



# Documentación Técnica

Fecha: 14/11/2025

Este documento contiene la documentación técnica para el proyecto SafeStep, perteneciente al curso Laboratorio STEAM+ de la Tecnicatura en Redes y Software del Instituto Tecnológico de Informática, año 2025.

## Proyecto: SafeStep

### 1. Integrantes

- Aaron Alonso
- Santiago Viñas
- Juan Beracocha



### 2. Descripción

#### 2.1 Descripción general del proyecto:

SafeStep es una pulsera inteligente diseñada para mejorar la autonomía y seguridad de las personas con baja visión o ceguera. El dispositivo utiliza sensores de proximidad para detectar obstáculos en el entorno y alertar al usuario mediante pulsos de vibración.

Su objetivo principal es facilitar un recorrido más seguro y brindar mayor independencia a las personas con baja visibilidad en su día a día. Debido al bajo costo de los materiales para su fabricación, SafeStep, como producto final, a diferencia de otras opciones que existen en el mercado, será accesible.



económicamente para que cualquier persona con problemas visuales o ceguera pueda obtenerla. También se diferenciará en su simplicidad, escasa invasividad y practicidad.

## **2.1 Descripción del prototipo:**

Como podemos observar, hasta el momento, **SafeStep aún es un prototipo. El producto final aún no fue presentado.** Este prototipo cuenta con los materiales, código, funcionamiento, diseño mecánico y diseño de software que a continuación se describen.

## **2.2 Problema que busca resolver:**

Las personas con discapacidad visual enfrentan dificultades al desplazarse de forma autónoma, especialmente en entornos desconocidos o con obstáculos inesperados. Esto genera inseguridad, dependencia y riesgo de accidentes. SafeStep busca que sea accesible económicamente para todo el mundo, y tener un diseño ergonómico para brindar confort y practicidad.

## **2.2 Solución propuesta:**

SafeStep actúa como un guía portátil. Detecta obstáculos cercanos mediante un sensor de proximidad infrarrojo y activa un motor vibratorio para advertir al usuario.

## **3. Materiales**



- Placa Micro:bit



- Sensor de proximidad infrarrojo KY-032



- Motor vibratorio (recuperado de un teléfono móvil)



- Transistor MOSFET



- Protoboard



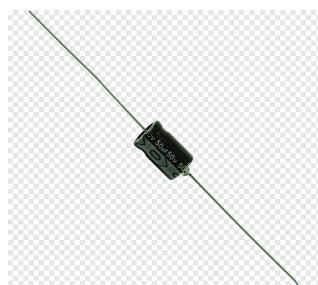
- Resistencias



- Cables Dupont



- Diodo

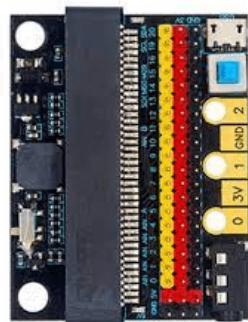




- PowerBank



- Adaptador edge connector para Micro:bit



- Caja para el ensamblaje





## 4. Diseño Mecánico

### Funcionamiento general:

El prototipo se compone de una pequeña carcasa que aloja la placa Micro:bit, el sensor de proximidad, el motor vibratorio y el powerbank. El sensor se orienta hacia adelante para detectar obstáculos en la trayectoria del usuario, mientras que el motor se ubica en contacto con la piel (muñeca o antebrazo) para transmitir vibraciones perceptibles.

### 4.1 Ensamblaje:

1. Colocar el Micro:bit dentro de una caja pequeña.
2. Conectar el sensor KY-032 orientado hacia el frente.
3. Ubicar el motor vibratorio en la parte inferior, en contacto con la piel.
4. Conectar el powerbank al usb del micro:bit.

## 5. Diseño Electrónico

### 5.1 Descripción:

El sensor KY-032 emite un haz infrarrojo que rebota en los objetos cercanos. Si detecta una reflexión dentro del rango configurado, envía una señal digital al pin P0 del Micro:bit.

El Micro:bit interpreta esa señal y activa el motor vibratorio a través de un transistor MOSFET, que actúa como interruptor y evita sobrecargar la placa.



