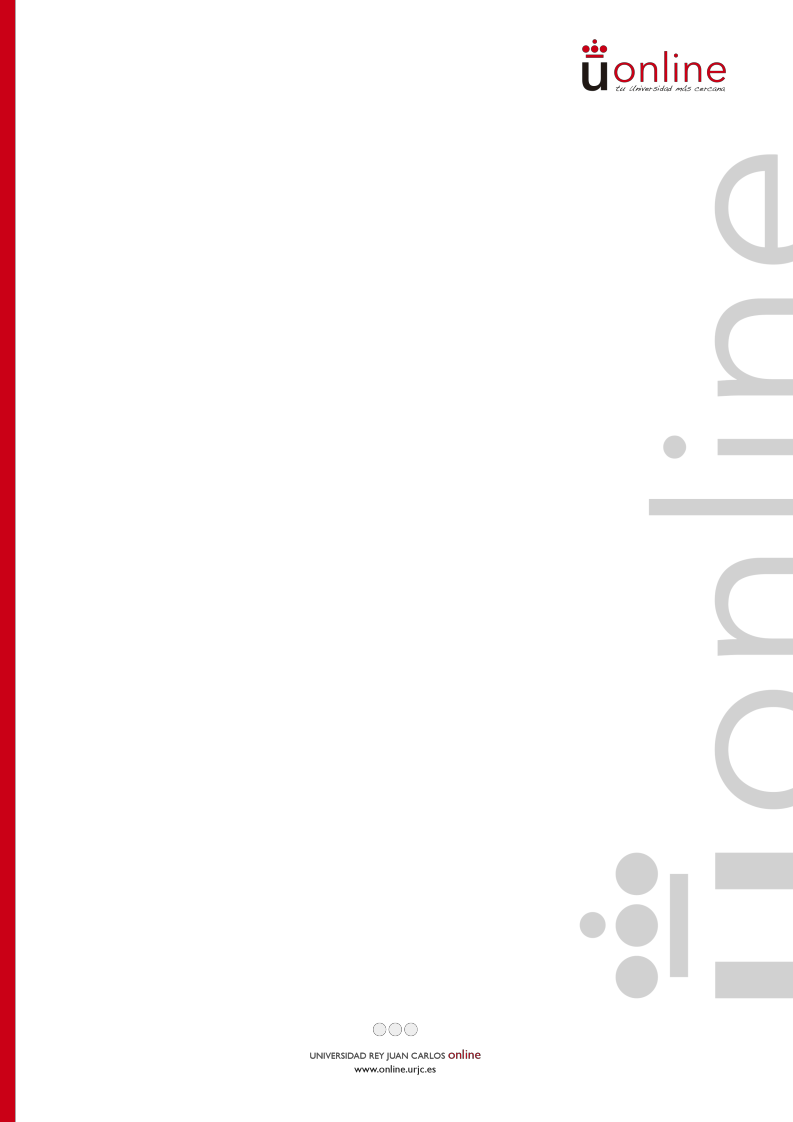
\* Emite una señal sonora indicando la conexion de la alarma e imprime un mensaje por el lcd

\*/

void passwordCorrecto(){

lcd.clear();

lcd.setCursor(0, 0);

 lcd.print("Correct key ");

delay(1000);

lcd.clear();

digitalWrite(buzzer, HIGH);

delay(120);

digitalWrite(buzzer, LOW);

delay(70);

}

/\*

\* Emite una señal sonora de desconexion e imprime un mensaje de confirmacion por el lcd

\*/

void passwordIncorrecto(){

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Error Password ");

delay(1000);

lcd.clear();

digitalWrite(buzzer, HIGH);

delay(300);

digitalWrite(buzzer, LOW);

delay(100);

}

# Presupuesto

# Placa Arduino uno 🡪 25 €

# Display LCD 1602 🡪 9.99 €

# Módulo GSM/GRPS 🡪 25.83 €

# Módulo HC-SR501 🡪 2.15 €

# Fuente alimentación Arduino 🡪 5.20 €

# Buzzer 5V 🡪 1 €

# Teclado Matricial 4x4 🡪 1.80 €

# Potenciómetro 🡪 3.15 €

## 

**8**

# Conclusiones

# La conclusión que podemos sacar de este proyecto, que consiste en un sistema de alarma, es que es un sistema de detección bastante fiable.

# Teníamos pensado en un principio añadirle una cámara, que cuando detectase el sensor pir un movimiento, la cámara se activara y realizara fotografías que posteriormente se almacenarían en la “nube”, pero debido al alto coste de la cámara lo descartamos.

