PROYECTO INFORMATICA.

Objetivo: (con contraseña)

Con la aparición del COVID las vacunas han jugado un roll fundamental para el avance y el progreso de la pandemia, sin embargo, estas requieren unas condiciones para su preservación muy específicas. Nuestro objetivo, es crear un dispositivo que capte la esencia de dichos dispositivos, pero adaptado a un uso mucho mas cotidiano. Al tratarse de productos con cierto valor, es de vital importancia el uso de una contraseña para acceder a ellos y para que no haya ningún tipo de problemas.

En resumen, nuestro objetivo sería: crear un sistema de refrigeración inteligente destinado a productos delicados que requieran una refrigeración constante y ser protegidos, que además de almacenar objetos, nos permita entre otras cosas ser capaces de controlar su temperatura, mandar avisos programados o hasta cerrar su puerta automáticamente si por algún despiste se queda abierta.

Con este proyecto pretendemos aprender sobre Arduino y la electrónica en general, implementar los conceptos aprendidos en clase además de crear algo que sea de utilidad.

Menú de opciones y medidas:

Primero empezaremos con la temperatura. Mediante el sensor implantado, podremos ver la temperatura actual, cambiar la temperatura en función de lo que se desee almacenar, así como sugerencias de la temperatura recomendada y avisos si esta baja por debajo de alguna mínima establecida, saltando dicho aviso al móvil.

Como función básica de cualquier nevera implantaremos unas luces led que nos permitan ver en su interior cuando se abra (y se apaguen cuando se cierren) a través de un sensor magnético que además contara el tiempo que está abierta para después de un tiempo prudencial se cierre automáticamente.

Dentro de las funciones de la alarma, de la que se ocuparan de avisar al usuario tanto un buzz como un aviso por la aplicación móvil, se encuentran las anteriormente mencionadas como avisos de regulación de la temperatura, del fallo de algún componente o cuando el tiempo de apertura a pasado un valor establecido.

Finalmente llegamos a los registros de la nevera que nos mostrara la temperatura media de la nevera, el número de veces que ha sido abierta, la hora y fecha y por cuanto tiempo, la variación de temperatura desde que se abre hasta que se cierra, así como la contraseña elegida y el numero de errores cometidos cada vez que se intenta abrir

Finalmente, una opción que nos permita salir del programa.

Esquema del menú principal

#include<stdio.h>

#include<stdlib.h>

int menu\_ppal(int);

int menu\_ppal(int opcion) {

int opcion;

do {

printf("MiNevera");

printf("-------------------");

printf("Elija una opcion");

printf("1-Estado de los componentes");

printf("2-Temperatura");

printf("3-Alarmas");

printf("4-Control de los sensores");

printf("5-Registros");

printf("6-Salir");

scanf\_s("%d", &opcion);

} while ((opcion != 1) && (opcion != 2) && (opcion != 3) && (opcion != 4) && (opcion != 5));

return opcion;

}

COMPONENTES A UTILIZAR

* Placa base : [Arduino MEGA](https://www.banggood.com/es/Mega2560-R3-ATmega2560-16AU-Control-Board-With-USB-Cable-For-Arduino-p-73020.html?p=HT2402839656201410KV): Este es el controlador principal, se ha elegido por el Arduino MEGA por el numero de E/S
* Sensor de temperatura:sensor LM35
* Luces led (compatible con arduino)
* Buzzer( compatible con arduino)
* Sensor de apertura y cierre: <https://es.rs-online.com/web/p/detectores-de-apertura-de-puertas-y-ventanas/1220719>

Componentes:

* Nuestra nevera se compondrá de una placa base Arduino uno, sensores de temperatura y humedad(DHT22), sistema de refrigeración(tipo), fuente de alimentación(por determinar), luces led , servo motores(tipo) y un display además de una aplicación móvil adicional con la que podremos monitorizar y controlar nuestro dispositivo

En cuanto a la compatibilidad de los componentes, al tratarse de piezas utilizadas en los circuitos expuestos abajo y dando por hecho que trabajaremos en Arduino, suponemos que los componentes serán compatibles entre sí.

Esquema del montaje en Arduino:

Interfaz de usuario gráfica, Diagrama

Descripción generada automáticamente

Programa básico para luces led:

(Para este programa se utiliza un interruptor en vez de un sensor magnético)

Texto

Descripción generada automáticamente con confianza media

En nuestro código, la puerta jugará un papel fundamental ya que será esta quien mantenga el contenido de su interior seguro, su funcionamiento será el siguiente:

Tras poner la contraseña correcta, el programa que se encarga de abrir las puertas (será algo parecido a lo de abajo), se pondrá en marcha y finalizará cuando este se haya abierto por completo(o haya llegado al número de grados asignados), para cerrarse, dos componentes entrarán en marcha, el primero, será un sensor de ultrasonidos ya que será este quien detecte si hay algún tipo de presencia para así no cerrarse, si pasado un tiempo( unos 5 seg) no detecta ninguna presencia, la puerta comenzará a cerrarse hasta quedar completamente cerrada(0 grados) poniéndose de nuevo en marcha la contraseña.

Como detalle adicional, contará con dos luces led en la puerta que indicaran cuando este esté abierto(verde) y cuando esté cerrado(rojo)

El siguiente programa (que al tratarse de una versión temprana no cuenta con los componentes de la contraseña ni del ultrasonido) es de una puerta que se abre de manera automática de 10 en 10 grados y que pasado un tiempo se vuelve a cerrar.

Texto

Descripción generada automáticamente con confianza media

Texto, Word

Descripción generada automáticamente con confianza media

Finalmente, haremos un breve ejemplo de un programa que recoge la temperatura leída y la muestra en pantalla.

Este sensor será capaz de detectar la temperatura y humedad para que los productos que se encuentren en su interior no se deterioren.

![Imagen que contiene electrónica, circuito

Descripción generada automáticamente]()

Así quedaría diseñado finalmente el programa para Arduino del sensor encargado de medir la temperatura y la humedad dentro del aparato, en este caso hemos decidido programarlo para el sensor DHT22, aunque también es perfectamente compatible para el sensor DHT11, solo habría que sustituir el nombre de uno por el otro dentro de nuestro programa de Arduino, en correspondencia con el que se haya escogido para realizar el trabajo.

Texto

Descripción generada automáticamente