## 5.2 Conexiones principales:

Cable dupont marrón: GND del sensor al negativo de protoboard

Cable dupont naranja: VCC del sensor al 3.3v del microbit

Cable dupont amarillo: Output del sensor al P0-Analog del microbit

Cable dupont rojo: Enable del sensor al 5v del microbit

Cable dupont violeta: P1 del microbit al D12 del protoboard

Cable dupont blanco: del 3.3v del microbit al positivo del protoboard

Cable dupont gris: del GND del microbit al negativo del protoboard

Cable dupont negro: de GND del microbit al negativo del protoboard

Motor: A8 del protoboard y positivo del protoboard

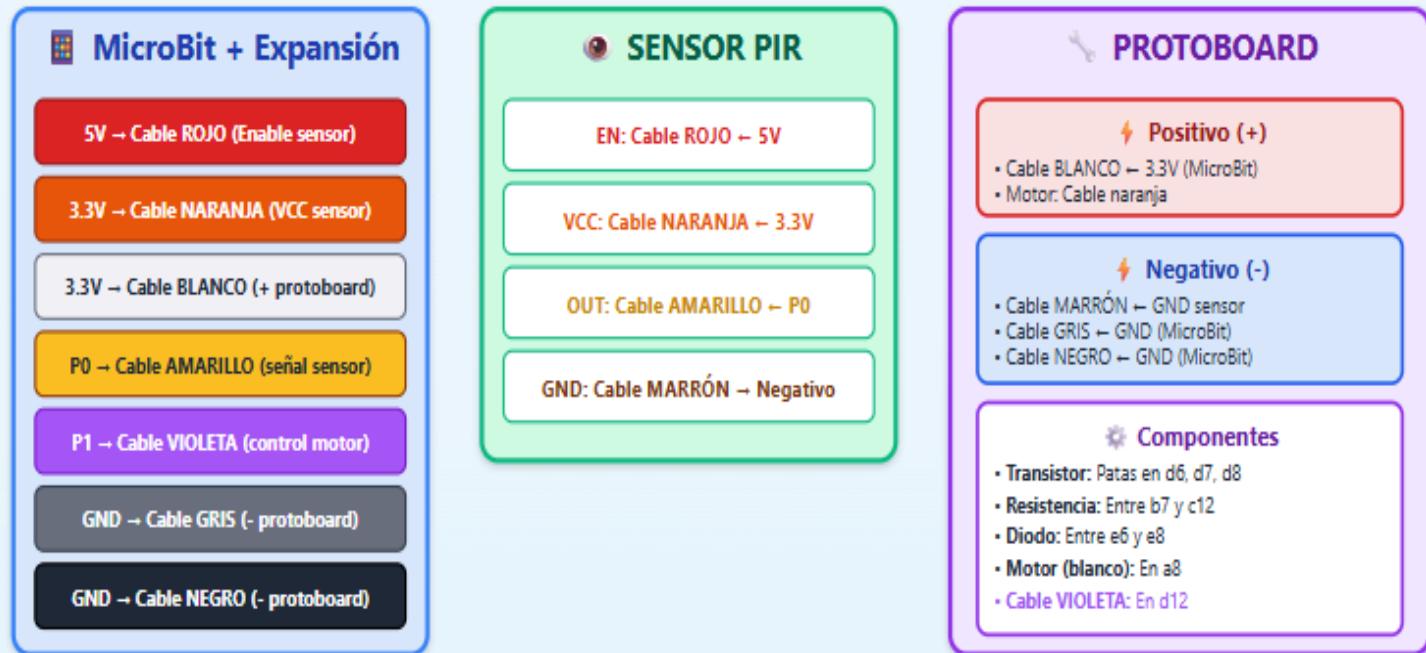
Transistor: D6, D7 y D8 del protoboard

Resistencia: B7 y C12 del protoboard

Diodo: E6 y E8 del protoboard



### 5.3 Diagrama de conexiones:



### ⚡ CIRCUITO DEL MOTOR EN PROTOBOARD

#### ⌚ Flujo de Señal

1. P1 → cable VIOLETA → d12
  2. d12 → RESISTENCIA (c12 a b7)
  3. b7 → Base del TRANSISTOR (d7)
  4. Transistor controla MOTOR en a8
- ✓ El transistor actúa como interruptor

#### 💡 Transistor NPN

- **BASE (d7):** Recibe señal de P1
  - **COLECTOR:** Conecta con motor (a8)
  - **EMISOR:** Conecta a GND (-)
- ⚠ Diodo (e6 a e8): Protege de voltaje inverso generado por el motor

### 🔋 ALIMENTACIÓN

- ⚡ Power Bank conectado por USB al MicroBit
- ⚡ Salida: 5V distribuido a través de la tarjeta de expansión



## 6. Diseño Software

### 6.1 Descripción general:

El programa del Micro:bit lee constantemente la señal del sensor KY-032.

