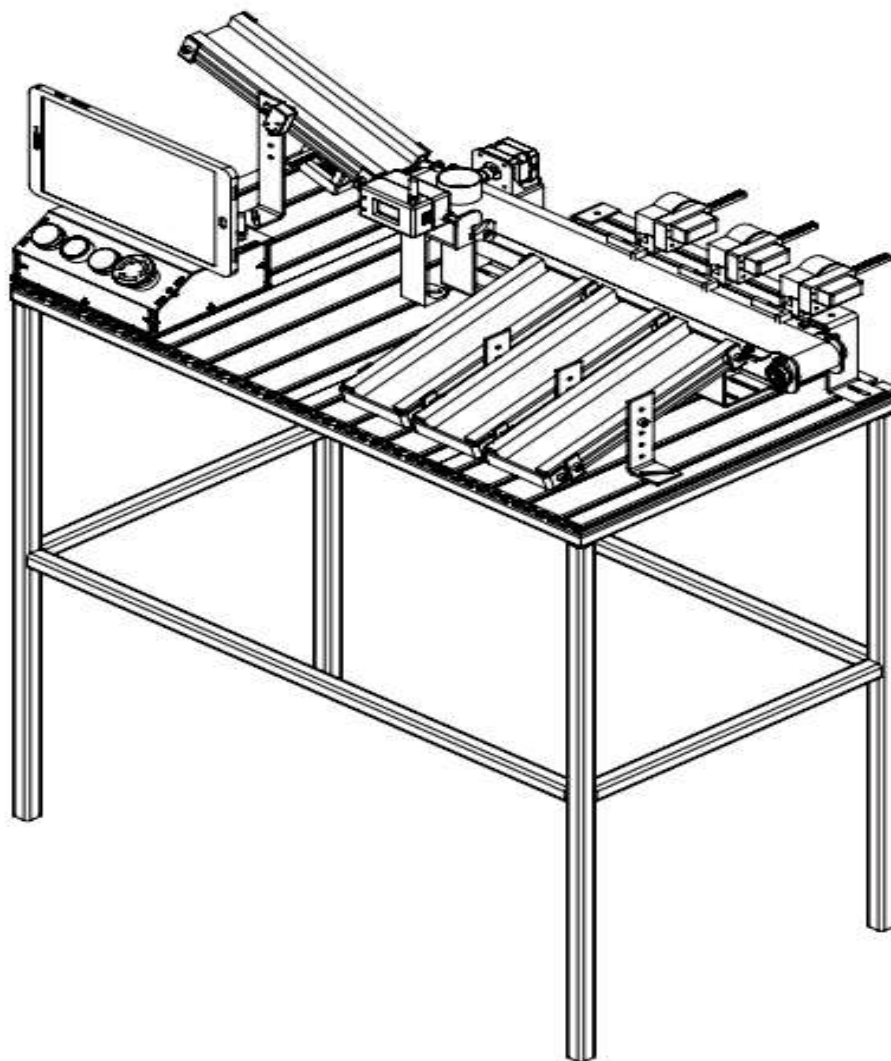


Módulos didácticos con redes de sensores inalámbricos aplicados dentro de un proceso secuencial



Manual de Usuario

Contenido

Introducción	3
Especificaciones técnicas del sistema	4
Partes del Sistema.....	5
Botones de acceso al usuario	6
Módulos inalámbricos	7
Configuración del módulo maestro	17
Conectar el módulo maestro a internet.....	19
Visualización de datos en Firebase	19
Instrucciones para el uso de la aplicación móvil	21
Puesta en marcha.....	24
Instrucciones para cargar los módulos	27
Plan de mantenimiento	29
Instalar Flutter en un pc con Windows	30
Configuración de Firebase con Flutter	31

Introducción

El presente proyecto consiste en conseguir una forma novedosa de practicar sobre la tecnología inalámbrica que se ve en la cátedra de Introducción a sistemas ciberfísicos de forma rápida, visual y con variedad de sensores donde el proyecto se basa en diseñar y construir módulos de sensores individuales que sean inalámbricos, con alimentación propia y conectados a una red mediante un protocolo específico y controlados por un solo módulo maestro, donde los datos comunicados se puedan visualizar desde diferentes dispositivos por internet (IoT). Estos módulos individuales deberán estar acoplados a la estructura del proceso secuencial, con las señales procedentes de los sensores acondicionadas y disponibles en una interfaz didáctica donde se pueda interactuar con estos para ajustar los parámetros de las entradas y salidas según el requerimiento del usuario.

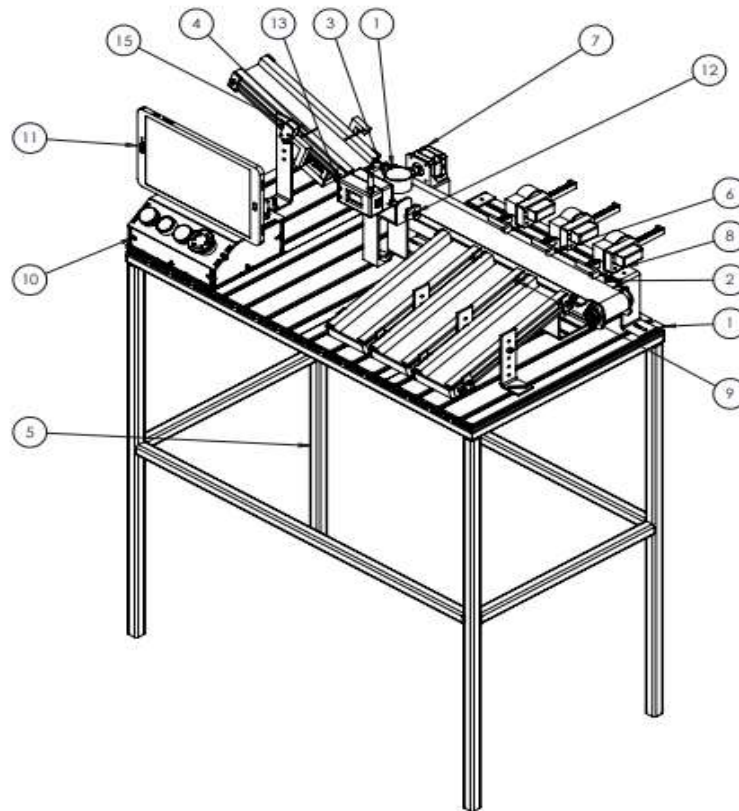
Mediante la interfaz se busca poder controlar el proceso secuencial según el objetivo que el usuario requiere, desde solo monitorear una señal de un sensor específico, hasta poder realizar un proceso secuencial completo ambientado a la industria 4.0, con el fin de poder realizar clasificación de objetos ya sea por su material, color, altura o a su vez contar estos objetos, donde según el proceso se puede colocar el módulo del sensor respectivo y mediante la interfaz programarlo para poder realizar una comunicación eficiente.

En el presente documento se encontrará la información necesaria para instalar librerías, configuración para los módulos además de recomendaciones para la puesta en marcha de la estación de clasificación, lo que se busca a través de este manual es enseñar al usuario el correcto funcionamiento y mantenimiento del sistema.

Especificaciones técnicas del sistema

Parámetro	Especificación
Motor de la banda	Motor nema 17D/Paso 34MM
Actuador lineal	Servomotor MG-995
Dimensión de la mesa	700mmx400mmx780mm
Protocolo de comunicación	Lora
Controlador	Heltec Lora 32 V3
Material de los soportes y la mesa ranurada	Aluminio 6061-T4
Material de la estructura de los módulos	PLA
Fuente de alimentación	Altek 750w Atx
Numero de módulos inalámbricos	4
Sensor para el módulo 1	Sensor RGB TCS34725
Sensor para el módulo 2	Sensor infrarrojo HW201
	Sensor capacitivo LJC30A3-H-Z/BX
Sensor para el módulo 3	Sensor Inductivo NPN LJ18A3- 8Z/BX
Sensor para el módulo 4	DHT22
Batería del módulo inalámbrico	ANERA LiPo 1000mAh
Numero de objetos de clasificación	8
Tablet	Alcatel 9032T

Partes del Sistema



N.º DE ELEMENTO	N.º DE PIEZA	CANTIDAD
1	Mesa ranurada	1
2	Banda transportadora	1
3	pieza	1
4	Dispensador de piezas	1
5	Estructura mesa	1
6	Actuador lineal	3
7	Actuador banda	1
8	Base actuador lineal	1
9	Rampa	3
10	Tablero control	1
11	Tablet	1
12	sujetador_soporte	1
13	Modulo inalámbrico	1
14	fuelle de poder	1
15	Sensor dispensador	1

Botones de acceso al usuario

Tablero de control

- **Paro de Emergencia:**



Botón de enclavamiento corta el movimiento del dispensador de objetos, la banda y los actuadores.

- **Pulsador de inicio**



Pulsador que permite el arranque del sistema de clasificación accionando el dispensador para liberar los objetos y encendiendo la banda.

- **Piloto luz verde**



Indica que el sistema arranco con normalidad lo que significa que la etapa de clasificación ha comenzado.

