IET NITK

SMART WALKING CANE

FOR THE VISUALLY IMPAIRED

CONTENT

PROBLEM STATEMENT:

- VISUALLY IMPAIRED INDIVIDUALS FACE MOBILITY CHALLENGES WITH TRADITIONAL CANES, WHICH LACK REAL-TIME OBSTACLE DETECTION, RELYING ON SOUNDS AND TACTILE CUES THAT INCREASE ACCIDENT RISKS.

- A SMART CANE WITH INTEGRATED TECHNOLOGY CAN ENHANCE SAFETY AND INDEPENDENCE BY PROVIDING RELIABLE OBSTACLE DETECTION AND INTUITIVE FEEDBACK.

INTRODUCTION

1. A SMART WALKING CANE IS A DEVICE THAT ENHANCES MOBILITY FOR VISUALLY IMPAIRED INDIVIDUALS BY INTEGRATING SENSORS AND EMBEDDED SYSTEMS.

2. CURRENT WALKING CANES LACK THE ABILITY TO PROVIDE REAL-TIME FEEDBACK ON OBSTACLES, CREATING A DEPENDENCY ON ENVIRONMENTAL SOUNDS AND TACTILE CUES ALONE.

3. THIS PROJECT AIMS TO ADDRESS THESE CHALLENGES BY PROVIDING AN ASSISTIVE DEVICE THAT OFFERS RELIABLE, REAL-TIME OBSTACLE DETECTION AND INTUITIVE FEEDBACK, PROMOTING SAFER AND MORE CONFIDENT NAVIGATION.

CANE MODEL

CANE MODELS

Fusion 360

CANE STRUCTURE

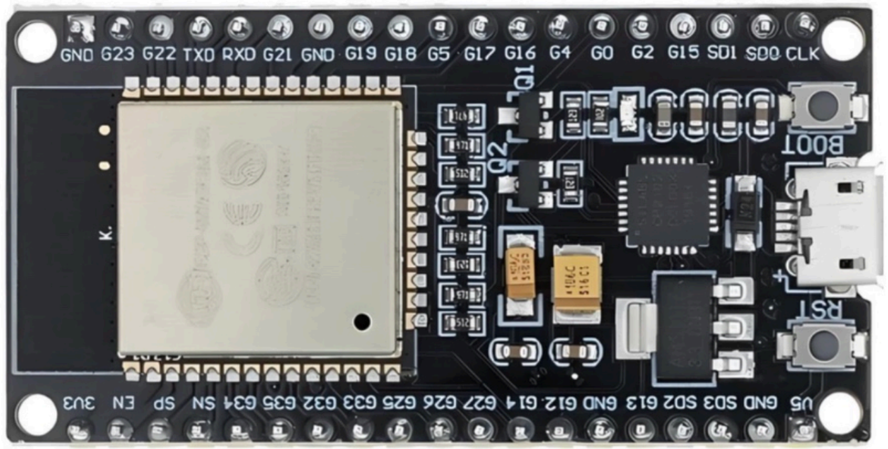
SENSOR HOUSING

HANDLE

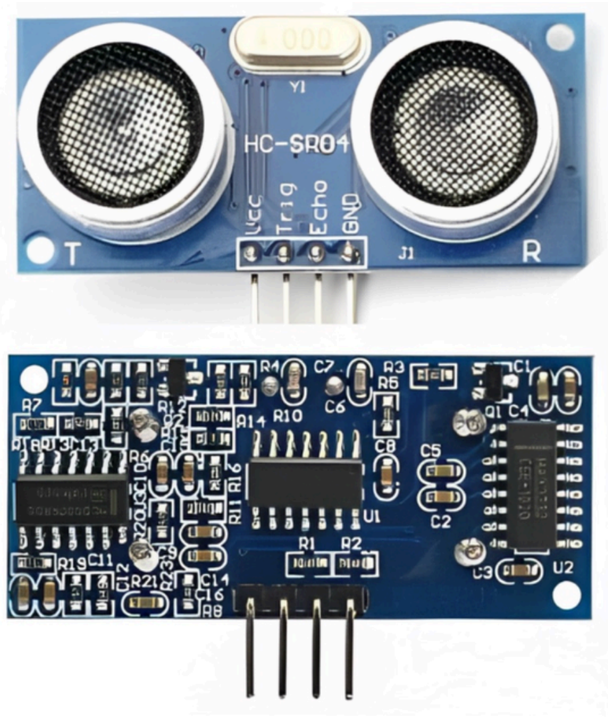
BATTERY ENCLOSURE

COMPONENTS

1. **MICROCONTROLLER** : ESP32.