Cuando el valor supera un umbral establecido, el motor vibratorio se activa. A su vez, utilizamos el sistema de sonido propio del micro:bit, para que, cuando el motor vibre, se emita un sonido.

### 6.2 Bloques de código importantes:

- Lectura analógica del pin P0.
- Condicional if para comparar el valor con el umbral ( $\approx 500$ ).
- Activación del pin de salida para encender el motor.
- Activación del sonido al ser activado el motor vibratorio.

### 6.3 Código fuente (MicroPython):

```
from microbit import *
import music

# Configuración de pines
sensor_pir = pin0 # Sensor de movimiento conectado a P0
motor = pin1 # Pin que controla el transistor

# Mensaje inicial
```



```
display.scroll("OK")

# Espera inicial para que el sensor se estabilice
sleep(2000)
display.scroll("LISTO")

# Bucle principal
while True:
    # Lee el sensor como analógico para ver valores
    valor_analogico = sensor_pir.read_analog()

    # Muestra el valor cada 500ms
    print(valor_analogico) # Esto se ve en la consola si conectas por USB

    # Si el valor es MENOR al umbral, hay movimiento
    if valor_analogico < 500:
        motor.write_digital(1) # Enciende motor cuando detecta
        music.pitch(440, 100)
    else:
        motor.write_digital(0) # Apaga motor cuando NO detecta
        music.stop()

    sleep(200)
```

#### 6.4 Explicación detallada del código:

**Líneas 1-2:** Importa librerías necesarias (microbit para hardware, music para sonido)

**Líneas 5-6:** Define pines de entrada (sensor en P0) y salida (motor en P1)

**Líneas 9-12:** Mensaje inicial "OK" y estabilización del sensor (2 segundos), luego "LISTO"



**Línea 15:** Inicia bucle infinito para lectura continua

**Línea 17:** Lee valor analógico (0-1023) del sensor

**Línea 20:** Imprime valor en consola para debug

**Línea 23:** Compara con umbral 500 (ajustable según necesidad)

**Líneas 24-26:** Si detecta (valor < 500): enciende motor, muestra cara feliz, emite beep

**Líneas 28-30:** Si NO detecta: apaga motor, cara triste, sin sonido

**Línea 32:** Pausa de 200ms entre lecturas

## 7. Solución de problemas comunes

Problema	Causa probable	Solución
Motor no vibra	Conexión incorrecta del pin P1	Verificar cable violeta esté en P1 y D12
Motor no vibra	Transistor mal conectado	Verificar patas en D6, D7, D8



<b>No detecta objetos</b>	<b>Sensor mal orientado</b>	<b>Apuntar sensor hacia adelante</b>
<b>No detecta objetos</b>	<b>Cable amarillo suelto</b>	<b>Verificar conexión en P0</b>
<b>Vibra constante mente</b>	<b>Umbral muy alto</b>	<b>Ajustar valor en código (línea 23) a mayor</b>
<b>Vibra constante mente</b>	<b>Objeto fijo cerca del sensor</b>	<b>Alejar sensor de superficies</b>
<b>Power bank se apaga</b>	<b>Auto-apagado por bajo consumo</b>	<b>Presionar botón cada 10-15 min</b>
<b>No enciende</b>	<b>Cable USB suelto</b>	<b>Verificar conexión USB al MicroBit</b>
<b>No enciende</b>	<b>Power bank descargado</b>	<b>Recargar power bank</b>
<b>MicroBit se reinicia</b>	<b>Sobrecarga eléctrica</b>	<b>Verificar que cable blanco esté en 5V no 3.3V</b>



<b>Código no responde</b>	<b>Error en carga del código</b>	<b>Volver a cargar archivo .hex</b>
<b>Sin sonido</b>	<b>Volumen del MicroBit bajo</b>	<b>No hay control de volumen, es fijo</b>

## 8. Futuras Mejoras

SafeStep será adaptable a mejoras significativas e increíblemente beneficiosas para el usuario. Entre las que están proyectadas, se destacan:

### 8.1 Mejoras de hardware:

- Sensor ultrasónico (HC-SR04): Mayor precisión y alcance (hasta 4 metros)
- Batería recargable integrada: Eliminar necesidad de power bank externo
- Múltiples motores: Diferentes intensidades según distancia (cerca/medio/lejos)
- LED indicador: Mostrar nivel de batería
- Carcasa impresa en 3D: Diseño ergonómico y profesional
- Sensor de temperatura: Alertas adicionales

### 8.2 Mejoras de software:

- Conectividad Bluetooth: Enviar alertas al celular
- Registro de rutas: Guardar trayectos frecuentes
- Calibración por app: Ajustar sensibilidad sin código
- Comandos de voz: Integración con asistentes virtuales