- **Piloto luz rojo**



Indica que el paro de emergencia fue pulsado debido a un fallo o avería en el sistema de clasificación.

Módulos inalámbricos

Módulo sensor de colores

El módulo sensor de colores tiene el objetivo de identificar los objetos de color amarillo rojo y verde.

Características

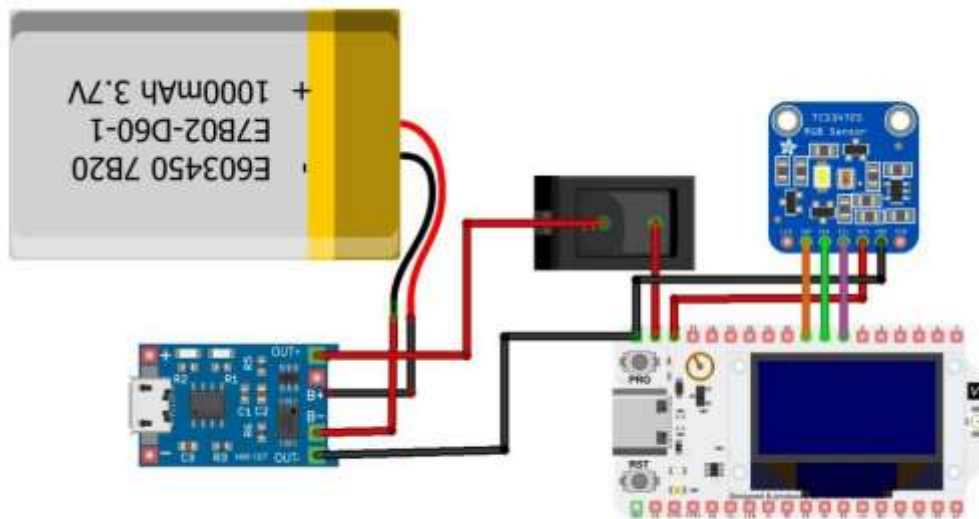


- Sensor de color tcs34725
- Voltaje de alimentación: 3 – 5 VDC
- Corriente nominal de operación: 60 a 100 mA
- Duración de la batería: 4h 30min
- Cargador: tipo micro USB V8 de 5v
- Programación: cable tipo C mediante IDE Arduino

Partes del módulo



Circuito de conexión interna



Programación del módulo

Para la comunicación por radiofrecuencia con el protocolo LoRa desde las tarjetas Heltec Lora V3 es necesario la utilización de la librería "LoRaWan_App.h" ya que el fabricante recomienda la librería "Heltec.h" pero es incompatible hasta el momento con la versión del chip LoRa SX1262, para el empleo del sensor de color TCS34725 es imprescindible la librería "Adafruit_TCS34725.h" e inicializarlo.

```
#include "LoRaWan_APP.h"
#include "Arduino.h"
#include "Adafruit_TCS34725.h"
Adafruit_TCS34725 tcs = Adafruit_TCS34725(TCS34725_INTEGRATIONTIME_50MS, TCS34725_GAIN_4X);
const char TX_Value[] = {};
```

Para configurar de manera adecuada la red de sensores LoRa, aunque en el país no esté regulado una frecuencia específica, el fabricante recomienda que al igual que en Estados Unidos y otros países de Sudamérica, se utilice el ancho de frecuencia 902-928 MHz, donde seleccionamos el de 920 MHz.


```

#define RF_FREQUENCY 920000000

#define TX_OUTPUT_POWER 14

#define LORA_BANDWIDTH 0

#define LORA_SPREADING_FACTOR 7
#define LORA_CODINGRATE 1

#define LORA_PREAMBLE_LENGTH 8
#define LORA_SYMBOL_TIMEOUT 0
#define LORA_FIX_LENGTH_PAYLOAD_ON false
#define LORA_IQ_INVERSION_ON false

#define RX_TIMEOUT_VALUE 500
#define BUFFER_SIZE 30

char txpacket[BUFFER_SIZE];
char rxpacket[BUFFER_SIZE];

```

Para el envío de datos, como ya se ha mencionado se ha seleccionado los tres colores RGB, donde se envía la palabra “Rojo” cuando el objeto percibido es de color rojo, de la misma forma con las palabras “Verde” y “Azul”.

```

void loop() {
    float red,green,blue;
    tcs.getRGB(&red,&green,&blue);
    int R= int(red);
    int G=int(green);
    int B=int(blue);
    String color = "";
    if((R-G>40)&(R-B>40)){
        color= "Rojo";
        Serial.print("Sent ");
        const char *TX_Value="Rojo";
        Serial.println(TX_Value);
        Radio.Send((uint8_t *)TX_Value, strlen(TX_Value));
        delay(1000);
    }
    if((G-B>20)&(G-R>30)){
        color= "Verde";
        Serial.print("Sent ");
        const char *TX_Value="Verde";
        Serial.println(TX_Value);
        Radio.Send((uint8_t *)TX_Value, strlen(TX_Value));
        delay(1200);
    }
    if((B-G>10)&(B-R>30)){
        color= "Azul";
        Serial.print("Sent ");
        const char *TX_Value="Azul";
        Serial.println(TX_Value);
        Radio.Send((uint8_t *)TX_Value, strlen(TX_Value));
        delay(1000);
    }
}

```

Módulo sensor de tamaño

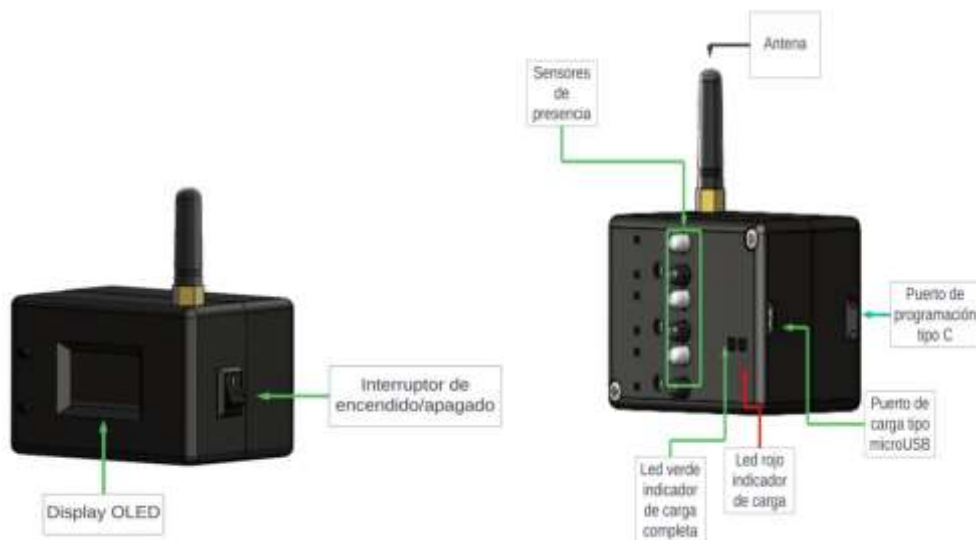
El módulo sensor de tamaño tiene el objetivo de identificar los objetos de tamaño como pequeño, medio y grande.

Características

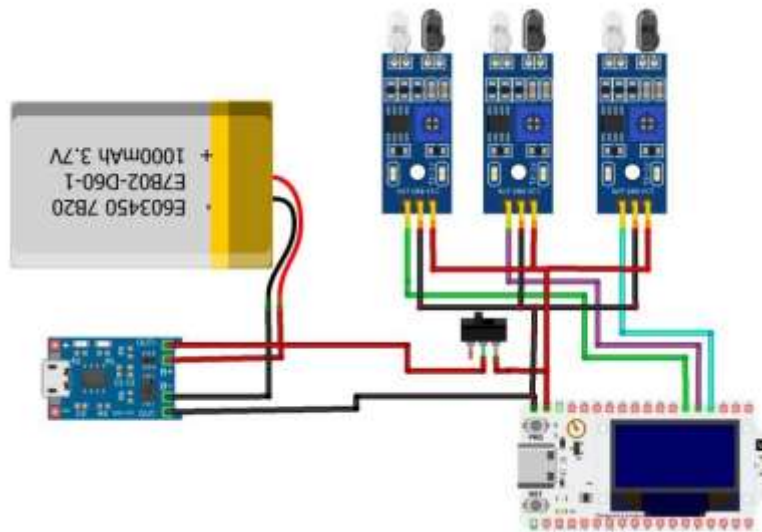


- 3 sensores de presencia FC-51
- Voltaje de alimentación: 3.3V – 5V
- Distancia de detección: 20 mm – 300 mm (ajustable)
- Angulo de detección: 35°
- Duración de la batería: 4h 30min
- Cargador: tipo micro USB V8 de 5v
- Programación: cable tipo C mediante IDE Arduino

