Project Tutorial Report

Team Information:

Name: Masuma Akter Akhi
ID: 0242310005341250

Name: Farhan Ibn Mustafa
ID: 0242310005341325

Name: Yeasmin Akter
ID: 0242310005341207

Name: Apurba Khandaker
ID: 0242310005341302

Name: Nafisa Anjum Rakhi
ID: 0242310005341351

Section: A

Batch: 40

Course Code: SE532

Course Name: Introduction to Robotics

Teacher Information:

Md Hafizul Imran

Senior Lecturer, Department of Software Engineering

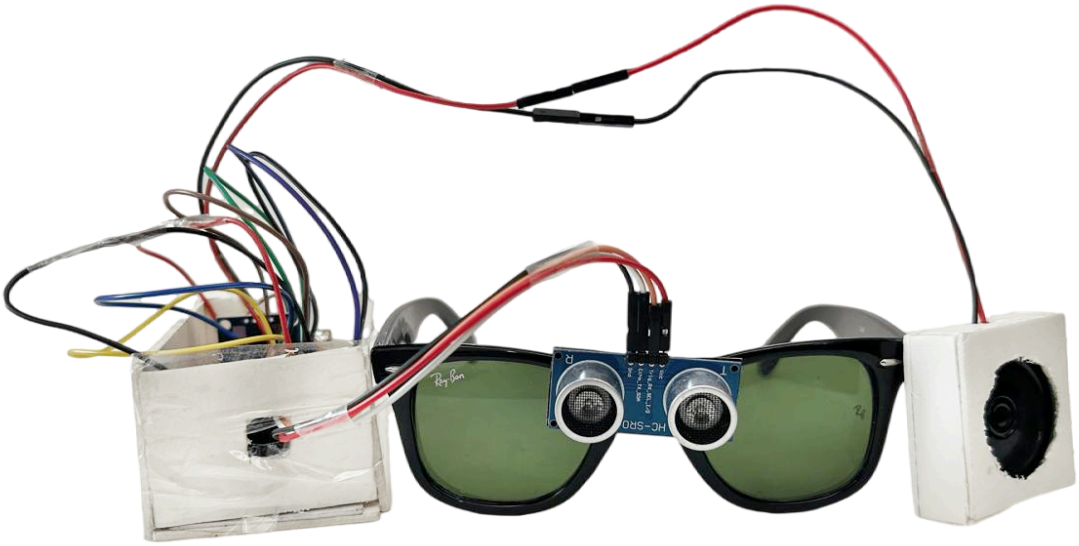
Lab In-Charge of Daffodil Robotics Lab

Convener, DIU Robotics Club

Daffodil International University

Submission Date: 9/12/2024

“ VisionBot ”



Objectives:

- To assist visually impaired individuals in navigating their environment safely.
- To detect obstacles within a preset range using ultrasonic sensors.
- To provide real-time auditory feedback through pre-recorded audio alerts.
- To ensure portability and comfort with a lightweight, wearable design.
- To demonstrate a cost-effective and user-friendly solution for mobility and safety enhancement.

Components:

Here is the list of the components that we used in the Smart Glass project:

- **Ultrasonic Sensor (e.g., HC-SR04)** – 1 Unit

    - Used for detecting obstacles by measuring the distance between the user and surrounding objects.
- **DFPlayer Mini MP3 Module** – 1 Unit
    - It helps to play pre-recorded audio alerts stored on an SD card.
- **Micro SD Card** – 1 Unit
    - Stores audio files for the DFPlayer Mini module.
- **3.5V Rechargeable Battery** – 1 Unit
    - It powers the entire system.
- **Jumper Wires and Standard Wires** – Assorted
    - To connect various components to ensure proper circuit functionality.
- **ESP32 Microcontroller** – 1 Unit
    - Serves as the main control unit, managing data from the ultrasonic sensor, GPS module, and DFPlayer.
- **Breadboard** – 1 Unit
    - Facilitates prototyping and testing of the circuit connections.
- **GPS Module (e.g., NEO-6M)** – 1 Unit
    - Provides location tracking and navigation capabilities for enhanced functionality.

Project Description:

The system here is a smart design that integrates object detection, audio alerts to help users have real-time situational awareness. Using an ultrasonic sensor, the system measures the distance of objects and provides audio feedback based on proximity through a DFPlayer Mini audio module.