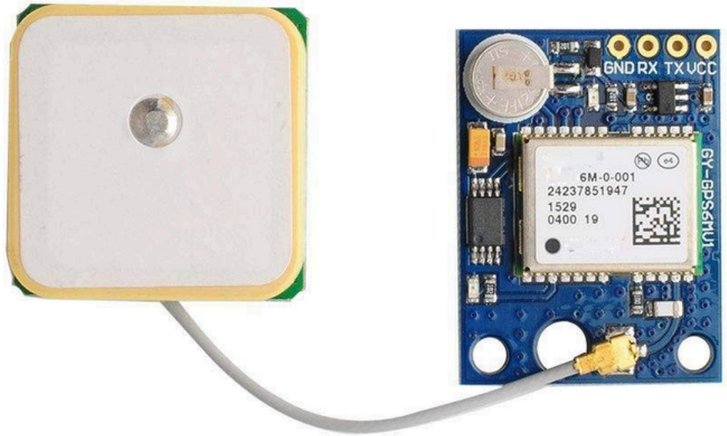


2. **SENSORS**:ULTRASONIC SENSOR (FOR OBSTACLE DETECTION).



COMPONENTS

3. **GPS MODULE** : FOR LOCATION TRACKING AND EMERGENCY ALERTS.



4. **BUZZER & VIBRATION MOTOR** : FOR FEEDBACK AND ALERTS.

METHODOLOGY

METHODOLOGY

1. ASSEMBLE THE SENSORS AND MODULES ONTO THE CANE.
2. CONNECT THE COMPONENTS WITH APPROPRIATE WIRING.
3. ENSURE PROPER POWER MANAGEMENT WITH A RECHARGEABLE BATTERY.

- **SENSOR DATA PROCESSING**:
  - PROGRAM THE MICROCONTROLLER TO READ AND PROCESS SENSOR INPUTS.
  - IMPLEMENT ALGORITHMS FOR OBSTACLE DETECTION AND FALL DETECTION.
- **COMMUNICATION SYSTEM**:
  - INTEGRATE BLUETOOTH, WI-FI, OR GPS FOR REAL-TIME TRACKING AND ALERTS.
  - DEVELOP AN APP FOR CAREGIVERS (THINGSBOARD).
- **USER FEEDBACK MECHANISM**:
  - IMPLEMENT VIBRATION OR SOUND FEEDBACK FOR WARNINGS.

CONNECTIONS

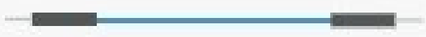
Pin Connections

This section provides details about the pin connections between the components used in the node and the ESP32 controller.