Partes del módulo



Circuito de conexión interna



Programación del módulo

De la misma forma que el anterior, se utilizan las mismas librerías para la comunicación pero sin la necesidad de utilizar una librería específica para el sensor, ya que este sensor envía una señal en bajo cuando detecta la presencia de un objeto, para lo cual para el sensor de tamaño es necesario colocar condiciones para cuando solo detecte el sensor de abajo, envíe por radiofrecuencia la palabra “pequeño”, cuando detecte el sensor de abajo y el de la mitad, la palabra “mediano” y si detectan los tres sensores, la palabra “grande”.

```
if (hayObstaculoA == LOW && hayObstaculoB == LOW && hayObstaculoC == LOW ) {
    size= "grande";
    Serial.print("Sent ");
    const char *TX_Value="grande";
    Serial.println(TX_Value);
    Radio.Send((uint8_t *)TX_Value, strlen(TX_Value));
    delay(1000);
}
if (hayObstaculoA == HIGH && hayObstaculoB == LOW && hayObstaculoC == LOW) {
    size= "mediano";
    Serial.print("Sent ");
    const char *TX_Value="mediano";
    Serial.println(TX_Value);
    Radio.Send((uint8_t *)TX_Value, strlen(TX_Value));
    delay(500);
}
if (hayObstaculoA == HIGH && hayObstaculoB == HIGH && hayObstaculoC == LOW) {
    size= "pequeño";
    Serial.print("Sent ");
    const char *TX_Value="pequeño";
    Serial.println(TX_Value);
    Radio.Send((uint8_t *)TX_Value, strlen(TX_Value));
}
```

Al igual que los anteriores, se necesita que envíe palabras cuando el objeto es de algún metal ferroso o no, para ello se utiliza la condición siguiente para cuando el sensor inductivo detecta la presencia de un objeto metálico ferroso envíe la palabra “si”, donde al contrario cuando el sensor capacitivo detecte la presencia de un objeto, pero el sensor inductivo no detecte la presencia de un objeto metálico ferroso, envíe la palabra “no”.

```
bool state = digitalRead(sensorPin);  
bool stateI = digitalRead(sensorPinI);  
  
if (state == 0)  
{  
    count+=1;  
  
    if(count == 1){  
        digitalWrite(2,HIGH);  
        Serial.print("Sent ");  
const char *TX_Value="no";  
        Serial.println(TX_Value);  
        Radio.Send((uint8_t *)TX_Value, strlen(TX_Value));  
    }  
}  
if(state == 1){  
    count = 0;  
    digitalWrite(2,LOW);  
}  
if (stateI == 0)  
{  
    countI+=1;  
  
    if(countI == 1){  
        digitalWrite(2,HIGH);  
        Serial.print("Sent ");  
const char *TX_Value="si";  
        Serial.println(TX_Value);  
        Radio.Send((uint8_t *)TX_Value, strlen(TX_Value));  
    }  
}
```

Módulo sensor de material

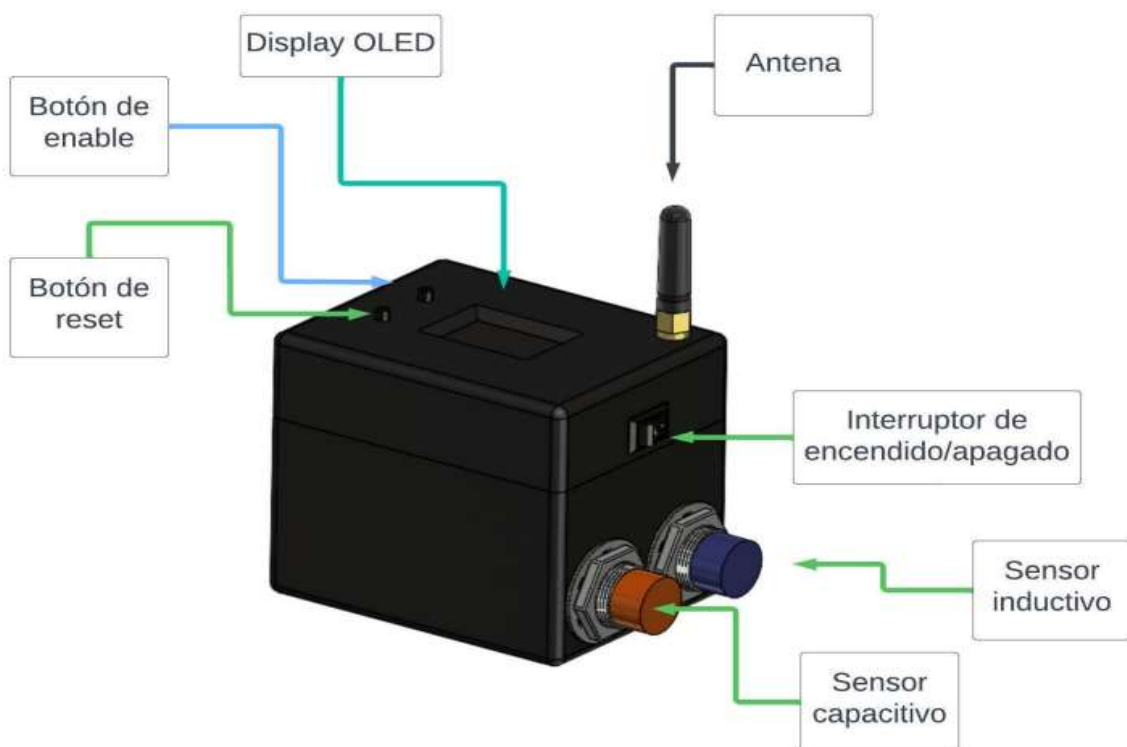
El módulo sensor de material tiene el objetivo de identificar los objetos por material como objetos metálicos y objetos no metálicos.

Características

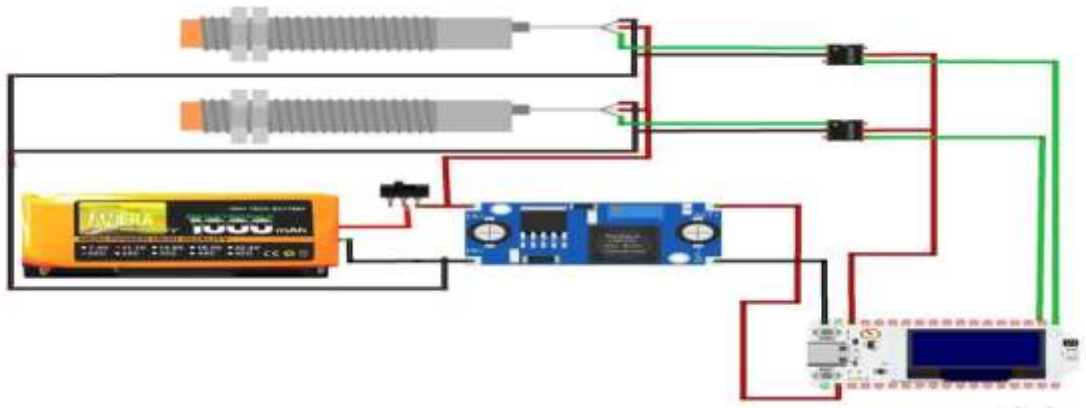


- Sensor capacitivo LJC30A3-H-Z/BX
- Sensor inductivo LJ18A3- 8Z/BX
- Voltaje de alimentación de los sensores de 6 a 32v
- Rango de detección: 8mm
- Duración de la batería: 2 horas
- Cargador: tipo Li-Po
- Programación: cable tipo C mediante IDE Arduino

Partes del módulo



Circuito de conexión interna



Programación del módulo

Al igual que los anteriores, se necesita que envíe palabras cuando el objeto es de algún metal ferroso o no, para ello se utiliza la condición siguiente para cuando el sensor inductivo detecta la presencia de un objeto metálico ferroso envíe la palabra “si”, donde al contrario cuando el sensor capacitivo detecte la presencia de un objeto, pero el sensor inductivo no detecte la presencia de un objeto metálico ferroso, envíe la palabra “no”.

```
bool state = digitalRead(sensorPin);  
bool stateI = digitalRead(sensorPinI);  
  
if (state == 0)  
{  
    count+=1;  
  
    if(count == 1){  
        digitalWrite(2,HIGH);  
        Serial.print("Sent ");  
const char *TX_Value="no";  
        Serial.println(TX_Value);  
        Radio.Send((uint8_t *)TX_Value, strlen(TX_Value));  
    }  
}  
if(state == 1){  
    count = 0;  
    digitalWrite(2,LOW);  
}  
if (stateI == 0)  
{  
    countI+=1;  
  
    if(countI == 1){  
        digitalWrite(2,HIGH);  
        Serial.print("Sent ");  
const char *TX_Value="si";  
        Serial.println(TX_Value);  
        Radio.Send((uint8_t *)TX_Value, strlen(TX_Value));  
    }  
}
```