Describe Clearly Each Technical Part of Our Tutorials:

1. Microcontroller

- **ESP32 Development Board:** We decided to use the ESP32 Development Board since it contains Bluetooth for wireless audio, an SD card for reading audio files, and is capable of handling several tasks at once, such as file reading and Bluetooth communication.

2. DFPlayer Mini Integration

- **DFPlayer Module:** The DFPlayer module is used to store audio files and it supports microSD cards formatted with FAT32 compatibility. This module has TX(Transmitter), RX (Receiver),VCC,GND connected to ESP32.

3. Wiring and Connectivity

1.Male-to-Male Wires: Connect components on the breadboard.

2.Female-to-Male Wires: Link modules to the ESP32 pins.

3.Female-to-Female Wires: Connect modules with direct female ports.

4. Breadboard

Serves as a prototyping platform for organizing component connections.

5. Ultrasonic Sensor

- **Purpose:** Detects obstacles and measures distance.

6. Battery

Powers the ESP32 and other components, ensuring portability for the smart glasses.

Key Libraries and Tools for Smart Glasses for the Blind

1. dFPlayermini_Fast.h :

- Used for controlling the DFPlayer Mini, a small MP3 player module.
- Plays pre-recorded audio files stored on an SD card, crucial for providing voice feedback to the user.

2. Arduino.h :

- Core library for Arduino boards.
- Provides basic functions like pin control - digitalWrite, analogRead and timing - millis, delay.

3.UART2:

- Refers to the second UART available on ESP32.
- Enables multiple device communication, for example, with GPS modules or the DFPlayer Mini.

TinyGPS++

- A lightweight library to parse GPS data.
- Gives location, including latitude and longitude, time, and other information coming from GPS modules.

Code Implementation for Smart Glasses Functionality:

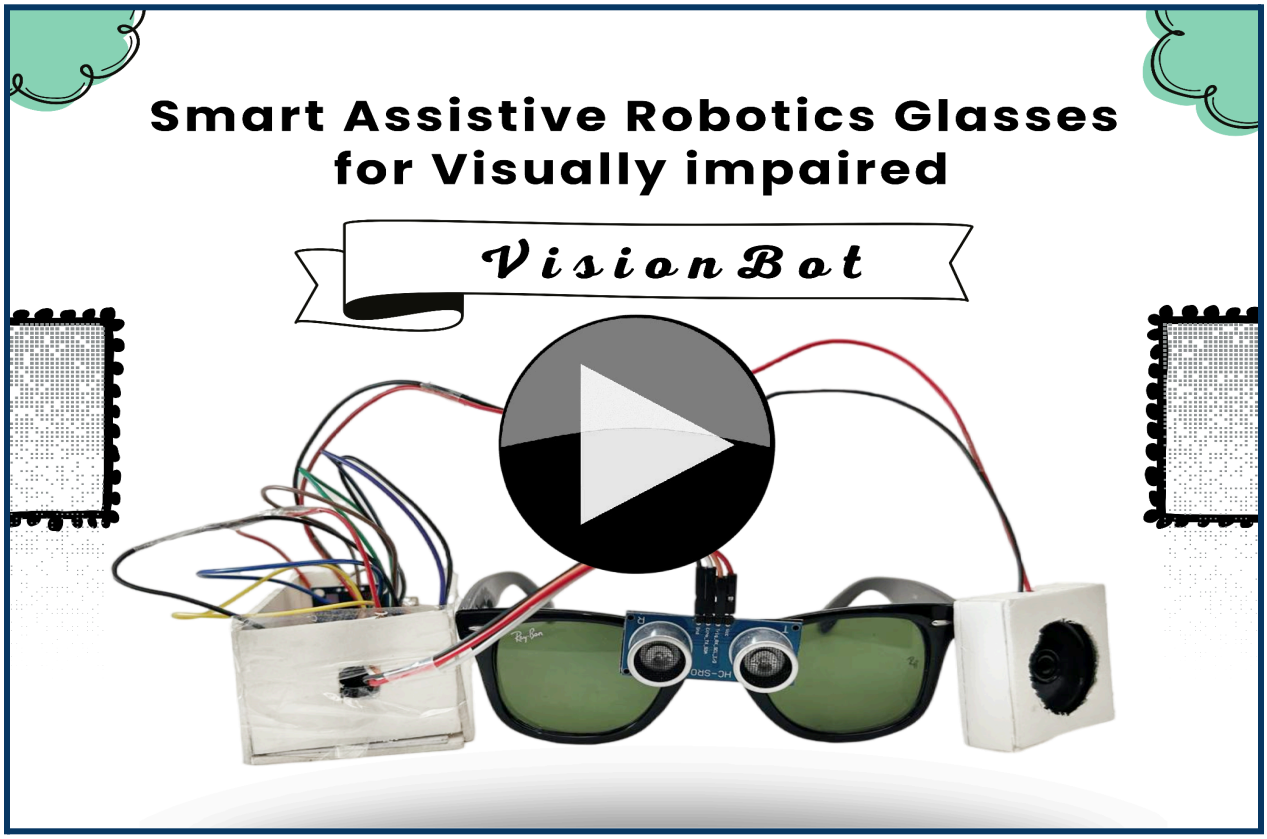
glasses.ino

```ino
#include "Arduino.h"
#include "DFPlayerMini_Fast.h"
// Ultrasonic sensor pins
#define TRIG_PIN 23 // Ultrasonic sensor trigger pin
#define ECHO_PIN 22 // Ultrasonic sensor echo pin

// Hardware Serial for DFPlayer Mini
HardwareSerial mySerial(1); // UART2
DFPlayerMini_Fast myDFPlayer;

void setup() {
  // Initialize Serial Monitor
  Serial.begin(115200);

  // Initialize Ultrasonic Sensor
  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);

  // Initialize DFPlayer Mini
  mySerial.begin(9600, SERIAL_8N1, 16, 17); // RX=GPIO16, TX=GPIO17
  Serial.println("Initializing DFPlayer...");
  if (!myDFPlayer.begin(mySerial)) {
    Serial.println("DFPlayer initialization failed!");
    while (true); // If there's an error, stop everything
  }
  myDFPlayer.volume(25); // Set volume level (0~30)
  Serial.println("Setup completed.");
}

void loop() {
  // Measure distance using Ultrasonic Sensor
  long distance = measureDistance();
  Serial.print("Distance: ");
  Serial.println(distance);
```