- Patrones de vibración personalizables: Diferentes alertas según tipo de objeto

## 9. Registro de Avances

- **8/08/2025:** Lluvia de ideas y selección del proyecto SafeStep.
- **15/08/2025:** Elección de plataforma (Micro:bit) y diseño inicial del sitio web.
- **22/08/2025:** Asignación de roles, creación del logotipo y listado de materiales.
- **29/08/2025:** Planificación de conexiones y asesoramiento externo en electrónica.
- **05/09/2025:** Primer prototipo con sensor KY-032 y Micro:bit.
- **12/09/2025:** Solución de errores en la lógica de detección con ayuda del profesor.
- **19/09/2025:** Obtención del motor vibratorio y planificación del circuito con transistor.
- **26/09/2025:** Discusión sobre diseño físico y montaje del prototipo.

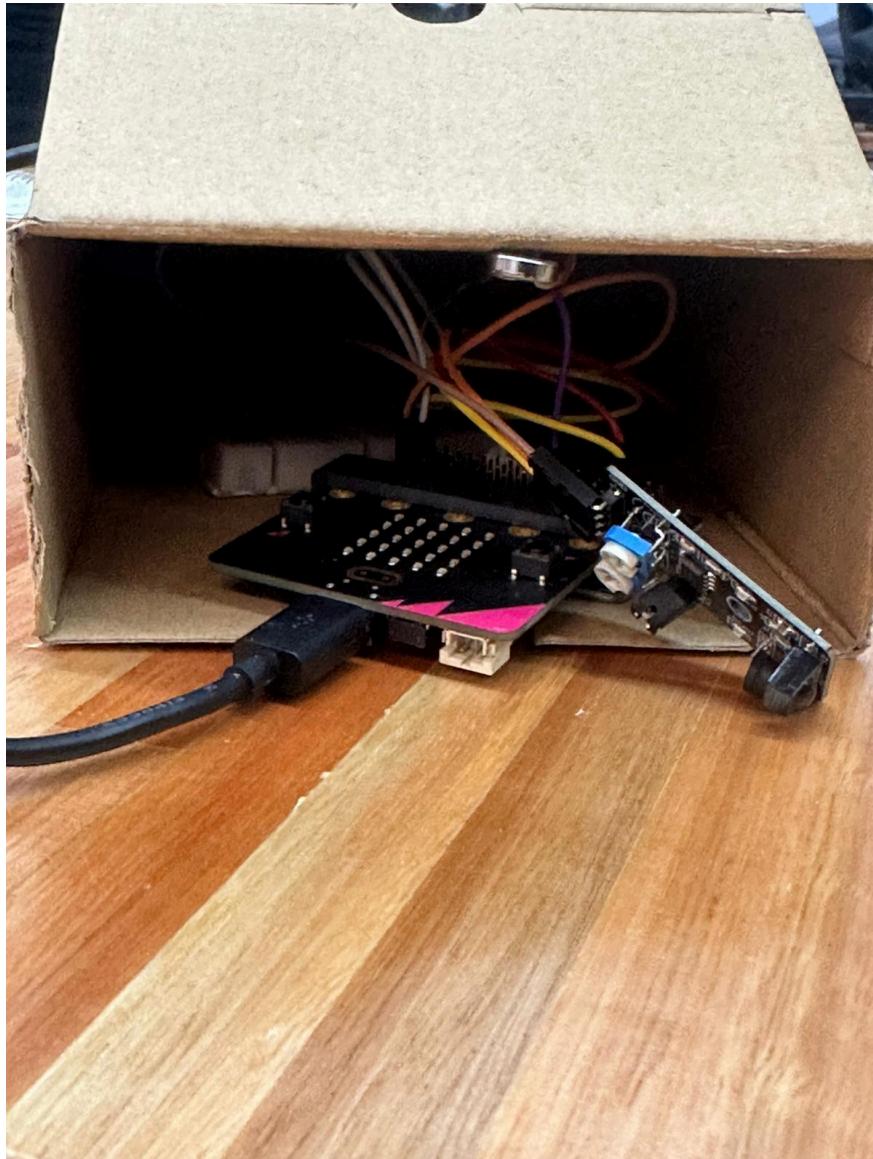