Módulo sensor de temperatura y humedad

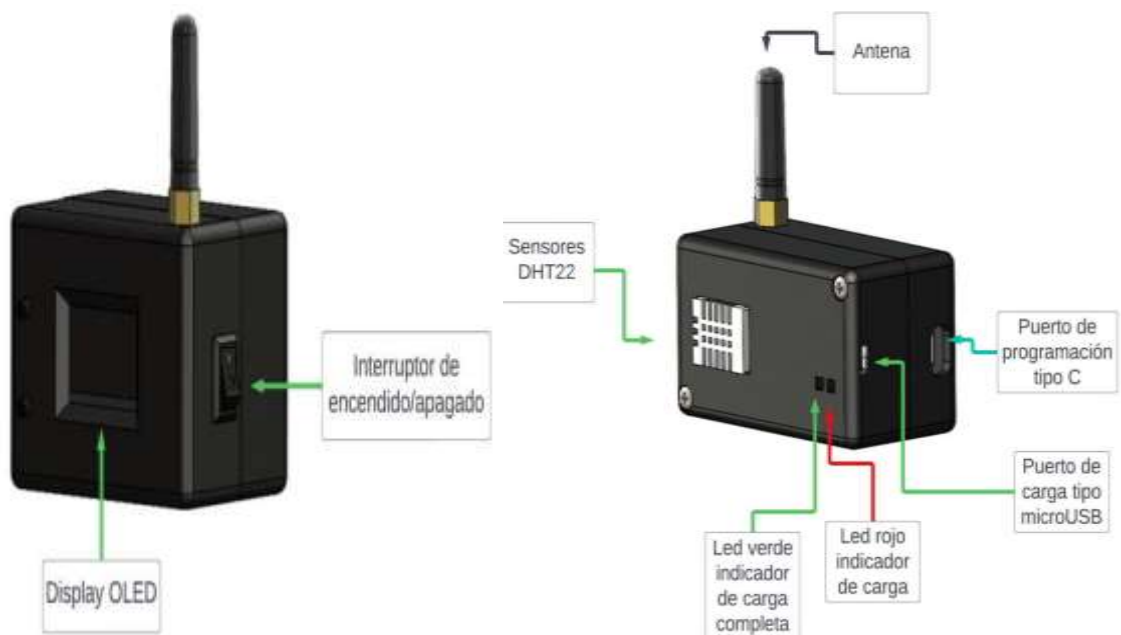
El módulo sensor de material tiene el objetivo de identificar la temperatura del ambiente.

Características

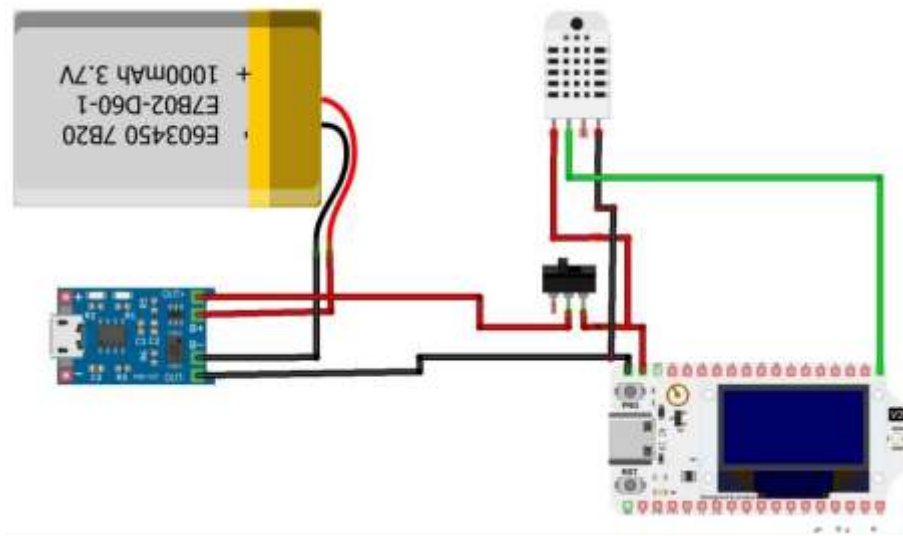


- Sensor digital de humedad y temperatura DHT22 (AM2302)
- Voltaje de alimentación: 3.3V – 5V
- Rango de medición de temperatura: - 40°C a 80 °C
- Rango de medición de humedad: De 0 a 100% RH
- Duración de la batería: 4h 30min
- Cargador: tipo micro USB V8 de 5v
- Programación: cable tipo C mediante IDE Arduino

Partes del módulo



Circuito de conexión interna



Programación del módulo

Para la utilización del sensor de temperatura y humedad DTH22 es necesario la incorporación de la librería “DHT.h” para su correcto manejo, con la diferencia en los anteriores sensores que este detecta los datos de la temperatura y humedad a la vez, donde para transformar a dato de texto se unen los datos, por lo cual es necesario adherir una letra antes del dato para separarlos al momento de enviar y recibir, la letra “T” para diferenciar el dato de la temperatura de la de humedad, como se observa en la figura 636.

```
void loop()
{
    temperatura = dht.readTemperature();
    humedad = dht.readHumidity();
    String txpacket_str = "T"+String(temperatura)+String(humedad);

    if(lora_idle == true)
    {
        txpacket_str.toCharArray(txpacket, BUFFER_SIZE);

        Serial.printf("\r\nSending Temperatura: \"%0.2f\" , length %d\r\n",temperatura, strlen(txpacket));
        Serial.printf("\r\nSending Humedad: \"%0.2f\" , length %d\r\n",humedad, strlen(txpacket));
        delay(1000);
        Radio.Send( (uint8_t *)txpacket, strlen(txpacket));
    }
}
```


Configuración del módulo maestro

Para la programación de la tarjeta embebida Heltec LoRa V3 para la recepción de los datos se necesita inicializarlas de la misma forma que los módulos individuales, ya que para la comunicación debe utilizar la misma frecuencia, donde para la aplicación, lo importante es llevar un conteo de las piezas detectadas por los módulos de sensores, para lo cual se deben colocar contadores y cada vez que reciba el dato de texto el contador sume según los objetos detectados, donde para el accionamiento de los actuadores es controlado con la tarjeta Arduino UNO, para lo cual cuando reciba la palabra del sensor, según corresponda, mande una señal digital al Arduino, comparando la primera letra del dato recibido según el caso.

```
rssi=rssi;
rxSize=size;
memcpy(rxpacket, payload, size );
rxpacket[size]='\0';
Radio.Sleep( );
Serial.printf("\r\nreceived packet \"%s\" with rssi %d , length %d\r\n",rxpacket,rssi,rxSize);
switch (rxpacket[0]){
    case 'R':
        countR++;
        contadorR = countR;
        digitalWrite(2,HIGH);
        delay(100);
        digitalWrite(2,LOW);
        break;
    case 'V':
        countG++;
        contadorG = countG;
        digitalWrite(3,HIGH);
        delay(100);
        digitalWrite(3,LOW);
        break;
    case 'A':
        countB++;
        contadorB = countB;
        digitalWrite(4,HIGH);
        delay(100);
        digitalWrite(4,LOW);
        break;
}
```

En la tarjeta Arduino UNO es necesario inicializar los servomotores e incluir la librería “Servo.h”, donde aquí se puede ajustar los tiempos según el actuador el cual se debe accionar ya que están a diferentes distancias del dispensador, donde para el accionamiento de este es necesario colocar una condición inicial que solo este

encendido cuando exista la presencia de un objeto en la rampa de entrada y a la vez presionemos el pulsador de inicio.

```
unsigned long currentMillis = millis();
if (digitalRead(sensorPin1) != sensor1State) {
  if (digitalRead(sensorPin1) == HIGH) {
    isInitialized = false;
    sensor1State = true;
  } else {
    sensor1State = false;
  }
}
if (digitalRead(sensorPin1) == LOW && !sensor1State && currentMillis - previousMillis1 >= interval1) {
  servo1.write(55);
  delay(2000);
  servo1.write(145);
  delay(1000);
  previousMillis1 = currentMillis;
}
```

De forma parecida para el accionamiento de los actuadores, cuando la señal digital provenga de la tarjeta Heltec LoRa V3, accione un servomotor en específico según corresponde.

```
if (digitalRead(sensorPin2) == HIGH) {
  delay(3800);
  servo2.write(180);
  delay(2000);
  servo2.write(0);
  delay(1400);
  Serial.println("paso 2");
}
if (digitalRead(sensorPin3) == HIGH) {
  delay(2800);
  servo3.write(180);
  delay(2000);
  servo3.write(0);
  delay(1400);
  Serial.println("paso 3");
}
if (digitalRead(sensorPin4) == HIGH) {
  delay(1500);
  servo4.write(180);
  delay(2000);
  servo4.write(0);
  delay(1400);
  Serial.println("paso 3");
}
```

En la Tarjeta ESP32 para el accionamiento del motor NEMA 17, es necesario la condición inicial que solo pueda accionarse cuando exista piezas en la rampa de entrada y se presione el pulsador de inicio, donde para el encendido o apagado desde una aplicación móvil, hay que incluir las condiciones desde la plataforma de Firebase.

```
void loop() {
  if(Firebase.RTDB.getInt(&fbdo, "/user1/banda") ){
    banda=fbdo.intData();
  }
  if (digitalRead(inicio)==HIGH || banda==1) {
    Firebase.RTDB.setInt("/user1/banda",1);
    while(digitalRead(sensor)==LOW){
      if(Firebase.RTDB.getInt(&fbdo, "/user1/banda") ){
        banda=fbdo.intData();
      }
      if(banda == 0){
        break;
      }
      digitalWrite(led, LOW);
      digitalWrite(dirPin, LOW);
      for (int i = 0; i < stepsPerRevolution; i++) {
        digitalWrite(stepPin, HIGH);
        delayMicroseconds(2000);
        digitalWrite(stepPin, LOW);
        delayMicroseconds(2000);
      }
      digitalWrite(dirPin, HIGH);
      digitalWrite(led, HIGH);
      delayMicroseconds(6000);
    }
  }
}
```

Nota:

Para poder programar las placas del módulo maestro retirar la tapa posterior del tablero de control y conectar la tarjeta embebida al computador.