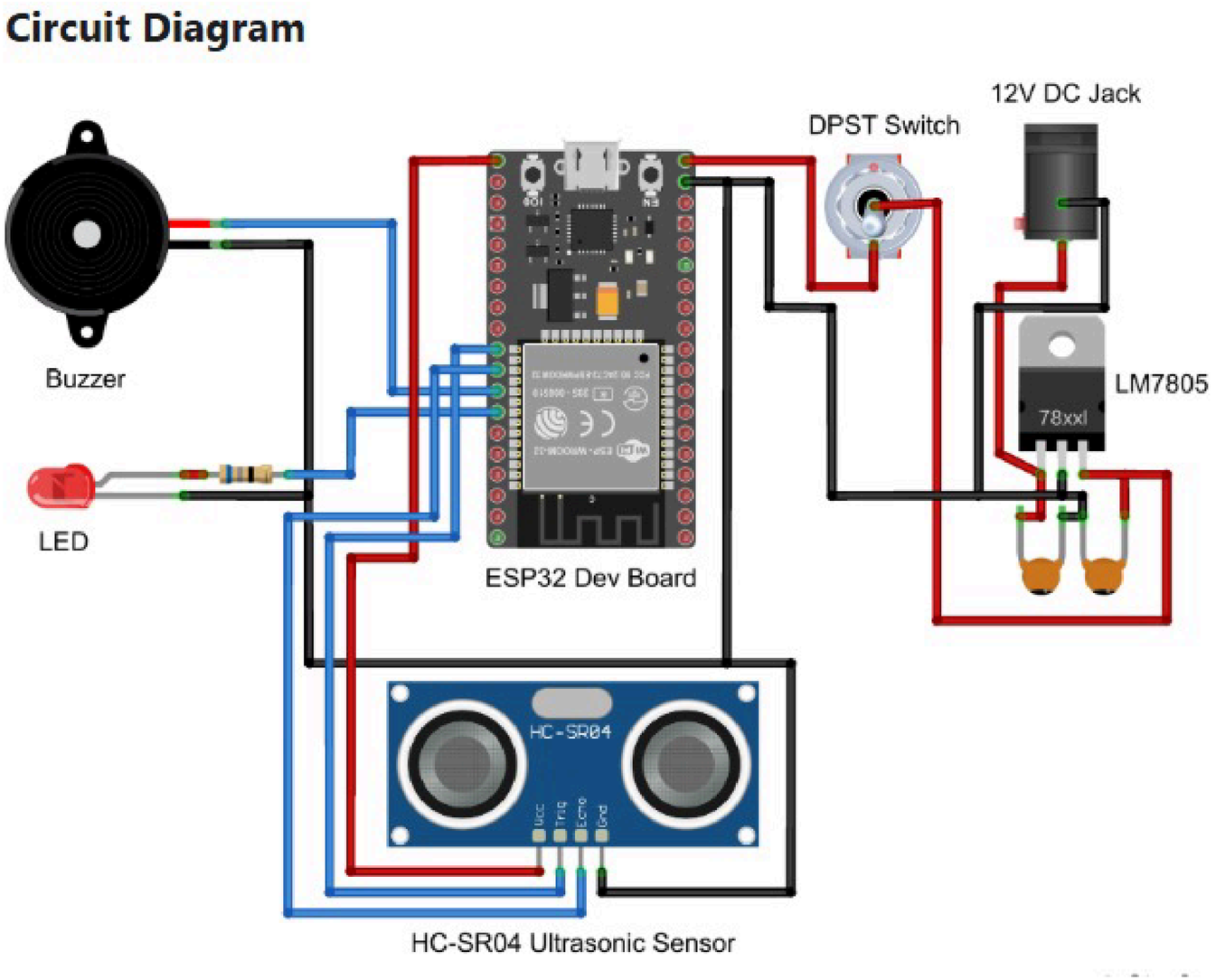
HC-SR04 Module Pin Connections

HC-SR04 Ultrasonic Sensor		ESP32 Dev Board
VCC		3.3V
GND		GND
Trig		5
Echo		18