glasses.ino

```ino

  // Play audio based on distance ranges
  if (distance > 0 && distance <= 15) {
    Serial.println("Object detected within 15 cm!");
    myDFPlayer.play(1); // Play 0001.mp3
    delay(3000); // Wait for the audio to finish
  } else if (distance > 15 && distance <= 25) {
    Serial.println("Object detected between 15 and 25 cm!");
    myDFPlayer.play(2); // Play 0002.mp3
    delay(3000); // Wait for the audio to finish
  } else if (distance > 25 && distance <= 65) {
    Serial.println("Object detected between 25 and 65 cm!");
    myDFPlayer.play(3); // Play 0003.mp3
    delay(3000); // Wait for the audio to finish
  } else {
    Serial.println("No object detected.");
    delay(500); // Small delay to avoid continuous measurement
  }
}

// Function to measure distance using Ultrasonic Sensor
long measureDistance() {
  digitalWrite(TRIG_PIN, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG_PIN, LOW);

  // Read echo pulse duration
  long duration = pulseIn(ECHO_PIN, HIGH);

  // Calculate distance in cm
  long distance = (duration * 0.034) / 2;
  return distance;
}
```

Project Tutorial Video:



Future Scope: Advanced Integration and Functional Expansion

This project can be enhanced further in the future for more features and applications. Facial recognition is a possible addition to this project. This could be done either with image processing on the ESP32 or by adding external modules. In such a way, the system will be able to recognize who is in front of it and provide personalized audio feedback, making it useful for access control, security, or assisting visually impaired users.

Another useful upgrade would be to increase the functionality of the GPS to enable navigation. The addition of a navigation module would, for example, enable the system to

lead users to specific locations with turn-by-turn, voice guidance coupled with accurate location tracking. These would further open new possibilities and make the system even more versatile.

Conclusion:

This project demonstrates how technology can be used to make the lives of visually impaired people easier. The system, comprising an ESP32 board, ultrasonic sensor, Bluetooth, and SD card module, detects obstacles and alerts the user through Bluetooth speakers or earphones in audio. The audio feedback is customizable, stored on the DFPlayer mini module. This is an economical and practical solution using very basic components: jumper wires, breadboard, and a portable battery. It is a big step in making technology work for everyone, and it opens up new possibilities for helping people who cannot see.