Conectar el módulo maestro a internet

Para poder conectar el modulo maestro a internet se debe tener el nombre de la red a la cual se va a conectar y la clave la misma la cual será ingresada en la programación del mismo.

```
6  #define WIFI_SSID "NombreRED"
7  #define WIFI_PASSWORD "ClaveRED"
```

Visualización de datos en Firebase

Para poder ingresa a la cuenta de Firebase primero se debe iniciar sesión con la cuenta de Gmail de la estación, la clave será otorgada por el docente a cargo.



Google
Clasificadora

clasificadoraespe@gmail.com

Ingresa tu contraseña

Mostrar contraseña

¿Olvidaste la contraseña?

Siguiente

Al ingresar a la plataforma Firebase presionar en el botón “ir a consola”



Seleccionar el proyecto “clasificadora”



En “Compilacion” seleccionar “RealTime Database”

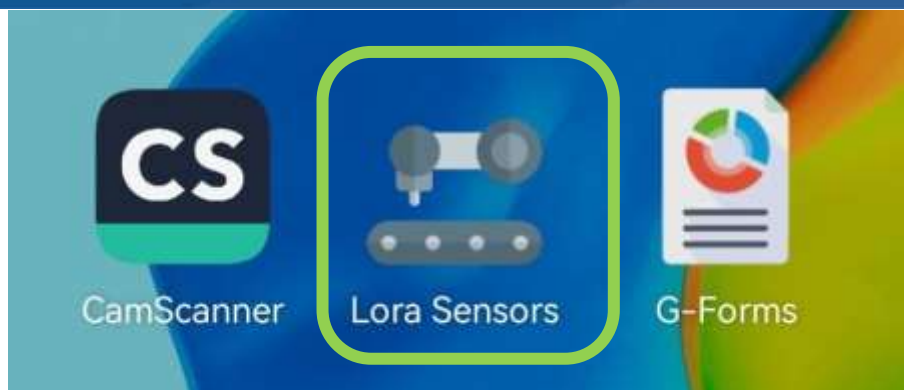


Se desplegará así la consola de los datos del sistema.

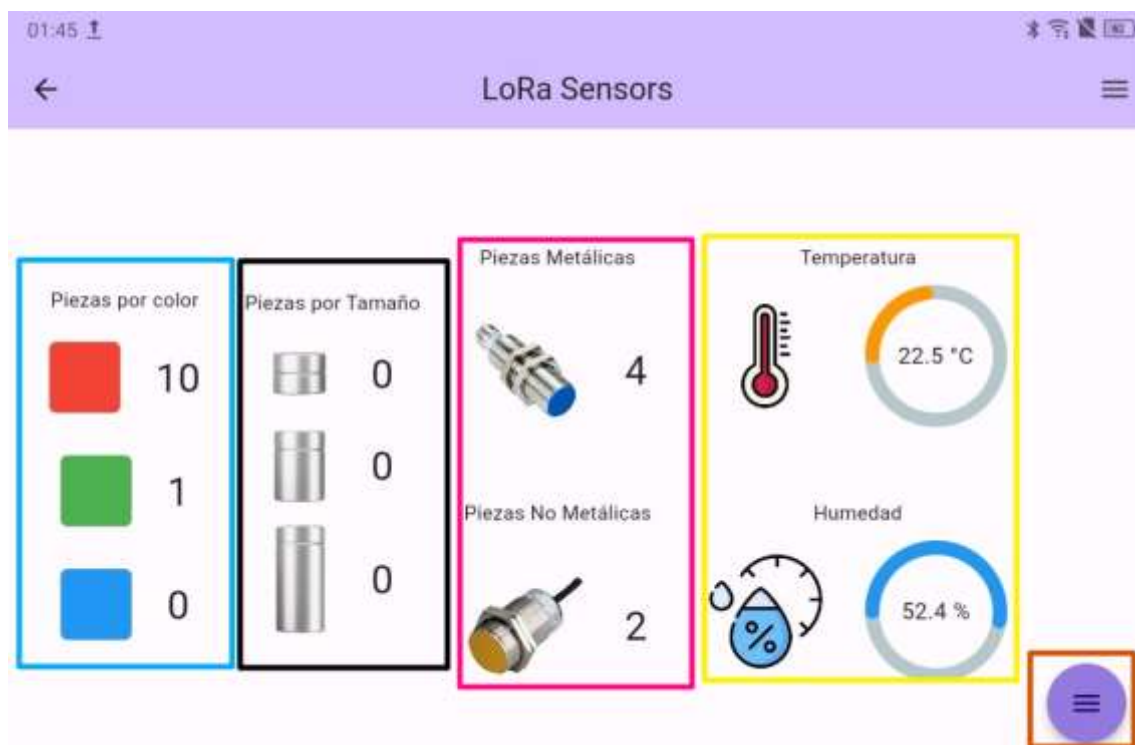


Instrucciones para el uso de la aplicación móvil

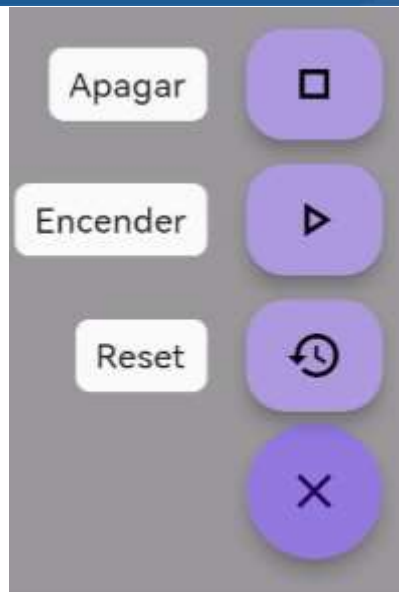
Una vez descarga e instalada la [aplicación móvil](#) en algún dispositivo conectado a internet, en la pantalla del celular aparecerá el icono de la aplicación.



Se despliega una pantalla con varios widgets para los contadores de los sensores de color, tamaño, material y indicadores para el sensor de temperatura y humedad



En la parte inferior existe un icono que al presionarlo aparecen los botones para prender o apagar la banda y un botón para el reseteo de los contadores.



La aplicación también dispone de icono para escanear los códigos QR de los módulos inalámbricos.



Cuando el botón de paro sea presionado se desplegará una pantalla de bloqueo la cual no se podrá retirar hasta que el paro sea desenclavado.



Puesta en marcha

Para poner en marcha el sistema de clasifco se debe seguir los siguientes pasos:

1. En la parte inferior de la mesa encender la fuente de alimentación



2. Verificar y comprobar que el maestro esté conectado a internet
3. Seleccionar el módulo para la clasificación de los objetos y colocarlo sobre el soporte del módulo.



4. Encender el módulo inalámbrico y abrir la aplicación móvil en la Tablet



5. Colocar los objetos de clasificación según el módulo de clasificación seleccionado.



6. Quitar el paro de paro de emergencia.



7. Pulsar el pulsador de inicio y verificar que las piezas se dispensen correctamente



8. Observar que los objetos de clasificación sean detectados por el módulo de sensor.



9. Verificar si los objetos fueron clasificados por los actuadores y comprobar en la aplicación móvil que los datos detectados por el sensor sean actualizados.



10. Si se desea cambiar el tipo de clasificación utilizar y colocar otro modulo y repetir los pasos 7, 8 y 9.
11. Al finalizar la practica apagar el módulo inalámbrico y retirarlo del soporte.