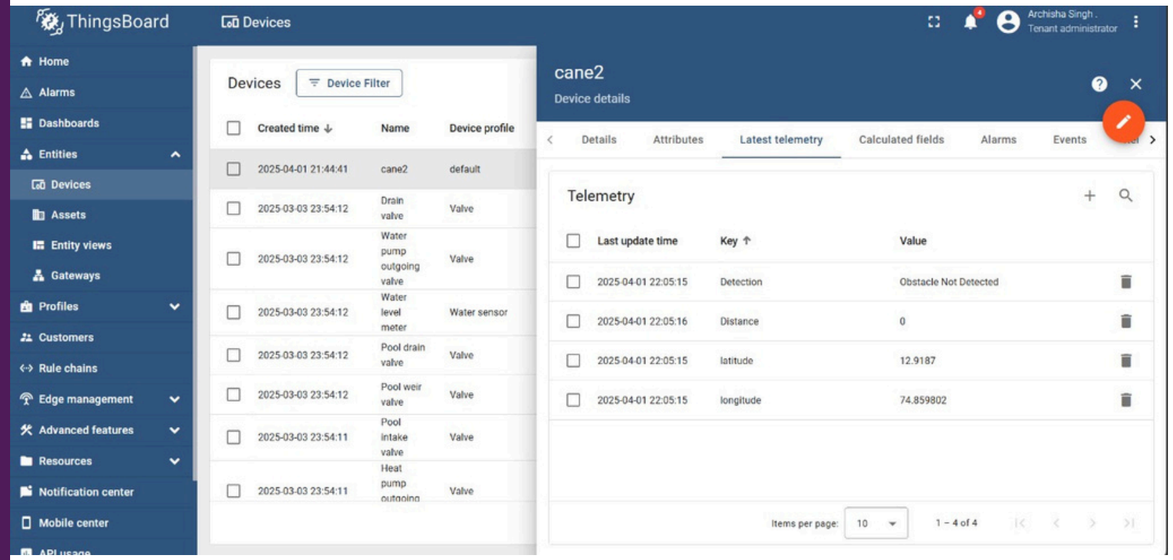
Buzzer Pin Connections

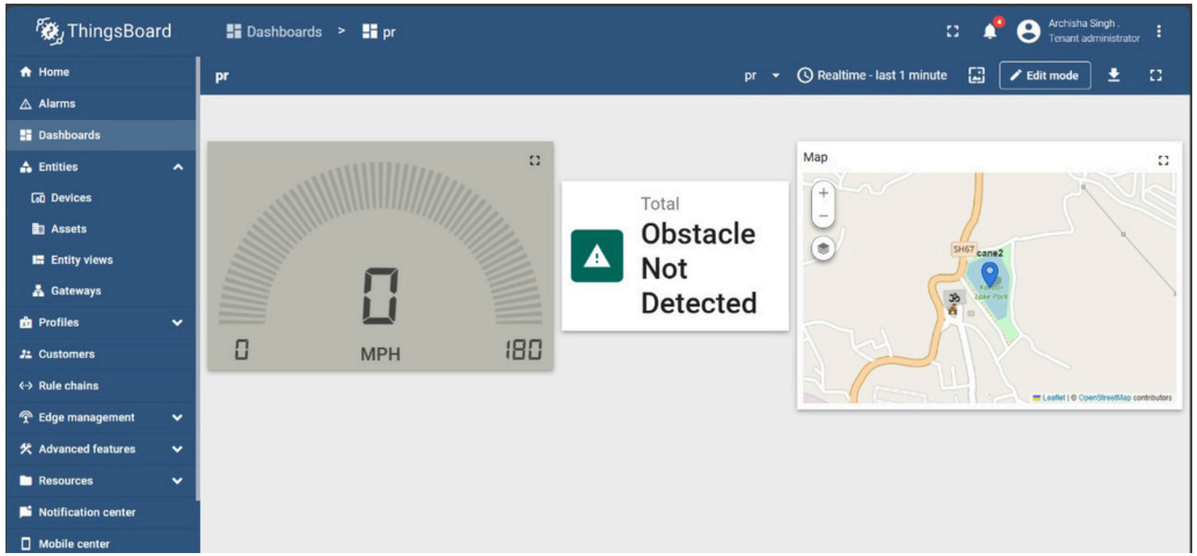
Buzzer		ESP32 Dev Board
Positive (+)		19
Negative (-)		GND