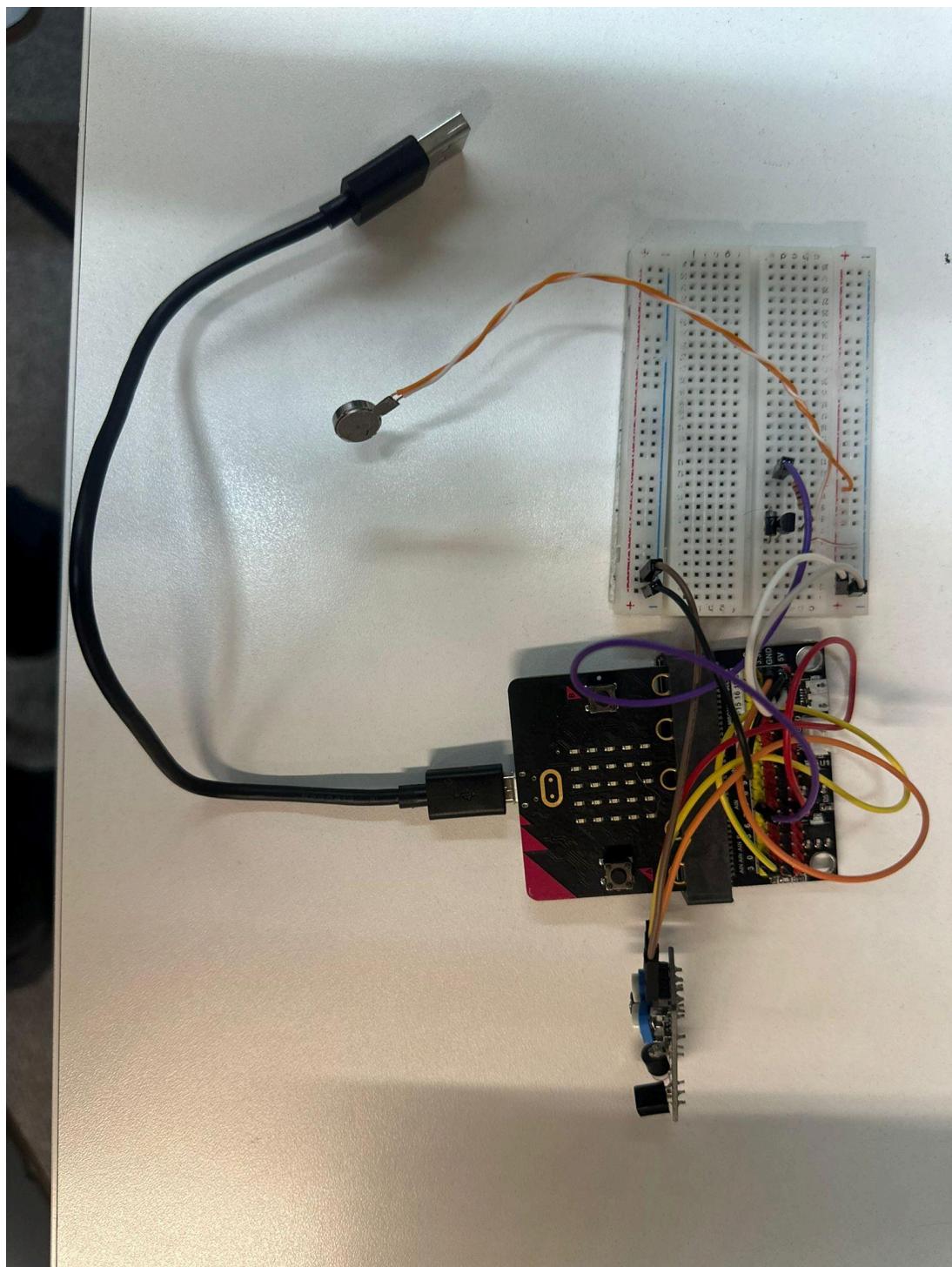
- **03/10/2025:** Soldadura y explicación técnica del circuito con ingeniero asesor.
- **10/10/2025:** Calibración del sensor y ajuste de umbral de detección.
- **17/10/2025:** Integración de componentes en la carcasa y prueba de alimentación.
- **24/10/2025:** Pruebas de campo controladas y ajustes de sensibilidad.
- **31/10/2025:** Validación final del prototipo, grabación de video y documentación técnica.



### 9.1 Prototipo final:









## 10. Referencias y Recursos

- Documentación oficial de BBC Micro:bit  
<https://microbit.org/es-es/>
- Asesoramiento del Ing. Tec. Michael Marichal
- Apoyo docente del profesor de Laboratorio STEAM+
- Material audiovisual y página web desarrollada con React, HTML, CSS y JavaScript

<https://svinas73.github.io/safestep/>