12. Retirar los objetos de clasificación del dispensador y presionar el paro de emergencia



13. Apagar la fuente.



Instrucciones para cargar los módulos

Para los módulos inalámbricos de temperatura humedad, color y tamaño seguir los siguientes paso continuación:

1. Apagar el módulo inalámbrico y no conectarlo al computador mientras se carga.



2. Utilizando un cable tipo micro usb conectar al puerto de carga y verificar que un led rojo se encienda.



3. Cuando el módulo haya carga completamente el led cambiara de color rojo a azul.



Para el módulo inalámbrico de material seguir los siguientes paso continuación:

1. Apagar el módulo inalámbrico y no conectarlo al computador mientras se carga.



2. Utilizando un cargador para baterías tipo lipo de 3 celda conectar la batería.



3. Cuando el módulo haya carga completamente el cargador sonara la alarma.



Plan de mantenimiento

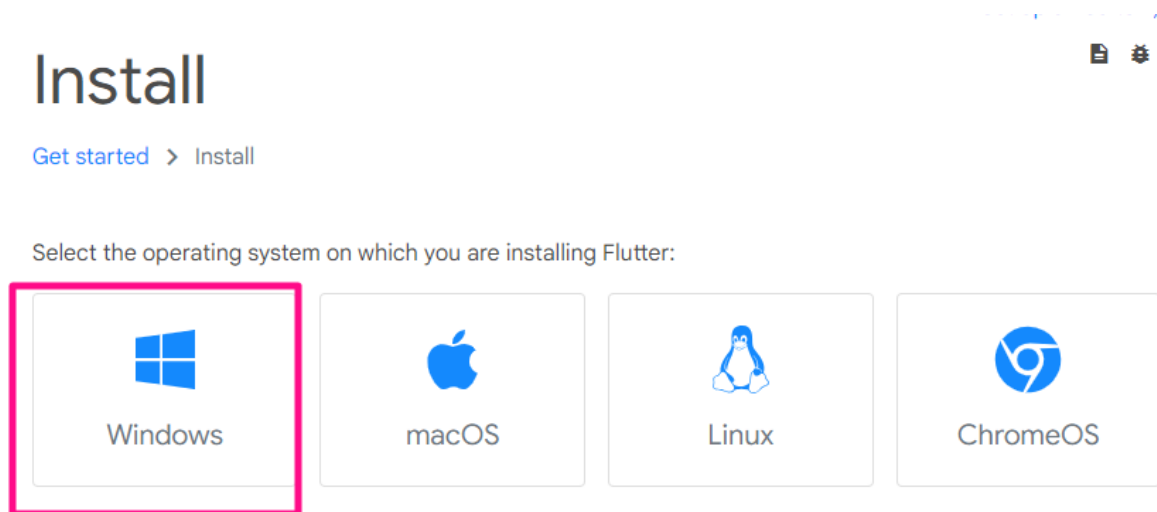
PLAN DE MANTENIMIENTO				
TAREAS	SUB-TAREAS	GAMA		
		Diarias	Mensuales	Anual
Lubricación	Lubricación de chumaceras de la banda		X	
Eléctricas	Revisión de motor de la banda		X	
	Inspección de paradas de emergencia	X		
	Revisión de tablero de mando		X	
Mecánicas	Revisión de apriete de tornillería		X	
	Cambio de chumaceras de la banda			X
	Revisión de rodillos		X	
	Inspección de tensión y estado de la banda			X
	Cambio de correas			X
Instrumentación	Limpieza de rampas	X		
	Cambio de baterías			X
	Calibración de sensores	X		

Proceso de mantenimiento de sistema

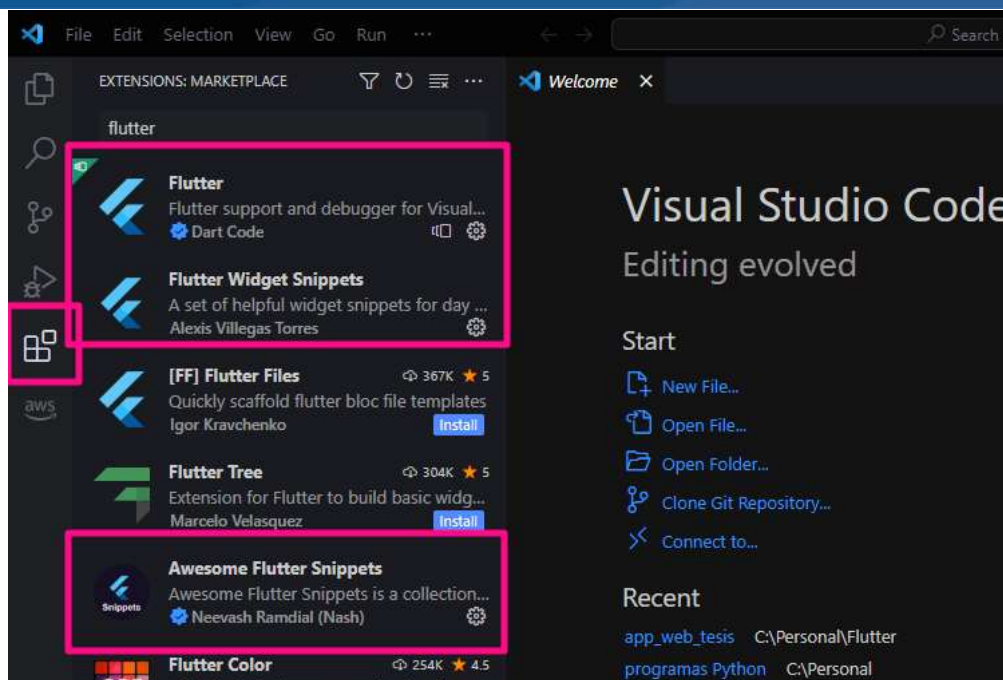
1. Revisar el sistema: El usuario debe revisar la máquina después de utilizarla.
2. Limpiar rampas: El usuario debe limpiar las rampas antes utilizar para evitar que los objetos de clasificación se atoren.
3. Aceitar máquina: El usuario debe aceitar o engrasar la máquina para correcto funcionamiento.
4. Calibración: antes de poner en marcha el sistema el usuario deberá verificar que los sensores estén correctamente calibrados.
5. Actuadores: el usuario debe revisar que los actuadores del sistema estén correctamente funcionando.

Instalar Flutter en un pc con Windows

Ingresa a la página oficial de Flutter <https://docs.flutter.dev/get-started/install> e instala la versión de Windows o el sistema operativo con el que se desea trabajar como se muestra en la siguiente imagen y seguir las instrucciones de la documentación oficial.



Como editor de código se recomienda usar Visual Studio Code, dentro de visual estudio ingresar a la sección de extensiones e instalar los plugins para poder utilizar flutter.



En la sección de búsqueda escribir Flutter, el plugin que es necesario es el primero de la imagen anterior que lleva el mismo nombre del framework, los otros que se ven seleccionados son opcionales, pero sí recomendables para codificar más rápido.

Una vez instalada y configurado las herramientas necesarias para su uso se procede a crear un nuevo proyecto con el comando CTRL+SHIFT+P y crear un nuevo proyecto de flutter escribiendo “flutter: new project”

Configuración de Firebase con Flutter

Ingresa en la documentación de Firebase con el enlace <https://firebase.google.com/docs/cli#install-cli-windows> para instalar la línea de comando de Firebase en el computador.

Set up or update the CLI



Install the Firebase CLI

You can install the Firebase CLI using a method that matches your operating system, experience level, and/or use case. Regardless of how you install the CLI, you have access to the same functionality and the `firebase` command.

Windows

macOS

Linux

Para que estos pasos funcionen de forma correcta se debe tener instalado Node.js, el cual se puede descargar e instalar desde su web oficial <https://nodejs.org/en>, la versión de nodejs utilizada en este trabajo de titulación es la versión v18.15.0 y la versión de firebase 12.4.7