CIRCUIT DIAGRAM



OUTPUT


ThingsBoard
Devices
Archisha Singh .
Tenant administrator
Home
Alarms
Dashboards
Entities
Devices
Assets
Entity views
Gateways
Profiles
Customers
Rule chains
Edge management
Advanced features
Resources
Notification center
Mobile center
Devices
Device Filter
Created time
Name
Device profile
2025-04-01 21:44:41 cane2 default
2025-03-03 23:54:12 Drain valve Valve
2025-03-03 23:54:12 Water pump outgoing valve Valve
2025-03-03 23:54:12 Water level meter Water sensor
2025-03-03 23:54:12 Pool drain valve Valve
2025-03-03 23:54:12 Pool weir valve Valve
2025-03-03 23:54:11 Pool intake valve Valve
2025-03-03 23:54:11 Heat pump outgoing Valve
cane2
Device details
Details
Attributes
Latest telemetry
Calculated fields
Alarms
Events
Telemetry
Last update time
Key
Value
2025-04-01 22:05:15 Detection Obstacle Not Detected
2025-04-01 22:05:16 Distance 0
2025-04-01 22:05:15 latitude 12.9187
2025-04-01 22:05:15 longitude 74.859802
Items per page: 10
1 – 4 of 4


ThingsBoard
Dashboards > pr
Archisha Singh .
Tenant administrator
Home
Alarms
Dashboards
Entities
Devices
Assets
Entity views
Gateways
Profiles
Customers
Rule chains
Edge management
Advanced features
Resources
Notification center
Mobile center
pr
Realtime - last 1 minute
Edit mode
0
MPH
180
Total
Obstacle
Not
Detected
Map
cane2
5H67
Leaflet | © OpenStreetMap contributors

CAD MODEL AND CODE SNIPPETS

```
duino IDE 2.3.4
Tools  Help
Select Board
ar.ino
const int trigPin=6;
const int echoPin=7;
long time;
float distance;
speed=0.0342
void setup() {
 Serial.begin(9600);
 pinMode(trigPin, OUTPUT);
 pinMode(echoPin, INPUT);
 Serial.println("ultrasonic Sensor Test");


}

void loop() {
  // put your main code here, to run repeatedly
 digitalWrite(trigPin, LOW );
 delayMicroseconds(2);
 digitalWrite(trigPin, HIGH);
 delayMicroseconds(10);
 digitalWrite(trigPin,LOW);
 time = pulseIn(echoPin, HIGH);
 distance = (time*speed)/2;
 if (distance >= 2 && ditance <= 400)
 Serial.print("distance: ");
 Serial.print(distance);
 Serial.println(" cm");
 serial.println("out of range");
 delay(500);
 }
```

```
tch  Tools  Help
Arduino Uno
r.ar.ino
  const int buzzerPin = 9;
  const int buttonPin= 7;
  int buttonState =0;
  void setup() {
    pinMode(buzzerPin, OUTPUT);
    pinMode(buttonPin, INPUT);

  }

  void loop() {
    buttonState =
    digitalRead(buttonPin);
    if(buttonState == LOW){
    digitalWrite(buzzerPin, HIGH);
    } else{
      digitalWrite(buzzerPin, LOW);

    }



  }
```

RESULT

SKILLS LEARNT AND CONCLUSION

LEARNT TO USE FUSION 360
LEARNT TO INTEGRATE THE COMPONENTS USING ESP32
CHECKED THE WORKING OF THE COMPONENTS USING TINKERCAD.
HTTPS://DRIVE.GOOGLE.COM/DRIVE/U/2/FOLDERS/1YYE992IGYMSL3FRS5MS-XMHD942HON-R

REFERENCES

1. Ravikiran, A., & Kumar, P. (2021). "Smart Walking Stick for the Visually Impaired Using IoT." International Journal of Engineering Research, 10(2), 45-52.

2. Gupta, S., & Singh, R. (2020). "An Intelligent Walking Cane with Obstacle Detection and Navigation Assistance." IEEE Xplore. DOI: [Insert DOI]

3. Khan, M. A., et al. (2019). "Design and Development of a Smart Cane for Elderly and Disabled People." Springer Advances in Assistive Technologies.

Web References

4. WeWALK Smart Cane. (n.d.). Retrieved from https://wewalk.io

5. SmartCane by IIT Delhi. (n.d.). Retrieved from http://assistech.iitd.ac.in/smartcane.php

Mentors:

Aditya
Priya Jha

Mentees:

Aashritha
Archisha
Greeshma
Pavani

THANK YOU