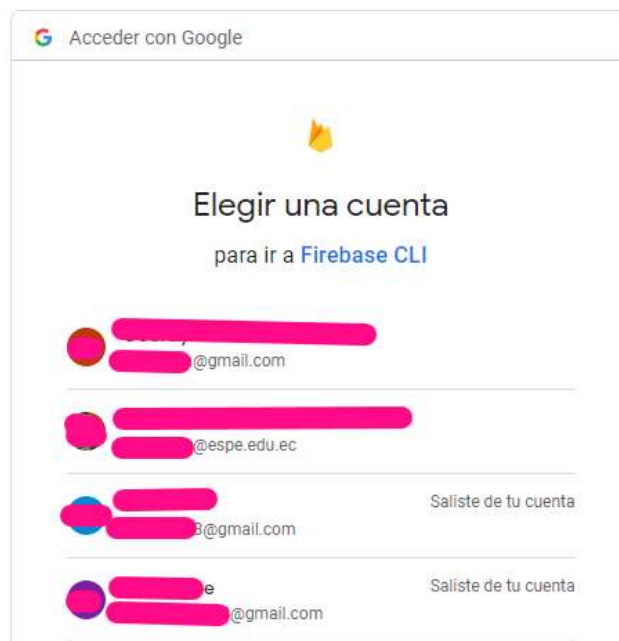
Para inicializar firebase en un proyecto de Flutter nuevo o uno ya existente se debe iniciar sesión en la consola con el comando "firebase login" como se muestra a continuación.

```
PS C:\Personal\Flutter\app_web_tesis> firebase login
i  Firebase optionally collects CLI and Emulator Suite usage and error reporting information to help improve our products. Data is collected in accordance with Google's privacy policy (https://policies.google.com/privacy) and is not used to identify you.
> Allow Firebase to collect CLI and Emulator Suite usage and error reporting information? Yes
i  To change your data collection preference at any time, run "firebase logout" and log in again.

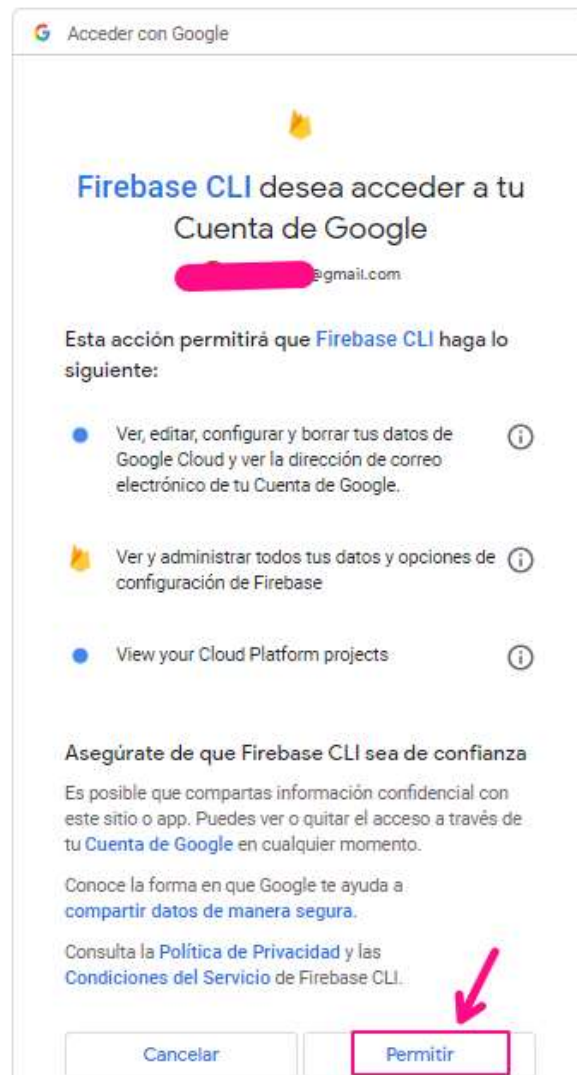
Visit this URL on this device to log in:
https://accounts.google.com/o/oauth2/auth?client_id=563584335860-fgrhmd47bqnekij5i8b5pr03ho849e6.apps.googleusercontent.com&scope=om
ail%20openid%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcloudplatformprojects.readonly%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Ff
irebase%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcloud-platform&response_type=code&state=138860946&redirect_uri=http%3A%2F%2Flocal
host%3A9085

Waiting for authentication...
[]
```

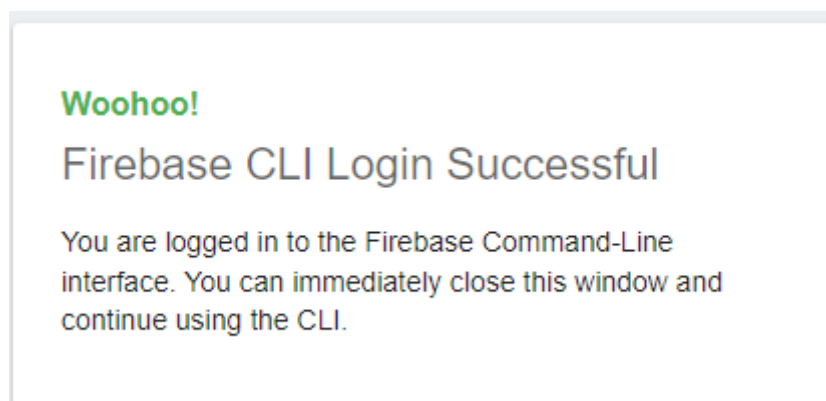
Al ejecutar ese comando en la ruta del proyecto de flutter creado se crea un link para iniciar sesión en la cuenta de Google que se vaya a usar para el firebase de la aplicación y se debe abrir una ventana en el navegador o en caso de que no se abra se puede copiar y pegar este link en el navegador de forma manual.



Se selecciona la cuenta a utilizar y se permiten las políticas de Google para el uso de la cuenta.



Si el ingreso fue exitoso se debe desplegar el siguiente mensaje.



Y en la consola de comandos se debe mostrar el email con el cual se configuró la cuenta de firebase.

```
Visit this URL on this device to log in:
https://accounts.google.com/o/oauth2/auth?client_id=563584335860-fgghnd47bqnekij5i8b5pr03ho849e6.apps.googleusercontent.com&scope=em
ail%20openid%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcloudplatformprojects.readonly%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Ff
irebase%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcloud-platform%2Fresponse_type=code&state=138860946&redirect_uri=http%3A%2F%2Flocal
host%3A9065

Waiting for authentication...

+ Success! Logged in as [redacted]@gmail.com
PS C:\Personal\Flutter\app_web_tesis>
```

Para cambiar de cuenta se puede usar el comando “firebase logout” y se vuelve a iniciar el proceso de ingreso de la nueva cuenta con el comando “firebase login”

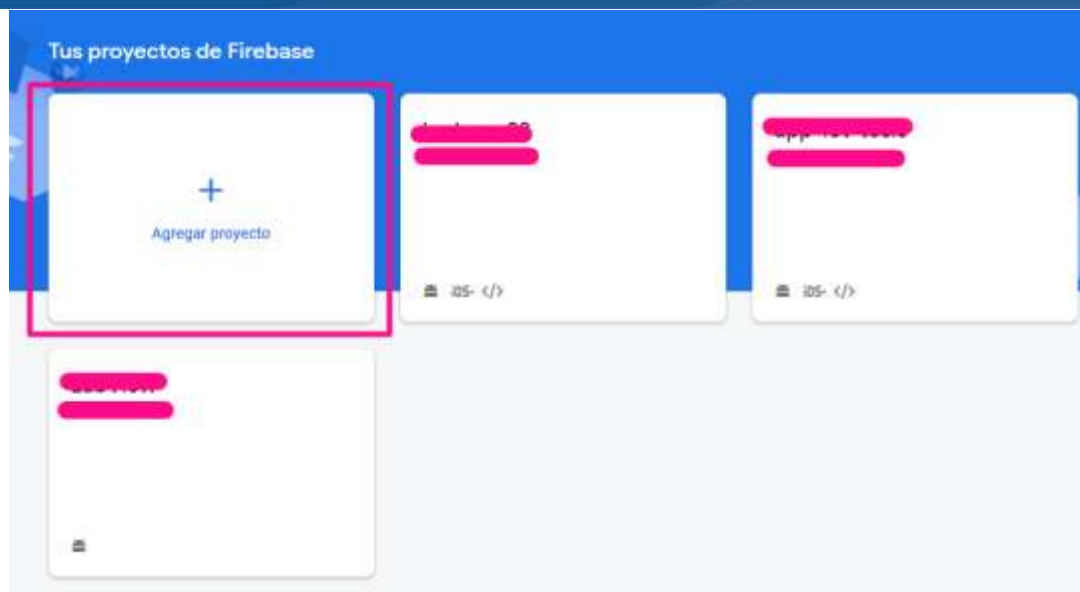
Para la interacción directa entre Firebase y Flutter se instala el componente FlutterFire CLI ejecutando el siguiente comando desde cualquier directorio:

“dart pub global activate flutterfire_cli”

```
PS C:\Personal\Flutter\app_web_tesis> dart pub global activate flutterfire_cli
Package flutterfire_cli is currently active at version 0.2.7.
  cli_util 0.3.5 (0.4.0 available)
> collection 1.18.0 (was 1.17.2)
  deep_pick 0.10.0 (1.0.0 available)
> ffi 2.1.0 (was 2.0.2)
  file 6.1.4 (7.0.0 available)
  http 0.13.6 (1.1.0 available)
  process 4.2.4 (5.0.0 available)
  pub_updater 0.2.4 (0.3.1 available)
> win32 5.0.6 (was 5.0.5)
Building package executables... (39.3s)
Built flutterfire_cli:flutterfire.
Installed executable flutterfire.
Activated flutterfire_cli 0.2.7.
```

En la imagen se puede observar los paquetes y sus versiones que utiliza firebase para la comunicación con flutter.

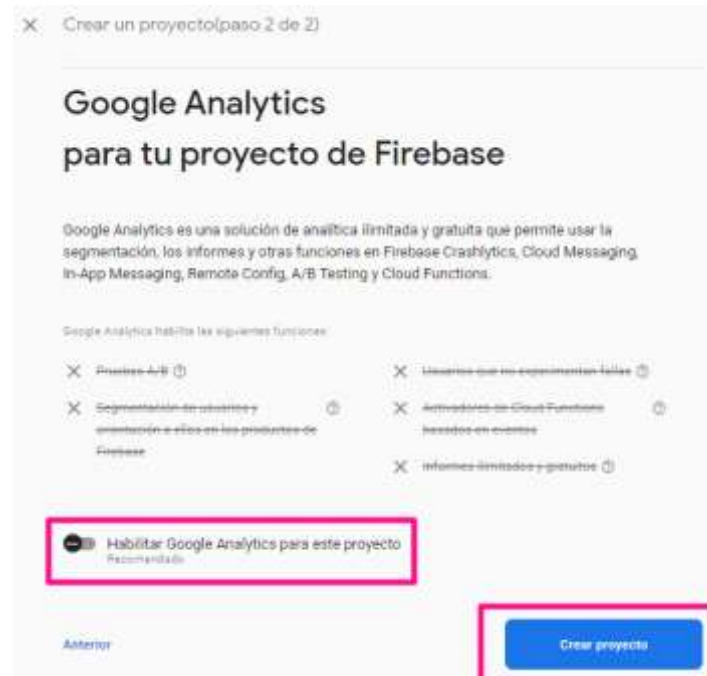
En el navegador crear un nuevo proyecto de Firebase.



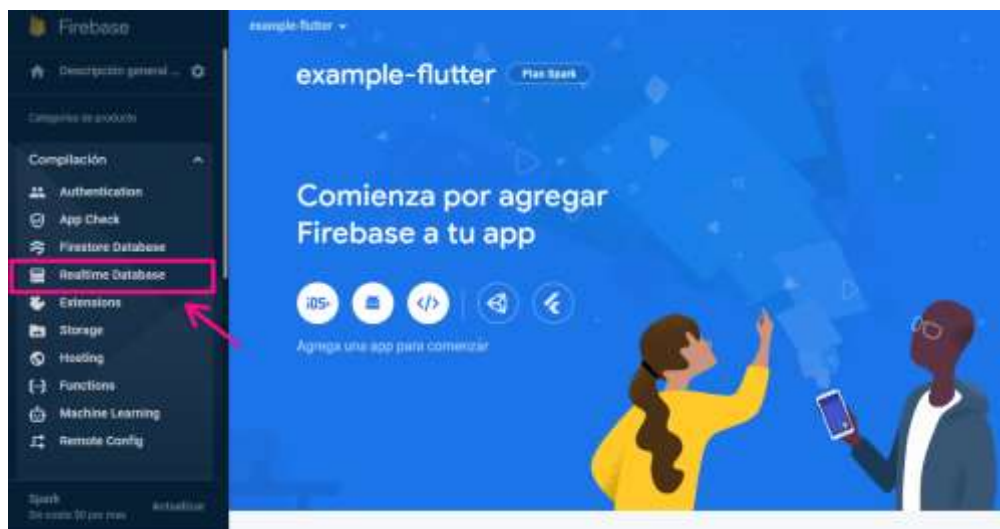
Asignar un nombre al proyecto y clickear el botón de continuar



Deshabilitar las opciones de Google Analytics



Seleccionar la opción de la base de datos en tiempo real



Se abre una ventana con la opción de crear una base de datos.

Realtime Database

Almacena y sincroniza datos en tiempo real

Crear una base de datos

Seleccionar el servidor más cercano, para este caso de Estados Unidos y se da en continuar

Configurar base de datos

1

Opciones de base de datos

2

Reglas de seguridad

El parámetro de configuración de la ubicación determina en qué lugar se almacenarán tus datos de Realtime Database.

Ubicación de Realtime Database

Estados Unidos (us-central1)

Cancelar

Siguiente

Se utiliza el modo de prueba para poder escribir y leer los datos sin restricciones

Configurar base de datos



-  Opciones de base de datos — **2 Reglas de seguridad**

Cuando definas la estructura de los datos, deberás crear reglas para protegerlos.

[Más información](#) 


☐ Comenzar en modo bloqueado

Tus datos son privados de forma predeterminada. El acceso de lectura/escritura de los clientes solo se otorgará como se indica en tus reglas de seguridad.

☒ Comenzar en modo de prueba

Para permitir una configuración rápida, los datos se abren de forma predeterminada. Sin embargo, debes actualizar las reglas de seguridad dentro de 30 días a fin de habilitar el acceso de lectura/escritura a largo plazo para los clientes.

```
{
  "rules": {
    ".read": "now < 1694322000000", // 2023-9-10
    ".write": "now < 1694322000000", // 2023-9-10
  }
}
```

 Las reglas de seguridad predeterminadas del modo de prueba permiten que cualquier usuario con acceso a tu referencia de base de datos pueda ver, editar y borrar todos los datos durante los siguientes 30 días.

Cancelar

Habilitar

Ya creada la base de datos se crean las variables del proyecto clickeando en el símbolo que se muestra en la imagen.

example-flutter ▾

Realtime Database

Datos

Reglas

Copias de seguridad

Uso

Extensiones **NUEVA**

Protege tus recursos de Realtime Database contra los abusos, como fraudes de facturación o suplantación de identidad.

<https://example-flutter-f0722-default-rtdb.firebaseio.com>

<https://example-flutter-f0722-default-rtdb.firebaseio.com/>

+



Las variables deben tener exactamente el mismo nombre que se muestra en la siguiente imagen.

```
user1
├── alerta: 0
├── banda: 0
├── contadorB: 0
├── contadorG: 0
├── contadorR: 0
├── counterMetalicas: 11
├── counterNoMetalicas: 12
├── humedad: 82
├── piezas_g: 4
├── piezas_m: 5
├── piezas_p: 8
├── statusReset: false
└── temperatura: 34
```


Desde el directorio de su proyecto Flutter, ejecute el siguiente comando para iniciar el flujo de trabajo de configuración de la aplicación.

“flutterfire configure”

```
PS C:\Personal\Flutter\app_web_tesis> flutterfire configure
i Found 3 Firebase projects.
? Select a Firebase project to configure your Flutter application with >
example-flutter-f0722 (example-flutter)
&ltcreate a new project>
```

Como se puede observar se despliegan los proyectos existentes en la cuenta de firebase incluido el nuevo proyecto creado, con las flechas del teclado se seleccionan el proyecto con el cual se quiere vincular a la app.

```
PS C:\Personal\Flutter\app_web_tesis> flutterfire configure
i Found 3 Firebase projects.
✓ Select a Firebase project to configure your Flutter application with · example-flutter-f0722 (example-flutter)
? Which platforms should your configuration support (use arrow keys & space to select)? >
✓ android
✓ ios
✓ macos
✓ web
```

Seleccionar hasta abajo al apartado de web para poder inicializar el proyecto con Flutter.

```
PS C:\Personal\Flutter\app_web_tesis> flutterfire configure
i Found 3 Firebase projects.
✓ Select a Firebase project to configure your Flutter application with · example-flutter-f0722 (example-flutter)
✓ Which platforms should your configuration support (use arrow keys & space to select)? · android, ios, macos, web
i Firebase android app com.example.app_web_tesis is not registered on Firebase project example-flutter-f0722.
i Registered a new Firebase android app on Firebase project example-flutter-f0722.
i Firebase ios app com.example.app_web_tesis is not registered on Firebase project example-flutter-f0722.
i Registered a new Firebase ios app on Firebase project example-flutter-f0722.
i Firebase macos app com.example.app_web_tesis is not registered on Firebase project example-flutter-f0722.
i Registered a new Firebase macos app on Firebase project example-flutter-f0722.
i Firebase web app app_web_tesis (web) is not registered on Firebase project example-flutter-f0722.
i Registered a new Firebase web app on Firebase project example-flutter-f0722.
✓ Generated FirebaseOptions file lib\Firebase_options.dart already exists, do you want to override it? · yes
✓ Generated FirebaseAppId file C:\Personal\Flutter\app_web_tesis\ios\Firebase_app_id file.json already exists, do you want to override it? · yes
✓ Generated FirebaseAppId file C:\Personal\Flutter\app_web_tesis\macos\Firebase_app_id file.json already exists, do you want to override it? · yes
? The google-services.json file already exists but for a different Firebase project (classificadora-93348). Do you want to replace it with Firebase project example-flutter-f0722? · yes

Firebase configuration file lib\Firebase_options.dart generated successfully with the following Firebase apps:

Platform  Firebase App Id
web       1:761884609503:web:17f4ad808d494a073e5232
android   1:761884609503:android:f0e362eadc80f33e5232
ios       1:761884609503:ios:0d26cf5d8000b63e5232
macos     1:761884609503:ios:0f19218b232daf3e5232

Learn more about using this file and next steps from the documentation:
> https://firebase.google.com/docs/flutter/setup
PS C:\Personal\Flutter\app_web_tesis>
```

Se aceptan todas las condiciones porque se crean nuevos archivos ya existentes, pero con la cuenta de firebase seleccionada.

Ingresa a <https://pub.dev/> para descargar los paquetes de flutter que se muestran a continuación

cupertino_icons: ^1.0.2

firebase_core: ^2.13.1

firebase_database: ^10.2.2

percent_indicator: ^4.2.3

provider: ^6.0.5

flutter_speed_dial: ^7.0.0

flutter_pdfview: ^1.3.1

file_picker: ^5.3.2

path_provider: ^2.0.15

path: ^1.8.3

flutter_barcode_scanner: ^2.0.0

animate_do: ^3.0.2

shared_preferences: ^2.2.0

syncfusion_flutter_charts: ^22.2.8

La version de flutter y dart usadas en este proyecto son:

Flutter sdk 3.10.6

Dart 3.0.6